Rich Probabilistic Models for Semantic Labeling

HABILITATIONSSCHRIFT
zur Erlangung der venia legendi

Fakultät für Elektrotechnik und Informatik
der Gottfried Wilhelm Leibniz Universität Hannover

vorgelegt am 28. January 2015 von

Dr.-Ing. Michael Ying Yang
aus Linhai, China

Referent:	Prof. Dr.-Ing. Bodo Rosenhahn
Korreferent:	Prof. Dr.-Ing. Monika Sester
	Prof. Dr.-Ing. Olaf Hellwich
Tag der mündlichen Prüfung:	27. 06. 2016

SELF-PUBLISHED BY THE AUTHOR

First printing, November 2016

To
Dandan & Kai

Acknowledgements

I want to thank Bodo Rosenhahn who gave me the freedom to choose my own research topics during my time as a Postdoctoral Fellow at the Leibniz University Hannover and who contributed to this monograph with fruitful discussions on various research topics. I am indebted to Wolfgang Förstner for his immense support and encouragement in writing this monograph. I would like to thank my colleagues from the Institute of Information Processing (TNT) for their contributions to a stimulating work environment. This research would not have been possible without the support and contribution of my collaborators and students: Zheng Zhang, Xiaojin Gong, Dengfeng Chai, Muhammad Shoaib, Oliver Müller, Saif Al-Shaikhli, Xuanzi Yong, Sitong Feng, and Wentong Liao. I would like to thank Dmitrij Schlesinger, Anita Sellent, Alexander Krull, Eric Brachmann, Stella Graßhof, and Hanno Ackermann for translating the German summary part.

I would also like to thank Kostas Daniilidis, Konrad Schindler, Carsten Rother, Christian Heipke, Helmut Mayer, George Vosselman, David Suter, Kim Boyer, Hans Georg Musmann, Yi Ma, Sven Dickinson, Bernt Schiele, Reinhard Klette, Daniel Cremers, Frank Dellaert, Jürgen Gall, Liqiu Meng, Liangpei Zhang, Olaf Hellwich, Monika Sester, Alper Yilmaz, Jörn Ostermann, Uwe Franke, Uwe Sörgel, Uwe Stilla, Stefan Hinz, Gunho Sohn, Clément Mallet, Yury Vizilter, Jason Corso, Sebastian Nowozin, Christoph Lampert, Raquel Urtasun, Thomas Brox, Shaogang Gong, and many others for conversations which have influenced my research.

Most important of all, I thank my family, Dandan & Kai. Their love, encouragement and tolerance have made this work possible.

Contents

List of Figures

LIST OF FIGURES

List of Tables

Chapter 1

Zusammenfassung

Diese Monographie beschäftigt sich mit Methoden und Anwendungen des semantischen Labeling. In diesem Kapitel geben wir zunächst die Motivation der Arbeit an, gefolgt von einem Überblick über unsere Beiträge zu den interdisziplinären Bereichen der Computer Vision sowie medizinischer Bildverarbeitung und Fernerkundung. Wir ausführen dabei Wahrscheinlichkeitsmodelle auf graphen. Unsere Beiträge werden mit Beispielbildern illustriert und decken drei Anwendungsgebiete ab, wobei jedem Gebiet ein separater Teil der Monographie gewidmet ist: (I) Regionsegmentierung, (II) medizinische Bildanalyse, und (III) Bildklassifizierung in der Fernerkundung. Die Übersicht Struktur dieser Monographie ist in Abbildung 1.1.

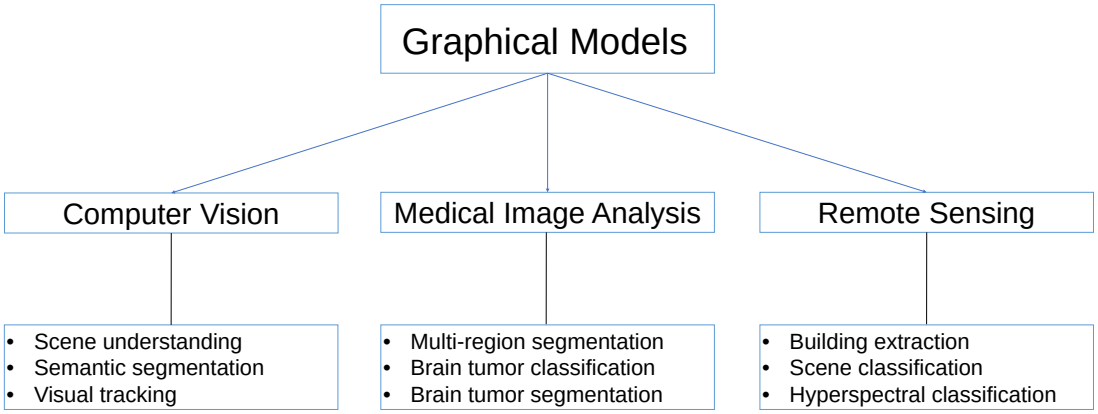


Figure 1.1: Die Übersicht Struktur dieser Monographie.

Motivation

Während Menschen stets aussagekräftige Informationen aus visuellen Daten fast mühelos extrahieren, stellt es sich heraus, dass selbst einfache Aufgaben wie Erkennung aus visuellen Daten, Detektion und Verfolgung von Objekten, Szeneninterpretation herausfordernde Probleme für Maschinen sind. Die Entwicklung künstlicher Systeme, die visuelle Informationen genauso zuverlässig wie Menschen verarbeiten können, hat viele potenzielle Anwendungen in Bereichen wie künstliche Intelligenz, Robotik, medizinische Bildgebung, Überwachung, Fernerkundung, Unterhaltung oder Sportwissenschaft. Das übergeordnete Ziel ist dabei, das menschliche visuelle System durch Computer zu simulieren.

Ein grundlegendes Ziel der Computer Vision ist es, die semantischen szenenbezogenen Informationen zu extrahieren, das nennt man „Scene Understanding“ (Szenenverstehen). Scene Understanding ist die Grundlage für viele Anwendungen: Überwachung, autonomes Fahren, Verkehrssicherheit, Roboternavigation, visuell kontrollierte mobile Navigationssysteme oder Aktivitätserkennung. Scene Understanding aus einem Bild oder einer Videosequenz erfordert viel mehr als einfach eine Aufzeichnung und Speichern oder das Extrahieren bestimmter Merkmale. Das Hauptziel ist es, eine Abbildung von Sensordaten auf semantische Informationen zu finden. Das ist eine sehr anspruchsvolle Aufgabe, unter anderem aufgrund der Variabilität der Daten. Diese Variabilität kann z.B. aufgrund von physikalischen Bedingungen auftreten, wie zum Beispiel der Beleuchtung oder die Lage der Szene relativ zum Sensor, oder durch die intrinsische Natur der Daten selbst. Daraus resultiert die Notwendigkeit der Erfassung von lokale, globale, oder dynamischen Aspekte der Szene. Um eine aussagekräftige Szenenbeschreibung zu erhalten müssen alle Informationen, die aus einer Szene extrahiert werden können, im Kontext betrachtet werden. Allerdings, während es leicht für den Menschen ist, erweist es sich als immer noch schwierig, derartige Informationen durch Computer zu erhalten.

Im Allgemeinen, lässt sich Scene Understanding als *Labeling Problem* formulieren indem jeder nicht beobachteten verborgenen Variable eine Klasse zugeordnet wird. Die Labels entsprechen verschiedenen Schätzungen wie beispielsweise einer Objektklasse bei Objektsegmentierung (Yang, 2015; Yang & Rosenhahn, 2014), einem Tiefenwert bei 3D-Rekonstruktion (Huang *et al.*, 2015), einem Pixelintensität im Fall der Bildentrauschen oder der Lage eines Objekts bei der Verfolgung (Müller *et al.*, 2013). Die Labels sind in der Regel bedingt voneinander abhängig, somit sind die Ausgabe stark strukturiert. Probabilistische graphische Modelle bieten einen allgemeinen Rahmen für die statistische Modellierung, Inferenz und das Lernen in künstlichen Vision-Systemen. Markovsche Zufallsfelder (MRF) sind die am häufigsten verwendeten graphischen Modellen im Computer-Vision, mit denen man lokale Kontextinformationen in das Model einbauen kann. In MRFs werden die Abhängigkeiten zwischen den Variablen in einer Wahrscheinlichkeitsverteilung durch Kanten zwischen den entsprechenden Knoten in einem Graphen repräsentiert. In Computer Vision wurden MRFs bei den frühen Arbeiten (Besag, 1974, 1986; Geman & Geman, 1984) volkstümlich. Eine Einschränkung dieser Modelle ist allerdings dass man damit nur lokale Merkmale verwenden kann. Diese Einschränkung wurde durch sogenannte bedingte Zufallsfelder (Conditional Random Fields – CRF) überwunden (Kumar & Hebert, 2003a; Lafferty *et al.*, 2001). Bei diesen Modellen können beliebige Funktionen verwendet werden, jedoch auf Kosten eines rein diskriminativen Ansatzes.

Das Ziel dieser Monographie ist es, die Methoden und Anwendungen des semantischen Labelings zu erforschen. Unsere Beiträge zu diesem sich rasch entwickelten Thema sind bestimmte Aspekte der Modellierung und der Inferenz in probabilistischen Modellen und ihre Anwendungen in den interdisziplinären Bereichen der Computer Vision sowie medizinischer Bildverarbeitung und Fernerkundung. Im folgenden Abschnitt fassen wir unsere wissenschaftlichen Beiträge zusammen.

Beiträge

Regionsegmentierung

Der erste Teil dieses Werks besteht aus Kapiteln, die verschiedene probabilistische grafische Modelle zusammen mit ihrer Anwendung auf Probleme der Bildsegmentierung, Objektsegmentierung und Tiefenupsampling, Szeneninterpretation, Videosegmentierung und Tracking anwenden.

Bildsegmentierung durch Bilayer Superpixel Gruppierung

Bildsegmentierung ist ein fundamentales low-level Problem der Computer Vision und der Bildverarbeitung. Es bereitet die Basis für high-level Bildverständnis wie zum Beispiel Objekterkennung, Bildersuche, Aktivitätserkennung, etc.

Obwohl es bereits eine Vielzahl von Segmentierungstechniken gibt, bleibt Segmentierung ein offenes Problem, da Bilder eine große Diversität und Mehrdeutigkeiten beinhalten. Die Aufgabe der Segmentierung ist es, Bildpixel in visuel bedeutsame Regionen einzuteilen, die für weitere Verarbeitung, z.B. bei der Erkennung, nützlich sind. Ansätze in der Literatur beinhalten Normalized Cuts (Shi & Malik, 2000), Mean Shift (Shi & Malik, 2000), grafenbasierte Methoden (Felzenszwalb & Huttenlocher, 2004b), und ultrametrische Contour Maps (Arbelaez *et al.*, 2011). In dieser Arbeit formulieren wir Segmentierung als die Aufgabe, Superpixel zu gruppieren. Wir schlagen einen neuen, grafenbasierten Segmentierungsalgorithmus vor, der es ermöglicht, gleichzeitig unterschiedliche Eigenschaften von Bilayer-Superpixel zu intergrieren. Die Grundidee ist es, Segmentierung als Gruppierung von einer Untermenge von Superpixeln zu formulieren, die einen Bilayergrafen teilt, dessen Kanten die Ähnlichkeit zwischen den Superpixeln beschreiben. Zunächst konstruieren wir dafür einen bipartiten Graphen, der Superpixel-Eigenschaften und Eigenschaften mit großer Reichweite beinhaltet. Dann werden Eigenschaften mit mittlerer Reichweite in einem hybriden Grafenmodell eingebunden. Das Segmentierungsproblem wird dann durch spektrales Clustern gelöst. Dieser Ansatz ist vollautomisch, bottom-up, und benötigt kein überwachtes Training. Diese Arbeit ist auf der Asian Conference on Pattern Recognition (ACPR) veröffentlicht worden (Yang, 2013).

Schätzung des Layouts aus Innenraumaufnahmen mit Störobjekten durch trajektorienbasierte Priors

Das Schätzen der Layouts oder der Struktur eines Innenraums ist wichtig für zahlreiche Aufgaben, z.B. Analyse von Aktivitäten (McKenna & Charif, 2004), Navigation von Robotern

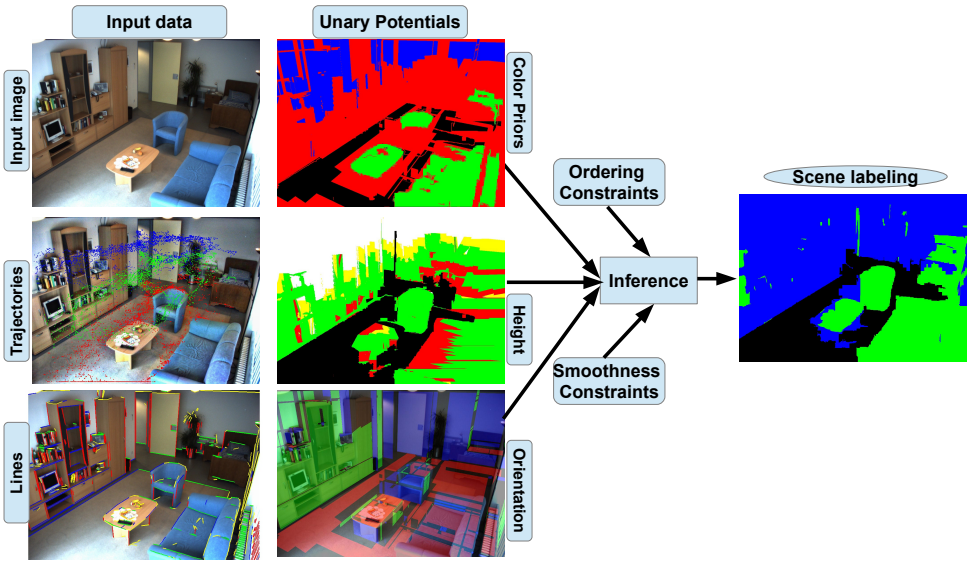


Figure 1.2: Vorgehen bei der unüberwachten, CRF basierten Szenensegmentierung für Regionen ohne Aktivität Die erste Spalte zeigt die Eingabedaten inklusive Bilder der Überwachungsszene, Trajektorien der Schüsselpunkte und Linien im Szenenbild. Die zweite Spalte zeigt die drei Merkmale, die verwendet werden um unäre Klassenpotentiale zu definieren: die Farbprioren, die relative, quantisierte Gröe und die Oberflächenorientierung. Die binäre Potentiale der Ordnungs- und der Glattheitsbedingungen werden hinzugefügt, um die Nachbarschaft homogener Regionen zu bestimmen. Die Inferenz findet die optimale Szenensegmentierung durch Energieminimierung des CRF.

(Thrun *et al.*, 2004), Szenenverständnis (Saleemi *et al.*, 2010) oder das Plazieren von Objekten (Jia *et al.*, 2013). Der aktuelle Stand der Technik verwendet für diesen Zweck entweder räumlich Bildmerkmale (Hedau *et al.*, 2009) oder trajektorienbasierte zeitliche Information (Zhang *et al.*, 2011). Keines der beiden Merkmale ist jedoch ausreichend um das Layout eines Innenraums zu bestimmten. Die größte Herausforderung für merkmalsbasierte Techniken ist die Tatsache, dass die meisten Innenräume mit Möbeln und Dekoration bestückt sind. Diese verschleiern meist die geometrische Struktur der Szene und verdecken die Grenzen zwischen Wand und Boden. Anschein und Verteilung von Störddaten kann in verschiedenne Innenräumen stark voneinander abweichen, so dass es extrem schwer ist, sie konsistent zu modellieren. Aus ähnlichen Gründen clustern trajektorenbasierte Techniken normalerweise die Daten der Trajektorien und modellieren nur die Pfade (Zhang *et al.*, 2011). Sie kümmern sich nicht um Störddaten oder Ruheplätze in der Szene.

Obwohl Bildmerkmale und Trajektionsdaten an sich nicht ausreichend sind für zuverlässige Bestimmung des Szenenlayouts, können sie doch für eine zuverlässigere Bestimmung kombiniert werden. Wir schlagen einen Algorithmus vor, der das szenensemantische Kontextmodell durch Bildsegmentierung unüberwacht lernt. Dazu verwenden wir die Trajektorien nicht direkt zur Layoutschätzung, sondern unser Segmentierungsalgorithmus verwendet

sowohl Bildmerkmale als auch trajektorienbasierte Merkmale. So können wir auch die Ruhe- und Sitzflächen in der Szene modellieren. Dazu nehmen wir an, dass es eine statische und unkalibrierte Überwachungskamera in der Szene gibt. Unter Verwendung von Pixel-Farbe und perspektivischen Hinweisen der Szene, wird jeder Pixel einer Klasse zugewiesen, die Sitzgelegenheiten, den Boden, oder statische Regionen wie Wände oder Decke beschreiben. Die globale topologische Ordnung der Klassen, in der sich z.B. Sitzgelegenheiten und Hintergrund über dem Fußboden befinden müssen, wird über eine Ordnungsbedingung lokal in ein Conditional Random Field (CRF) eingefügt. Abbildung 1.2 gibt einen Überblick über CRF basierte Segmentierung für unüberwachte Szenenlayoutbestimmung. Ein Inferenzalgorithmus, der auf Graph Cut basiert, wird auf unser CRF angewendet, um die endgültige Segmentierung oder das Layout zu bestimmen. Die vorgeschlagenen Methode liefert auch auf schweren realen Szenen sehr genaue Segmentierungen. Diese Arbeit ist in Image Vision Computing (IVC) (Shoaib *et al.*, 2014) erschienen.

Gemeinsame Objektsegmentierung und Tiefenupsampling

In den letzten Jahren ist die gemeinsame Verwendung von Tiefensensoren und Kameras immer bliebter geworden. Davon profitieren auch viele Computer Vision Anwendungen. Diese Arbeit wendet sich der tiefengestützten objektbasierten Bildsegmentierung und bildbasierten Tiefenupsampling zu. Erstere Aufgabe verwendet Tiefeninformation um Bilder in Regionen zu segmenitieren, die Objekten entsprechen. Vorherige Arbeiten zu dieser Aufgabe verlassen sich zumeist auf dichte Tiefenkarten aus Stereobildpaaren (Ladickỳ *et al.*, 2012; Sengupta *et al.*, 2013). In letzter Zeit, sind auch dünnbesetzte 3D Punktwolken und die rekonstruierten, korrespondierende Tiefenkarten in semantischer Segmentierung für Straßenszene benutzt worden (Chen *et al.*, 2014; Huang *et al.*, 2014). In diesen Arbeiten, ist Tiefeninformation entweder als geometrischer Prior oder als feste Bedingung im Rahmen eines Markov Rrandom Fields (MRF) integriert worden, um so die Segmentierung zu verbessern. Unsere zweite Aufgabe, nämlich hochaufgelöste Tiefenkarten aus wenigen Meßwerten zu erzeugen, verwendet hochaufgelöste Bilder als Führung. Existierende Forschung verwendet hauptsächlich Techniken wie bilaterales Filtern (Yang *et al.*, 2007), Sparse Representation (Gong *et al.*, 2014), oder MRF (Diebel & Thrun, 2005; Zhu *et al.*, 2010). Es ist jedoch eine gemeinsame Schwäche beider Aufgaben, dass sie anfällig für Fehler in die Anleitung sind. Genauer gesagt, die Genauigkeit der Bildsegmentierung läßt nach, wenn die verwendete Tiefenkarte verrauscht ist oder verwaschene Kanten aufweist. Genauso führt ein Segment, das mehrere Objekte durchqueren, zu einer falschen Auflösungserhöhung der Tiefenkarte. Um solche eine Fehlerfortpflanzung zu verhindern, schlagen wir vor, diese beiden Probleme simultan zu lösen.

Wir schlagen eine gemeinsame Methode vor, um tiefenassistierte Objektsegmentierung und bildassistent Tiefenupsampling durchzuführen. Dazu formulieren wir diese beiden Aufgaben als ein Bi-Task Labeling Problem, das in einem Markov random field (MRF) definiert wird. Eine Methode alternierender Richtungen wird für die gemeinsame Inferenz angepasst, so dass jedes Unterproblem alternierend gelöst wird. Dabei wird das Unterproblem der Bildsegmentierung mit Graph Cuts gelöst, ein Algorithmus der diskrete Objektlabel sehr effizient berechnet. Tiefenupsamlling wird durch das Lösen eines linearen Gleichungsystems adressiert, das kontinuierliche Tiefenwerte liefert. Wie Abbildung 1.3 zeigt, werden durch dieses Vorgehen

robuste Ojektsegmentierungen und genaue Tiefenkarten erreicht. Diese Arbeite ist in IEEE Signal Processing Letters (SPL) (Huang *et al.*, 2015).

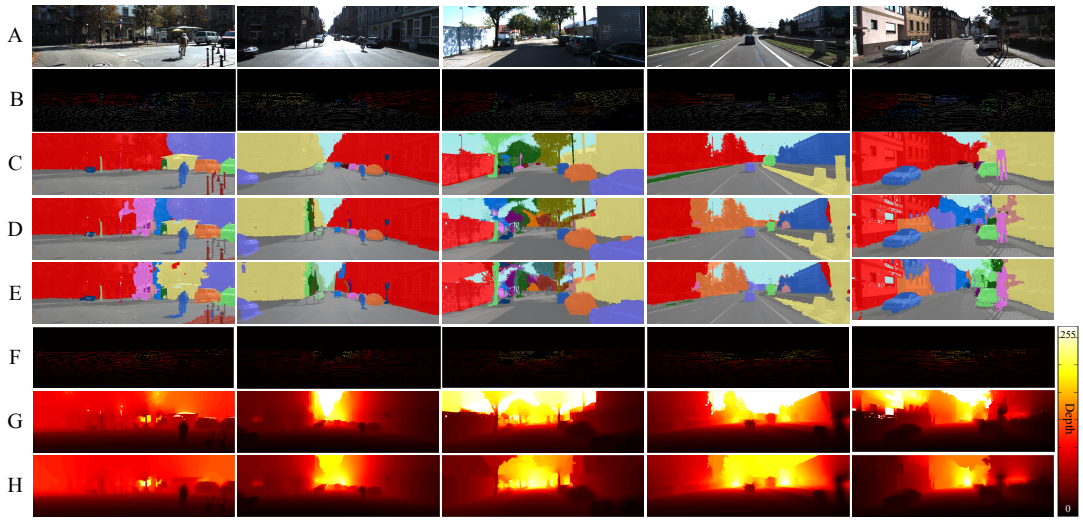


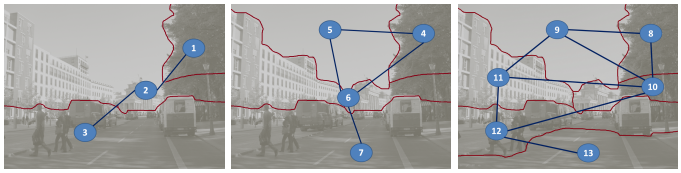
Figure 1.3: Ergebnisse der Objektsegmentierung und des Tiefenupsamplings auf dem KITTI Datensatz. (A) Eingabebild. (B) Objekte Saat. (C) Wahre Objektsegmentierung. (D) Ergebnis der gemeinsamen Segmentierung. (E) Ergebnisse einer Segmentierung ohne Tiefenupsampling. (F) Dünnbesetzte Tiefenkarte. (G) Ergebnisse des gemeinsamen Tiefenupsamplings. (H) Tiefenupsampling ohne Objektsegmentierung.

Ein generisches probabilistisches graphisches Modell zur regionsbasierten Szeneninterpretation

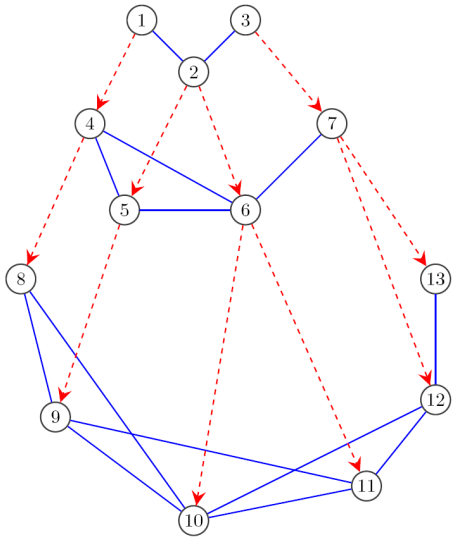
Die Aufgabe der semantischen Szeneninterpretation ist es, die Bildregionen und ihre Beziehungen untereinander semantischen Klassen zuzuordnen. Dies ist ein essentieller Bestandteil vieler Computer-Vision-Anwendungen, wie z.B.: Objekterkennung, 3D-Rekonstruktion oder Wahrnehmung in der Robotik. Auf Grund der mehrdeutigen Erscheinung der semantischen Klassen in verschiedenen Bilder, stellt die korrekte semantische Szeneninterpretation eine schwierige Herausforderung dar (Tsotsos, 1988). Die Ursache dieser mehrdeutigen Erscheinung liegt zum einen in den physikalischen Bedingungen, wie z.B. der Beleuchtung oder der Pose der Beispiel Szenenkomponenten relativ zur Kamera.

Bilder künstlicher Szenen, wie z.B. Bilder einer Fassade, zeigen starke kontextuelle Abhängigkeiten in Form von räumlichen Interaktionen zwischen verschiedenen Komponenten. Benachbarte Pixel gehören oft der selben Klasse an und unterschiedliche Regionen erscheinen oft in bestimmten räumlichen Konfigurationen. Die Modellierung solcher räumlichen und hierarchischen Strukturen ist entscheidend für das Erreichen einer guten Klassifikationsgenauigkeit.

Graphische Modelle, ob gerichtet oder ungerichtet, bieten ein bewährtes Mittel zur statistischen Modellierung der oben genannten Abhängigkeiten. Für diesen Zweck werden häufig zwei Arten graphischer Modelle eingesetzt: Bayesnetze (BNs) (Sarkar & Boyer, 1993),



(a) Multi-scale segmentation



(b) The graphical model

Figure 1.4: Illustration der Struktur des graphischen Modells. Blaue Kanten zwischen den Knoten repräsentieren, die Nachbarschaft innerhalb einer Skala (ungerichtete Kanten), rote gestrichelte Kanten repräsentieren die hierarchischen Beziehungen zwischen verschiedenen Regionen (ungerichtete oder direkte Kanten).

basierend auf gerichteten Graphen und Random-Fields (RFs) (Besag, 1974), basierend auf ungerichteten Graphen. RFs beschreiben hauptsächlich gegenseitige Abhängigkeiten, wie z.B. räumliche Korrelation. BNs werden hingegen meist zur Modellierung von kausalen Beziehungen verwendet. Beide Arten von Modellen wurden erfolgreich in Computer-Vision eingesetzt. Allerdings haben beide auch bestimmte Beschränkungen in Bezug auf die Repräsentation der Beziehungen verschiedener Variablen. BNs sind nicht geeignet um symmetrische Beziehungen darzustellen. RFs bieten eine natürliche Möglichkeit zur Modellierung solcher Beziehungen. Sie sind jedoch nicht geeignet um kausale Abhängigkeiten zu beschreiben. Der Hauptbeitrag unserer Arbeit ist die Entwicklung eines generischen statistischen Graphischen Modells zur Szeneninterpretation, siehe Abbildung 1.4, welches verschiedene Typen von Bild-Features, sowie räumlicher struktureller und hierarchischer Information nahtlos integriert. User Modell vereinigt Ideen existierender Methoden, wie z.B. Conditional-Random-Fields (CRFs). Die Arbeit ist auf der International Conference on Computer Vision Theory and Applications (VISAPP) (Yang, 2015) veröffentlicht worden.

Videosegmentierung mit gemeinsamen Objekt- und Trajektorienlabelling

Unüberwachte Videoobjektsegmentierung ist ein schwieriges Computer-Vision-Problem, da hier grosse Datenmengen verarbeitet werden müssen und das Erscheinungsbild von Objekten sich im Laufe des Videos signifikant verändern kann. Objektsegmentierung bietet die Grundlage für eine Vielzahl potentieller Anwendungen, wie z.B. Objekterkennung, 3D-Rekonstruktion, Aktivitätserkennung und Video-Retrieval. Auf Grund dieser vielen Anwendungsmöglichkeiten, gibt es in den letzten Jahren eine zunehmende Zahl von Arbeiten (Grundmann *et al.*, 2010; Lee *et al.*, 2011), die sich mit dem Thema beschäftigen. Viele Ansätze erweitern Einzelbildmethoden zur Anwendung auf mehrere Bilder, wobei sie die Redundanz entlang der Zeitachse und die Glattheit des Bewegungsfeldes ausnutzen. Solche Ansätze leiden unter Problemen wie Drift, Verdeckung und dem veränderlichen Erscheinungsbild der Objekte. Die Nutzung von Features, die sich zeitlich über mehrere Videobilder erstrecken, könnten Helfen diese Probleme zu lösen. In der Tat ist es so, dass Videomaterial viele solcher Langzeitinformationen enthält. Beispiele sind Objektbewegung, zeitliche Kontinuität, sowie die Interaktion von Objekten über einen grösseren Zeitraum. Bewegungsegmentierung nutzt solche Informationen und formuliert so eine Clusteraufgabe um Pixel in aller Frames zu gruppieren. Existierende Bewegungsegmentierungsmethoden liefern allerdings nur an diskreten und vereinzelten Positionen Resultate (Brox & Malik, 2010).

Um die oben genannten Probleme zu bewältigen formulieren wir Bild- und Bewegungsegmentierung als gemeinsame Aufgabe. Hier stellen wir eine Methode zur Segmentierung des Vordergrungs in Raum und Zeit vor. Unser Ansatz beachtet Objektränder, wie in Abbildung 1.5 gezeigt, und erzeugt zudem ein Trajektorien-Labelling. Anders als bisherige Meth-

Figure 1.5: Videoobjektsegmentierung.

oden segmentieren wir die Pixel mit Hilfe eines neuen, räumlich dicht, zeitlich jedoch dünn formulierten, graphischen Modells.

Der wissenschaftliche Hauptbeitrag unserer Arbeit ist ein vollautomatischer Bottom-Up-Ansatz zur Kombination von Objekt- und Bewegungssegmentierung. Unser Ansatz ist als Inferenz in einem vereinigtem CRF-Modell formuliert. Das CRF beschreibt Pixelklassifikation und Trajektorienclustering in einer einzigen Energiefunktion, die sowohl dichte lokale Interak-

tionen als auch einige globale Abhängigkeiten zusammenfasst. Wir optimieren über Pixel und Trajektorien in einem gemeinsamen Lösungsraum mit Hilfe eines Raum-Zeit-CRF: Sowohl Vordergrundsegmentierung als auch Trajektorienclustering werden mit Hilfe von Potentialfunktionen abgebildet. Ein auf Koordinatenabstieg basiertes Optimierungsverfahren wird verwendet um die Inferenz in dem Modell durchzuführen. Nach unsrem Kenntnisstand ist dies die erste Arbeit, die Objektsegmentierung und Trajektorienclustering in einem gemeinsamen probabilistischen Modell kombiniert. Die Arbeit erschien auf der IEEE Winter Conference on Applications of Computer Vision (WACV) (Yang & Rosenhahn, 2014).

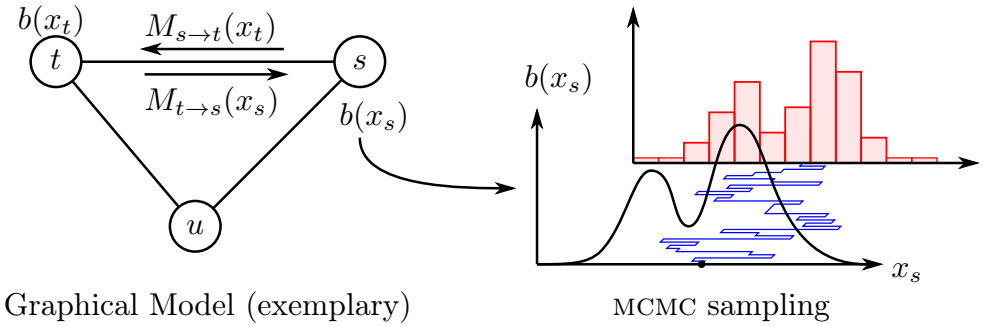


Figure 1.6: Particle Belief Propagation. Links: Message-Passing-Mechanismus. Rechts: MCMC-Sampling des Beliefs $b(x_s)$ mit einer Beispielhaften Samplekette eines Particles (blau) und dem korrespondierendem Histogramm (rot).

Slice Sampling Particle Belief Propagation

Markov-Random-Fields (MRFs) bilden ein wirksames Instrument zur Modellierung von Abhängigkeiten zwischen beobachteten Variablen. Inferenz ist in solchen Modellen ein Problem, dass bereits in der Vergangenheit breit angegangen wurde. MRFs und ihre Inferenzmethoden können in zwei Kategorien eingeteilt werden, in diskrete und kontinuierliche Labelling-Probleme. Die meisten Arbeiten zum Thema Inferenz in MRFs befassen sich mit diskreten Problemen (Boykov *et al.*, 2001; Kolmogorov, 2006). Oft können solche Methoden nicht ohne weiteres auf Probleme angewendet werden, für die ein kontinuierlicher Label-Raum eine natürlichere Wahl wäre. Das Relational-Feature-Tracking bietet hierfür ein Beispiel (Salzmann & Urtasun, 2012). In jüngerer Zeit wurden Message-Passing-Algorithmen zur Anwendung auf kontinuierlichen, statt auf diskreten, Problemen vorgeschlagen (Ihler & McAllester, 2009; Peng *et al.*, 2011; Sudderth *et al.*, 2010). Diese Methoden nutzen Markov Chain Monte Carlo (MCMC) Sampling um Message-Verteilungen zu approximieren. Alle bisherigen Ansätze basieren auf Metropolis-Hastings-Sampling. Diese Sampling-Strategie besteht aus zwei Schritten: (a) Kandidaten-Partikel werden aus einer Proposal-Verteilung gezogen, die leicht zu samplen ist. (b) Die Kandidaten werden auf Basis einer Transitionsverteilung zufällig entweder akzeptiert oder verworfen. Bei dieser Strategie ist die Wahl der Proposal-Verteilung von grosser Wichtigkeit. Hier muss ein Kompromiss zwischen dem schnellen Erforschen des Lösungsraums (eine breite Verteilung ist vorteilhaft) und einer hohen Akzeptanzrate (geringe Be-

wegung ist vorteilhaft) gefunden werden. Eine schematische übersicht wird in Abbildung 1.6 dargestellt.

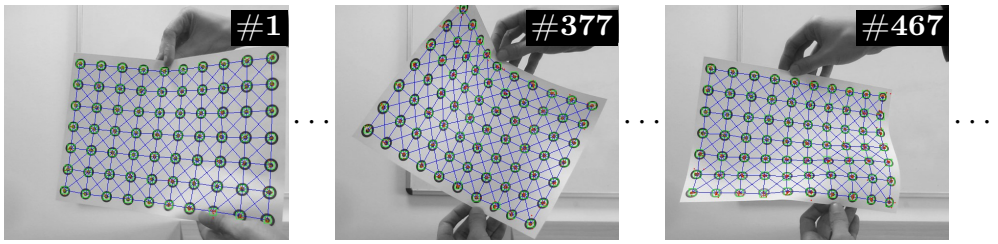


Figure 1.7: 2D-Feature-Tracking-Beispiel.

Der wissenschaftliche Hauptbeitrag unserer Arbeit ist ein neuer Particle-Belief-Propagation-Algorithmus, der Slice-Sampling (SPBP) anstatt Metropolis-Hastings-Sampling verwendet. Unsere Methode nutzt die Struktur der PBP-Message-Passing-Gleichungen aus um direktes Sampling aus der Zielverteilung zu ermöglichen. Hierzu benötigt sie keine schwer zu optimierende Proposal-Verteilung. Voraussetzung für unsere Methode ist allerdings eine analytische Beschreibung oder Begrenzung der Potentialfuntionen. Unsere Methode wurde anhand einer komplexen 2D-Relational-Feature-Tracking-Anwendung (siehe Abbildung 1.7) verifiziert. Die Arbeit erschien auf der IEEE International Conference on Computer Vision (ICCV) (Müller *et al.*, 2013).

Medizinische Bildanalyse

Im zweiten Teil dieser Habilitation werden Ansätze für die medizinische Bildanalyse beschrieben, die sowohl einen Graph Prior, als auch Dictionary Learning verwenden und angewandt werden für Semantic Labeling, Segmentierung, Tumorklassifikation und -segmentierung.

Multi-Region Labeling und Segmentierung unter Verwendung eines Topologischen Vorwissens und Atlas-Informationen in Bildern von Gehirnen

Die medizinische Bildsegmentierung und das Erkennen anatomischer Strukturen, entsprechend der vorliegenden Art des Gewebes, sind wichtig für eine akkurate Diagnose und Therapie. Wir schlagen einen neuen Ansatz für Labeling und Segmentierung mehrerer Bereiche/Regionen vor. Dieser basiert auf einem topologischen Vorwissen des zu segmentierenden Graphen, Registrierung der Label und topologischen Informationen eines Atlas, unter Verwendung einer Multi-Levelset Energieminimierung. Wir verwenden einen topologischen Graph Prior und Atlas Informationen um eine Kontur zu erstellen, basierend auf der Topologie, repräsentiert durch Relationen im Graph. Diese Methode ist dazu in der Lage, angrenzende Objekte mit sehr ähnlichen Grauwerten zu segmentieren, die mit Standardmethoden nur schwer zu trennen sind. Der topologische Graph wird von einem schlecht aufgelösten und verrauschten Bild auf die topologische Information des Atlas registriert, um die Labels der Regionen zu erhalten. Wir präsentieren einen Graph Prior und Registrierungstechniken, um zu erklären, wie sie zu präzisen

Segmentierungen und Klassifikationen mehrerer Regionen führen. Der vorgestellte Algorithmus kann verrauschte und schlecht aufgelöste Regionen von Gehirnen aus MRI Bildern unterschiedlicher Modalität segmentieren und klassifizieren, wie in der Abbildung 1.8 zu sehen.

Diese Arbeit erschien in Computerized Medical Imaging and Graphics Journal (CMIG) (Al-Shaikhli *et al.*, 2014c).

Gehirntumorklassifikation unter Verwendung von Sparse Coding und Dictionary Learning

Die frühe Erkennung von Gehirntumoren ist für eine effektive Behandlung sehr wichtig. Die Klassifikation von Gehirntumoren wird als eine der wichtigsten und anspruchsvollsten Aufgaben in der medizinischen Bildverarbeitung angesehen, da es schwer fällt, relevante Informationen zu extrahieren, mit denen der Tumor von normalem Gehirngewebe zu unterscheiden ist. Der Beitrag besteht aus einer modifizierten Sparse Coding und Dictionary Learning basierten Klassifikation. Wir verwenden die K-SVD Methode, um das Dictionary und die Sparse Coding Steps zu aktualisieren. Auf Grund der hohen Ähnlichkeit der Pixelwerte zwischen normalen Gewebe und Tumorgewebe und darüberhinaus der Variabilität in Form, Ort und Grösse des Tumors, ist zusätzlich eine Verwendung von topologischen und texturbasierten Merkmalen gerechtfertigt, um das Dictionary zu lernen. Basierend auf der (unveränderten) Topologie eines

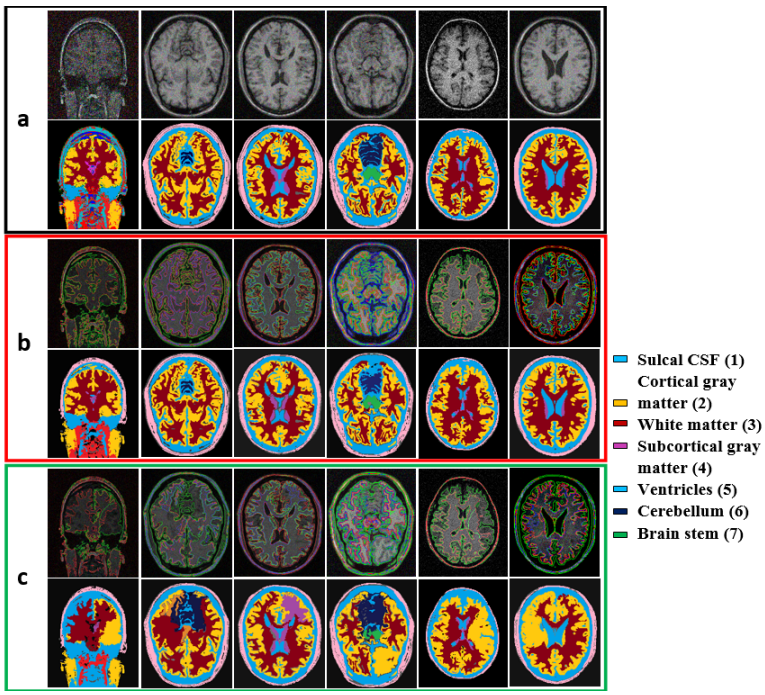


Figure 1.8: Beispiele eines multi-region Labeling und Segmentierung. (a) Stellt die Eingabebilder und die Ground Truth Daten da, (b) zeigt die Segmentierung mit einem topologischen Graphen und Atlas Information, (c) zeigt die Segmentierung mit einem topologischen Graphen ohne Atlas Information.

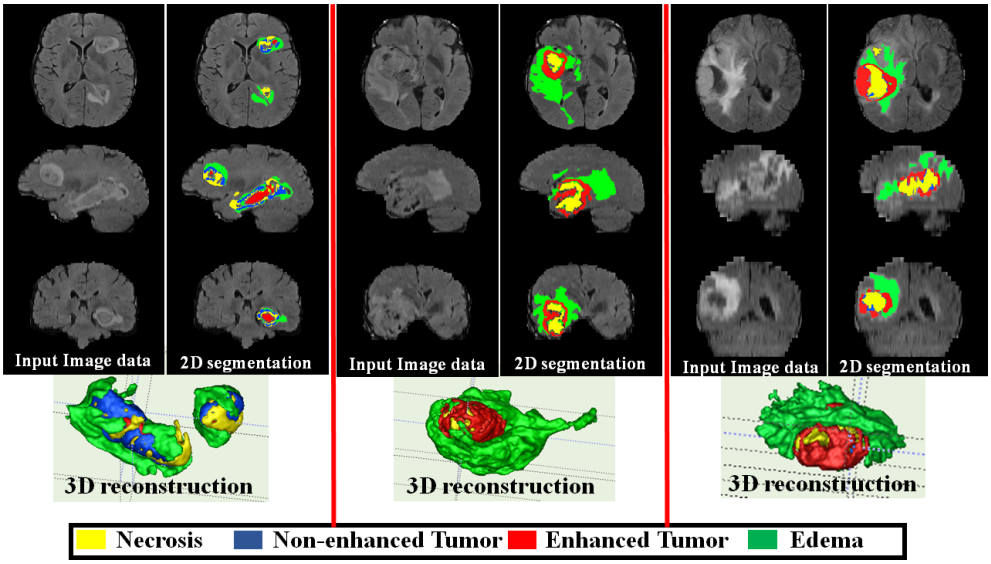


Figure 1.9: 3D Gliom Segmentierung der BraTS Trainingsdaten. In jedem Beispiel enthält die erste Spalte die Eingabebilder (axiale, koronale und sagittale Ebene), die zweite Spalte die 2D Tumor Segmentierung und die vierte Zeile die 3D Rekonstruktion des Tumors.

normalen Gehirns treffen die topologischen Merkmale eine Aussage darüber, ob es sich um einen normalen oder abnormalen Fall handelt. In Anwesenheit eines Tumors wird sich die Topologie eines normalen Gehirns verändern. Zusätzlich geben die texturbasierten Merkmale eine gute Unterscheidung der Typen des Gehirntumors. Das Neue an unserem Algorithmus ist die Verwendung von Topologie- und Textur-basierten Merkmalen für das Lernen, anstatt direkt auf den Pixelwerten zu lernen. Diese Arbeit erschien auf der IEEE International Conference on Image Processing (ICIP) (Al-Shaikhli *et al.*, 2014a).

Coupled Dictionary Learning für Automatische Mulit-Label Gehirntumorsegmentierung in Flair MRI Bildern

Wir schlagen einen neuen gekoppelten Dictionary Learning Ansatz vor (ein Dictionary der originalen Bilddaten und eines der assoziierten Labels), für eine automatische multi-label Segmentierung von Gehirntumoren, illustiert in Abbildung 1.9. Der Beitrag besteht aus einer neuen, voll-automatischen multi-label Segmentierung, welche gekoppelte Dictionaries verwendet, die aus einer Trainingsdatenmenge mit einer einzigen Bildmodalität (Flair MRI), assoziiert mit gelabelten Bildaten (Ground Truth Segmentierung), gelernt wurden. Aus der Trainingsdatenmenge der Bilder werden Patches extrahiert und zu einer Matrix in einem Dictionary zusammengefügt. Jeder Patch hat einen korrespondierenderen Patch in einem Label Dictionary. Das Label Dictionary repräsentiert die vier Labels des Vordergrunds (Nekrose, enhanced Tumor, non-enhanced Tumor und ödem) und ein Label für den Hintergrund. Für Tests benötigt die vorgestellte Methode Bilder der Modalität single MRI als Testdaten. Nachdem die Patches aus den Trainingsdaten extrahiert wurden, wird die Ähnlichkeit zwischen den Patches der Test-

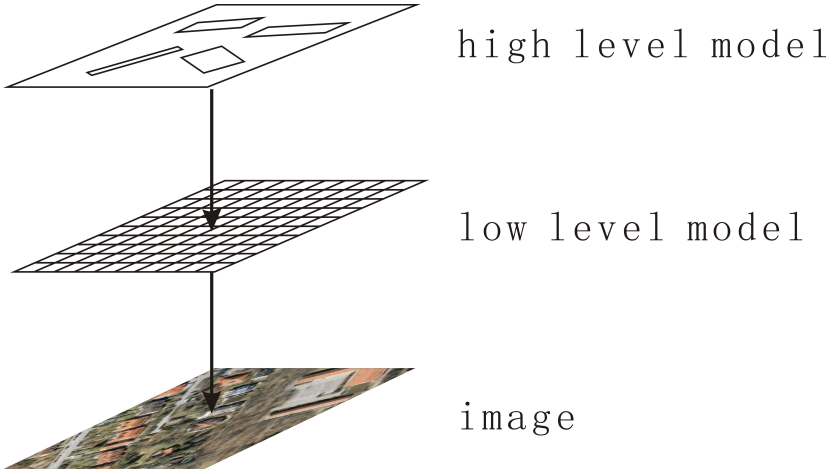


Figure 1.10: Hybride Repräsentation für die Extraktion von Gebäuden. Ein Marked-Point-Prozess wird für die Repräsentation von High-Level-Wissen verwendet, d.h. für die Gebäude und ihre Relationen. Ein Markov Random Field repräsentiert Low-Level-Informationen, d.h. die Eigenschaften aller Pixel. Jeder markierte Punkt auf dem High-Level steht für ein Gebäude und entspricht einer rechteckigen Region im Markov Random Field auf dem Low-Level.

daten und denen des Dictionaries der Bilder der Trainingsdaten bestimmt. Danach werden die korrespondierenden Teile in dem Label Dictionary ausgewählt. Das Label Dictionary wird verwendet, um die Vordergrund- und Hintergrundlabel für die Segmentierung mittels Graph-Cut zur Verfügung zu stellen.

Diese Arbeit wurde auf dem International Symposium on Visual Computing (ISVC) (Al-Shaikhli *et al.*, 2014b) präsentiert.

Fernerkundungsbildklassifikation

Der dritte Teil dieser Monographie besteht aus Kapiteln, wurde die Kopplung von Markovsche Zufallsfelder (MRF), Marked-Point Prozessen (MPP) und Gaußprozessen (GP) für die Fernerkundungsbildklassifikation beschreiben.

Kombination von MRFs und MPPs für die Extraktion von Gebäuden in Fernerkundungsbildern

Die automatische Extraktion von Gebäuden aus Fernerkundungsbildern ist ein aktives Forschungsfeld. Trotz der Forschungsbestrebungen der letzten Jahrzehnte ist eine voll automatische Extraktion schwierig. Das Hauptproblem ist die Art der Repräsentation von Objekten und Bildern (Mayer, 1999). Statistische Ansätze bieten einen mächtigen Rahmen für Modellierung und Inferenz. Markov Random Fields und Marked-Point-Prozesse können kontextbedingte Entitäten gut repräsentieren. Mittels MRFs können Low-Level-Informationen bezüglich einzelner Bild-Pixel und Interaktionen zwischen Pixeln effizient repräsentiert werden. Andererseits kann High-Level-Wissen, wie beispielsweise freie semantische Strukturen oder variable Topology,

nur schwer in MRFs integriert werden. Auf der Basis von Spatial-Point-Prozessen kann High-Level-Wissen eingespeißt werden, indem man Punkte und Punktverbindungen mit Markierungen verknüpft. Konkrete Formen können mit geometrischen Markierungen repräsentiert werden. Allgemeine Formen können aufgrund des Bildinhalts jedoch nicht bestimmt werden. Dies ist eine Folge der unzureichenden Repräsentation mittels Low-Level-Informationen.

Motiviert durch die komplementären Charakteristika von MRFs und MPPs kombinieren wir beide Ansätze. So können wir Low-Level-Informationen und High-Level-Wissen gleichermaßen repräsentieren. Auf Basis der kombinierten Repräsentation entwickeln wir einen Ansatz, welcher Gebäude aus einzelnen Fernerkundungsbildern extrahiert, siehe Abbildung 1.10. Diese Arbeit erschien im ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, ISPRS Congress (Chai *et al.*, 2012).

Multi-Source Multi-Scale Hierarchical Conditional Random Field für die Fernerkundungsbildklassifikation

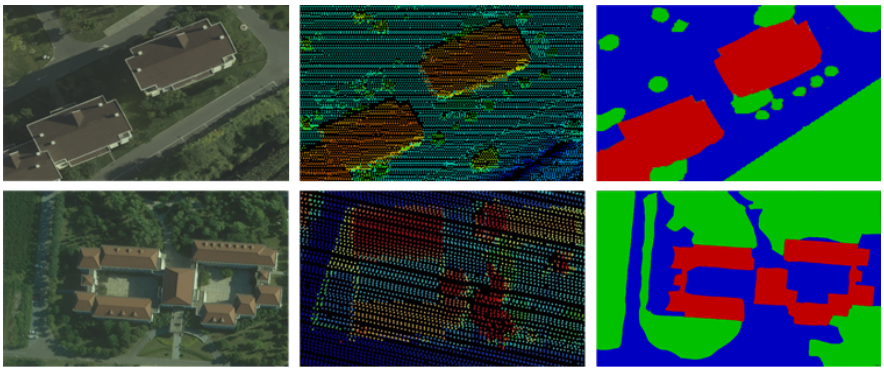


Figure 1.11: Ein Klassifikationsergebniss des MSMSH-CRF-Modells auf den Beijing-Airborne-Daten (Zhang *et al.*, 2013), *Links*: Fernerkundungsbild, *Mitte*: LiDAR-Punkt-Wolke, *Rechts*: Klassifikationsergebnis (rot - Gebäude, blau - Straße, grün - Vegetation).

Die Fusion von Fernerkundungsbildern mit LiDAR-Daten verbindet komplementäre Informationen für Fernerkundungsanwendungen wie Objektklassifizierung und -erkennung. Diese Arbeit präsentiert ein neues Multi-Source Multi-Scale Hierarchical Conditional Random Field (MSMSH-CRF), welches Features aus Fernerkundungsbildern und LiDAR-Punkt-Wolken für Bildklassifikation verbindet. Der Hauptbeitrag dieser Arbeit ist ein neues CRF-basiertes Modellierungsschema für komplementäre Multi-Source-Daten, wie beispielsweise die Textur von Fernerkundungsbildern und die Elevation in LiDAR-Daten.

Um verschiedene Ebenen von Kontextinformationen in Bildern zu nutzen, schlagen wir hierarchische Multi-Skalen-Potentiale vor, welche durch die Aggregation von Evidenz von der lokalen zur globalen Ebene erweitert werden. Zieht man die Kopplung des selben Objekts in den Fernerkundungsbildern und den LiDAR-Daten in Betracht, besteht der Nutzen der hierarchischen Multi-Source-Potentiale darin, vollen Nutzen aus der Kategorie-Konsistenz von

Multi-Source-Daten zu ziehen. Abbildung 1.11 zeigt exemplarische Ergebnisse der MSMSH-CRF-Klassifikation.

Diese Arbeit erscheint in den ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Photogrammetric Image Analysis (PIA) (Zhang *et al.*, 2015).

Verbindung von Gaußprozessen und Markov Random Fields für die hyperspektrale Bildklassifikation

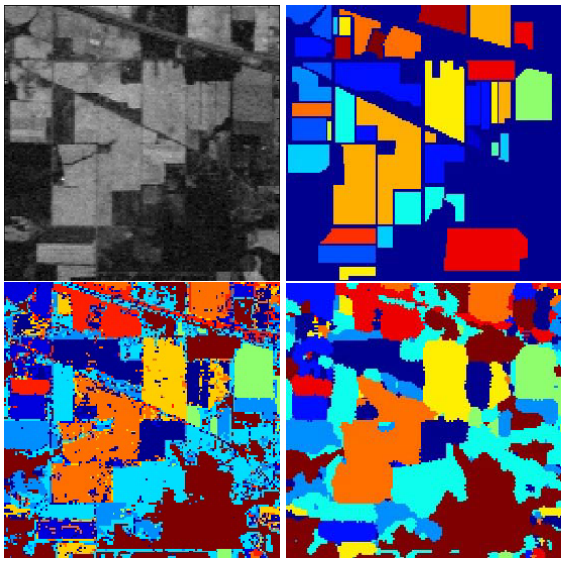


Figure 1.12: Hyperspektrale Bildklassifikationsergebnisse. *Oben links*: Daten von indischen Pinien, *Oben rechts*: Ground Truth, *Unten links*: Klassifikationsergebnis des GP, *Unten rechts*: Klassifikationsergebnis des GP-MRF.

Die ergiebigen Informationen von hyperspektralen Daten ermöglichen die Charakterisierung, Identifikation und Klassifikation der Erdoberfläche mit verbesserter Genauigkeit und Robustheit. Kernel-basierte Methoden in Form von SVMs haben sich bezüglich Genauigkeit und Robustheit als exzellenten Ansatz für HSI erwiesen (Camps-Valls & Bruzzone, 2005; Melgani & Lorenzo, 2004). Gaußprozesse (GPs) stellen einen weiteren Repräsentanten von potentiell vielversprechenden kernel-basierten Methoden dar. Allerdings haben Bayessche GPs bisher nicht viel Aufmerksamkeit in der Fernerkundungs-Community erhalten.

Diese Arbeit präsentiert ein Framework für ein GP-MRF, welches GPs und MRFs für die genaue Klassifikation von hyperspektralen Fernerkundungsbilddaten verbindet. Diese Methode nutzt die Beziehungen zwischen benachbarten Pixeln und integriert sie in die spektrale Information, um spektral-spatiale Klassifikation zu ermöglichen. Das Framework besteht aus zwei Schritten. Erstens: Ein GP sagt Klassen-Wahrscheinlichkeiten für jeden Pixel vorher. Zweitens: Ein MRF extrahiert räumliche Kontextinformationen aus der Labelvorhersage des ersten Schritts. Die Klassifikationsergebnisse werden dann aus den spektral-spatialen Informationen geschlossen. Durch Regularisierung des MRFs konnten verbesserte Klassifikation-

sergebenisse erzielt werden, siehe Abbildung 1.12. Die Arbeit erscheint im Rahmen des IEEE Joint Urban Remote Sensing Event (JURSE) (Liao *et al.*, 2015).

Chapter 2

Summary

This monograph studies the methods and applications of semantic labeling. In this chapter, we review the motivation of our work and our contributions in the interdisciplinary fields of computer vision, medical imaging and remote sensing. We argue in support of probabilistic models. Our contributions are illustrated with sample image results and cover three classes of application domains, each one being devoted a separate part of the monograph: (I) object segmentation, (II) medical image analysis, and (III) remote sensing image classification. The overview structure of this monograph is illustrated in Figure 2.1.

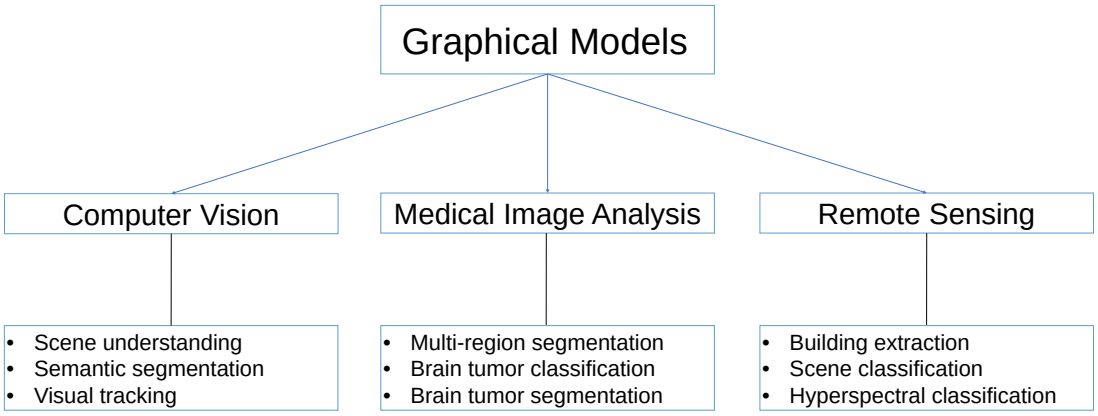


Figure 2.1: Overview structure of the thesis.

Motivation

While humans constantly extract meaningful information from visual data almost effortlessly, it turns out that simple visual tasks such as recognizing, detecting and tracking objects or understanding what is going on in the scene are extremely challenging problems for machines. To design artificial vision systems that can reliably process information as humans do has many potential applications in fields such as artificial intelligence, robotics, medical imaging, surveillance, remote sensing, entertainment or sports science. It is therefore our ultimate goal to be able to emulate the human visual system with computational algorithms.

A fundamental goal of computer vision is to discover the semantic information within a given scene, so called scene understanding. Scene understanding is the basis for many applications: surveillance, autonomous driving, traffic safety, robot navigation, vision-guided mobile navigation systems, or activity recognition. Understanding the scene in an image or video requires much more than recording and storing it and extracting some features. The overall goal is to find a mapping to derive semantic information from sensor data, which is an extremely challenging task partially due to the ambiguities in the appearance of the data. These ambiguities may arise either due to the physical conditions such as the illumination and the pose of the scene components with respect to the sensor, or due to the intrinsic nature of the data itself. Therefore, there is the need of capturing local, global or dynamic aspects of the acquired observations, which are to be utilized to understand what is occurring in a scene. All information which is possible to extract from a scene must be considered in context in order to get a comprehensive scene understanding, but this information, while it is easily captured by humans, is still difficult to obtain from machines.

Generally speaking, scene understanding can be formulated as a *labeling problem* that tries to assign a label to each unobserved hidden variable. The labels correspond to various estimation, such as an object class label in the case of object segmentation (Yang, 2015; Yang & Rosenhahn, 2014), a depth label in the case of depth upsampling (Huang *et al.*, 2015), a pixel intensity in the case of image denoising, or location and orientation in the case of relational tracking (Müller *et al.*, 2013). The labels are typically conditionally dependent on each other and the output labeling tends to be highly structured. Probabilistic graphical models provide a generic framework for statistical modeling, inference and learning in artificial vision systems. Markov random fields (MRFs) are the most commonly used graphical models in computer vision, which allow one to incorporate local contextual information in a principled manner. In MRFs, the dependencies among variables in a probability distribution are represented by edges connecting corresponding nodes in a graph. MRFs have been made popular in computer vision by the early works of Besag (1974, 1986); Geman & Geman (1984). Their limiting factor that they only allow for local features has been overcome by conditional random fields (CRFs) (Kumar & Hebert, 2003a; Lafferty *et al.*, 2001), where arbitrary features can be used for labeling, at the expense of a purely discriminative approach.

The task is to infer the most probable or maximum *a posteriori* (MAP) labelling of the random field, and can be found by minimizing the corresponding energy function (Yang, 2011). In general, minimizing the energy function is NP-hard. But, there exist a number of algorithms which compute the exact solution for a particular family of the energy functions in polynomial

time. For example, max-product belief propagation exactly minimizes the energy functions defined over the graphs with no loops (Yedidia *et al.*, 2000). However, many energy functions encountered in MRF and CRF models are NP-hard to minimize (Kolmogorov & Rother, 2007). Most multi-label energy functions are non-submodular. They are instead solved using the approximate algorithms. These algorithms belong to two categories: message passing algorithms, such as sum-product algorithm, belief propagation (Yedidia *et al.*, 2000), tree-reweighted message passing ((Kolmogorov, 2006; Wainwright *et al.*, 2005)), and move making algorithms, such as Iterated Conditional Modes (Besag, 1986), $\alpha\beta$-swap, and α-expansion (Boykov *et al.*, 2001).

The aim of this monograph is to studies the methods and applications of semantic labeling. We contribute to this active topic with the modeling and inference aspects of probabilistic models and their applications in the interdisciplinary fields of computer vision, medical imaging and remote sensing, which is shown in the following sections.

Contributions

Object Segmentation

The first part of this monograph consists of chapters that explore different probabilistic graphical models and their applications to the problems of image segmentation, object segmentation and depth upsampling, scene interpretation, video segmentation and tracking.

Image Segmentation by Bilayer Superpixel Grouping

Image segmentation is a fundamental low-level problem in computer vision and image processing. It provides the basis for high-level image understanding such as object recognition, image retrieval, activity recognition,etc.. Despite a variety of segmentation techniques have been proposed, it remains a challenging problem due to the broad diversity and ambiguity in an image. The task of segmentation is to group image pixels into visually meaningful objects, which are useful for further processing such as recognition. It has long been a challenging problem in computer vision and image processing. Approaches to image segmentation in the literature include normalized cuts (Shi & Malik, 2000), mean shift (Comaniciu & Meer, 2002), graph-based method (Felzenszwalb & Huttenlocher, 2004b), and ultrametric contour maps (Arbelaez *et al.*, 2011). We address the segmentation as a superpixel grouping problem. We propose a novel graph-based segmentation framework which is able to integrate different cues from bilayer superpixels simultaneously. The key idea is that segmentation is formulated as grouping a subset of superpixels that partitions a bilayer graph over superpixels, with graph edges encoding superpixel similarity. We first construct a bipartite graph incorporating superpixel cue and long-range cue. Furthermore, mid-range cue is also incorporated in a hybrid graph model. Segmentation is solved by spectral clustering. Our approach is fully automatic, bottom-up, and unsupervised. The work appeared at the Asian Conference on Pattern Recognition (ACPR) (Yang, 2013).

Estimating Layout of Cluttered Indoor Scenes Using Trajectory-based Priors

Estimating layout or structure of an indoor scene is important for many tasks, such as activity analysis (McKenna & Charif, 2004), robot navigation (Thrun *et al.*, 2004), scene understanding (Saleemi *et al.*, 2010) or object placement (Jia *et al.*, 2013). State of the art methods either use spatial image features (Hedau *et al.*, 2009) or use trajectory based temporal information (Zhang *et al.*, 2011) for this purpose. However, either of the features are not enough to estimate the indoor scene layout. A major challenge for image features based techniques arises from the fact that most indoor scenes are cluttered by a lot of furniture and decorations (Wang *et al.*, 2013). They often obscure the geometric structure of the scene, and also occlude boundaries between walls and the floor. Appearances and layouts of clutters can vary drastically across different indoor scenes, so it is extremely difficult to model them consistently. Similarly trajectory based techniques normally cluster the trajectory data and model only the paths (Zhang *et al.*, 2011). They do not take care of the clutter or resting places in the scene.

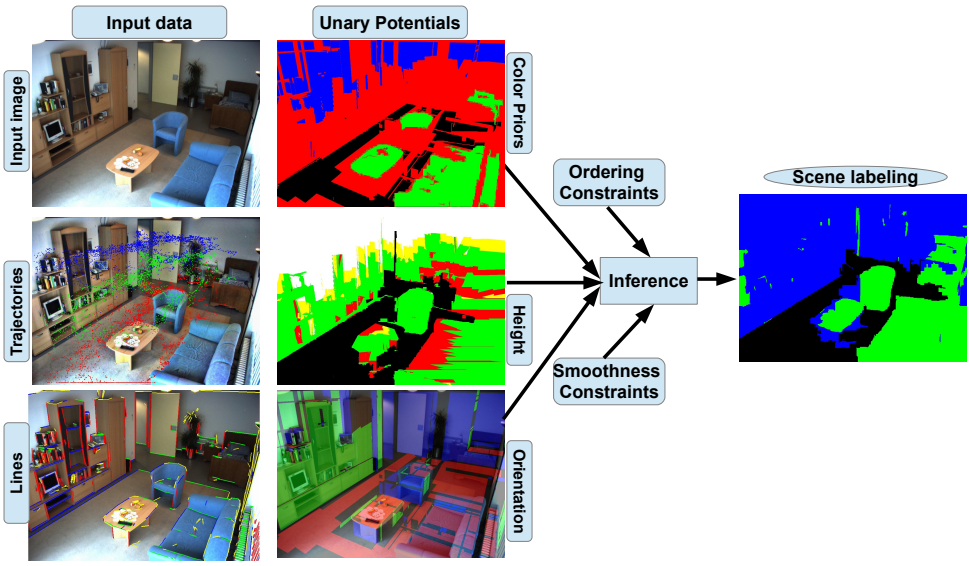


Figure 2.2: Unsupervised scene segmentation procedure for inactivity zones using CRF. First column shows the input data including image of the surveillance scene, key-point trajectories and lines in the scene image. Second column shows the three features used to define unary class potentials, i.e. color priors, relative quantized height and surface orientations. Binary potentials, i.e. ordering and smoothness constraints are added to define homogeneous regions neighborhood relationship. Inference procedure finds the optimal scene segmentation by minimizing energy on CRF.

Though image features and trajectory data are not self sufficient for reliable scene layout estimation but they can be used together to achieve reliable indoor scene layout estimation. We propose a mechanism which learns the scene semantic context model using image segmentation mechanism in an unsupervised way. We do not use trajectories directly for scene layout estimation rather our segmentation mechanism used features both image and trajectory based

features. We are also able to model the resting or sitting places in the scene. We assume that we have a static and uncalibrated surveillance camera in the scene. Using pixel-level color and perspective cues of the scene, each pixel is assigned to a particular class either a sitting place, the ground floor, or static background areas like walls and ceiling. The global topological order of classes, such as sitting objects and background areas are above ground, is locally integrated into a conditional random field (CRF) by an ordering constraint. Figure 2.2 gives an overview of the CRF-based image segmentation for unsupervised scene layout estimation procedure. A graph cut based inference algorithm is run on our CRF to define the final scene segmentation or layout. The proposed method yields very accurate segmentation results on challenging real world scenes. The work appeared in Image Vision Computing (IVC) (Shoaib *et al.*, 2014).

Joint Object Segmentation and Depth Upsampling

In recent years, the conjunctive use of ranging sensors and cameras has become more and more popular, which benefits many computer vision applications. This work focuses on depth assisted object-level image segmentation and image guided depth upsampling. The former takes advantage of depth information to segment an image into regions that correspond to objects. Previous works on this problem mainly rely upon depth maps inferred from dense stereo vision (Ladickỳ *et al.*, 2012; Sengupta *et al.*, 2013). Recently, sparse 3D point clouds and the reconstructed corresponding dense depth maps are exploited as well in semantic segmentation for road scenarios (Chen *et al.*, 2014; Huang *et al.*, 2014). In these works, depth information is integrated either as geometric priors or as hard constraints within a Markov random field (MRF) framework to improve segmentation performance. The latter problem, aiming to generate high-resolution depth maps from sparse measurements, takes high-resolution visual images as guidance. Existing researches mainly use techniques such as bilateral filtering (Yang *et al.*, 2007), sparse representation (Gong *et al.*, 2014), or MRF (Diebel & Thrun, 2005; Zhu *et al.*, 2010). However, a common weakness shared by both is that they suffer from errors existing in their guidance. More specifically, the performance of image segmentation will be degenerated if the used depth map is noisy or overly smoothed on edges. Likewise, a segment that crosses over object boundaries may lead to wrong depth upsampling results. In order to prevent from such error propagation, we propose to solve these two problems jointly.

We propose a joint method to perform both depth assisted object-level image segmentation and image guided depth upsampling. To this end, we formulate these two tasks together as a bi-task labeling problem, defined in a Markov random field (MRF). An alternating direction method is adopted for the joint inference, solving each sub-problem alternatively. More specifically, the sub-problem of image segmentation is solved by Graph Cuts, which attains discrete object labels efficiently. Depth upsampling is addressed via solving a linear system that recovers continuous depth values. By this joint scheme, robust object segmentation results and high-quality dense depth maps are achieved, as shown in Figure 2.3. The work appears in IEEE Signal Processing Letters (SPL) (Huang *et al.*, 2015).

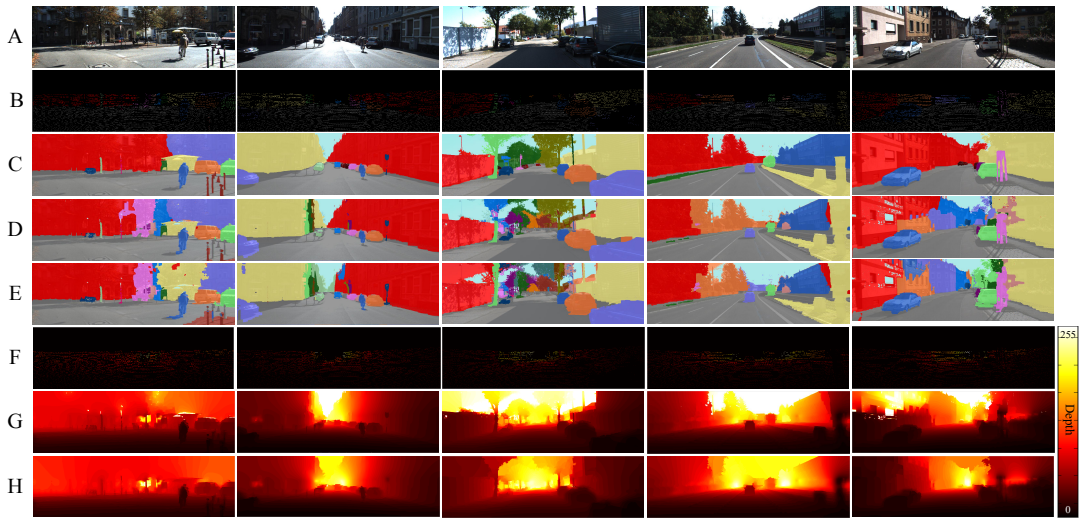
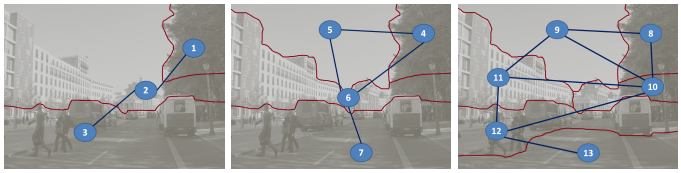


Figure 2.3: Object-level image segmentation and depth upsampling results on the KITTI dataset. (A) Original Image. (B) Object seeds. (C) Object-level image segmentation ground truth. (D) Joint segmentation result. (E) Stand-alone segmentation result. (F) Sparse depth map. (G) Joint depth upsampling result. (H) Stand-alone depth upsampling result.

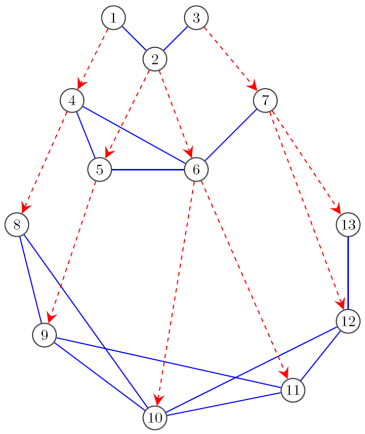
A Generic Probabilistic Graphical Model for Region-based Scene Interpretation

The task of semantic scene interpretation is to label the regions of an image and their relations into semantically meaningful classes. Such task is a key ingredient to many computer vision applications, including object recognition, 3D reconstruction and robotic perception. The problem of scene interpretation in terms of classifying various image components in the images is a challenging task partially due to the ambiguities in the appearance of the image data (Tsotsos, 1988). These ambiguities may arise either due to the physical conditions such as the illumination and the pose of the scene components with respect to the camera, or due to the intrinsic nature of the data itself. Images of man-made scenes, e. g. building facade images, exhibit strong contextual dependencies in the form of spatial and hierarchical interactions among the components. Neighboring pixels tend to have similar class labels, and different regions appear in restricted spatial configurations. Modeling these spatial and hierarchical structures is crucial to achieve good classification accuracy, and help alleviate the ambiguities.

Graphical models, either directed models or undirected models, provide consistent frameworks for the statistical modeling. Two types of graphical models are frequently used for capturing such contextual information, i. e. Bayesian networks (BNs) (Sarkar & Boyer, 1993) and random fields (RFs) (Besag, 1974), corresponding to directed and undirected graphs. RFs mainly capture the mutually dependent relationships such as the spatial correlation. On the other side, BNs usually model the causal relationships among random variables. Both have been used to solve computer vision problems, yet they have their own limitations in representing the relationships between random variables. BNs are not suitable to represent symmetric relationships that mutually relate random variables. RFs are natural methods to model symmetric relationships, but they are not suitable to model causal or part-of relationships. Our key



(a) Multi-scale segmentation



(b) The graphical model

Figure 2.4: Illustration of the graphical model architecture. The blue edges between the nodes represent the neighborhoods at one scale (undirected edges), and the red dashed edges represent the hierarchical relation between regions (undirected or directed edges).

contribution is the development of a generic statistical graphical model for scene interpretation, as illustrated in Figure 2.4, which seamlessly integrates different types of the image features, and the spatial structural information and the hierarchical structural information defined over the multi-scale image segmentation. It unifies the ideas of existing approaches, e. g. conditional random field and Bayesian network, which has a clear statistical interpretation as the MAP estimate of a multi-class labeling problem. The work appears at the International Conference on Computer Vision Theory and Applications (VISAPP) (Yang, 2015).

Video Segmentation with Joint Object and Trajectory Labeling

Unsupervised video object segmentation is a challenging problem in computer vision because it involves a large amount of data and object appearance may significantly change over time. Object segmentation is the basis for many potential applications including object tracking, object recognition, 3D reconstruction, activity recognition, and video retrieval. Due to its potential applications, there is increasing number of works (Grundmann *et al.*, 2010; Lee *et al.*, 2011) addressing the problem of video object segmentation in recent years. Many approaches extend single image segmentation techniques to multiple frames, exploiting the fact that there is redundancy along the time axis and that the motion field is smooth. The problems associated

with these methods include drift, occlusion, and appearance adaption. Integrating long-term cues in the segmentation process might help solve these problems. In fact, video provides rich additional cues beyond a single image. These cues include object motion, temporal continuity, and long-range temporal object interactions, etc. Motion segmentation exploits these cues, which formulates clustering objectives to group pixels from all frames. However, motion segmentation results are only in discrete and sparse positions available (Brox & Malik, 2010).

We overcome aforementioned problems by merging image segmentation and motion segmentation. We propose a method to obtain a spatio-temporal foreground segmentation of a video that respects object boundaries, as shown in Figure 2.5, and at the same time perform trajectory labeling. Different from previous approaches, we address the foreground segmen-

Figure 2.5: Video object segmentation. Input: unannotated video. Output: Foreground object in each frame.

tation by partitioning frames using a novel graphical model on pixel level, which is dense in spatial domain, yet sparse in temporal domain. The main scientific contribution is a fully automatic and unsupervised bottom-up approach for the combination of object segmentation and motion segmentation, which is formulated as inference in a unified CRF model. The CRF contains pixel labeling and trajectory clustering in a single energy function, which integrates dense local interaction and sparse global constraints. We optimize over pixels and trajectories in the joint space via a space-time CRF: both foreground estimation and trajectory clustering are modeled as energy potentials. An optimization scheme based on a coordinate ascent style procedure is proposed to solve the inference problem. To the best of our knowledge, this work is the first one to combine object labeling and trajectory clustering in a unified probabilistic framework. The work appeared at the IEEE Winter Conference on Applications of Computer Vision (WACV) (Yang & Rosenhahn, 2014).

Slice Sampling Particle Belief Propagation

Markov Random Fields (MRFs) are a powerful tool for modeling relational dependencies among observations. Inference in such models is an inherent problem which has been widely addressed in the past. MRFs, and hence its inference methods, can be classified in two categories: dis-

cretely and continuously labeled problems. Most works on MRF optimization specialize on a discrete label space (Boykov *et al.*, 2001; Kolmogorov, 2006). Often such approaches are hard to apply on tasks where a continuous label space would be a more natural choice, such as feature tracking with relational constraints (Salzmann & Urtasun, 2012). Recently, message passing approaches working in continuous rather than discrete label space were proposed (Ihler & McAllester, 2009; Peng *et al.*, 2011; Sudderth *et al.*, 2010). These approaches use MCMC methods to approximate the message distributions. All previously proposed MCMC based belief propagation methods use Metropolis-Hastings (MH) sampling. This sampling strategy consists of two steps: (a) sampling a candidate particle from an easy to sample *proposal distribution*, and (b) accept or reject the candidate depending on a transition probability. Applying this sampling technique involves a careful design of the proposal distribution, which is a compromise between exploring the label space (using a broad proposal distribution) and maximizing the transition acceptance ratio (minimize sample moves) at the same time. A schematic overview of the PBP framework is shown in Figure 2.6.

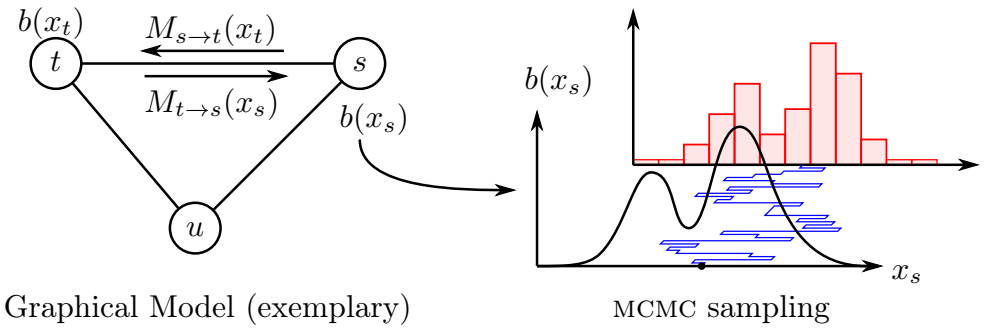


Figure 2.6: Particle Belief Propagation framework. Left: Message passing mechanism. Right: MCMC particle sampling of the belief $b(x_s)$ with an exemplary MCMC sampling chain of one particle (blue) and its corresponding histogram (red).

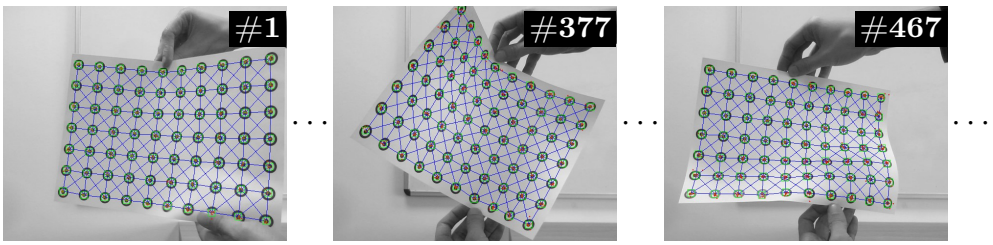


Figure 2.7: Relational 2D feature tracking example.

The main scientific contribution is a novel particle belief propagation algorithm using slice sampling (S-PBP) instead of Metropolis-Hastings. This method exploits the structure of the PBP message passing equations for direct sampling from the target distribution and does not depend on a proposal distribution which is difficult to tune, provided the unary and binary potentials

are defined by analytic functions or can be bounded by one. Our findings are verified on a complex 2D relational feature tracking application as shown in Figure 2.7. The work appeared at the IEEE International Conference on Computer Vision (ICCV) (Müller *et al.*, 2013).

Medical Image Analysis

The second part of this monograph consists of chapters that describe both graph prior and dictionary learning approaches for medical image analysis, with applications to semantic labeling and segmentation, tumor classification and segmentation.

Multi-Region Labeling and Segmentation Using a Graph Topology Prior and Atlas Information in Brain Images

Medical image segmentation and anatomical structure labeling according to the types of the tissues is important for accurate diagnosis and therapy. We propose a novel approach for multi-region labeling and segmentation, which is based on a topological graph prior, registration of the labels, and the topological information of an atlas, using a multi-level set energy minimization method. We consider topological graph prior and atlas information to evolve the contour based on a topological relationship presented via a graph relation. This method is capable of segmenting adjacent objects with very close gray level that would be difficult to segment correctly using standard methods. The topological graph is registered from the low resolution and noisy source image to the topological information of an atlas to obtain region labeling. We present the graph prior and label registration techniques to explain how it gives precise multi-region segmentation and labeling. The proposed algorithm is capable of segmenting and labeling different regions in noisy or low resolution brain MRI images of different modalities, as shown in Figure 2.8. The work appeared in Computerized Medical Imaging and Graphics Journal (CMIG) (Al-Shaikhli *et al.*, 2014c).

Brain Tumor Classification Using Sparse Coding and Dictionary Learning

Early identification of brain tumors is important to treat the tumors effectively. Multi-class brain tumor classification is considered as one of the most important and challenging tasks in medical imaging due to the difficulty to extract the relevant information that can help to discriminate the tumor from the normal brain tissue. The contribution is a modified sparse coding and dictionary learning based multi-class classification. We proposed to use the K-SVD method to update both of the dictionary and sparse coding steps. Furthermore, due to the high degree of similarity in pixel intensities between normal brain tissue and tumor, and the variability of the tumor shape, location, and size, this variability justifies the use of topological and texture features to learn the dictionary. The topological feature gives information whether the case is normal or abnormal based on the assumption that the topology of normal brain is fixed. Therefore, the presence of tumor in the brain will change the normal brain topology. In addition, the texture features provide a good discrimination of the brain tumor types. The main novelty in our algorithm is the use of topology and texture features for learning, instead of

applying learning directly on pixel values. The work was presented at the IEEE International Conference on Image Processing (ICIP) (Al-Shaikhli *et al.*, 2014a).

Coupled Dictionary Learning for Automatic Multi-Label Brain Tumor Segmentation in Flair MRI images

We propose a novel coupled dictionary learning approach (one dictionary of the original image data and one of the associated label image data) of automatic multi-label brain tumor segmentation, as shown in Figure 2.9. The contribution is a novel fully automatic algorithm for multi-label segmentation using coupled dictionaries learned from single modality (Flair MRI modality) image training data with associated label image data (ground truth segmentation). Patches are extracted from the training image data and concatenated to a matrix in a dictionary. Each patch has its corresponding patch in a label dictionary. The label dictionary represents four foreground labels (necrosis, enhanced tumor, non-enhanced tumor, and edema) and one background label. For testing, the proposed method requires single MRI modality input of the testing data. After extracting the patches from the test image data, the patch similarity is retrieved between the patches of the testing data and these in the dictionary of the training image data, then the corresponding atoms in the label dictionary are selected. The label dictionary is used to provide the foreground and background labels for graph-cut segmentation. The work was presented at the International Symposium on Visual Computing (ISVC) (Al-Shaikhli *et al.*,

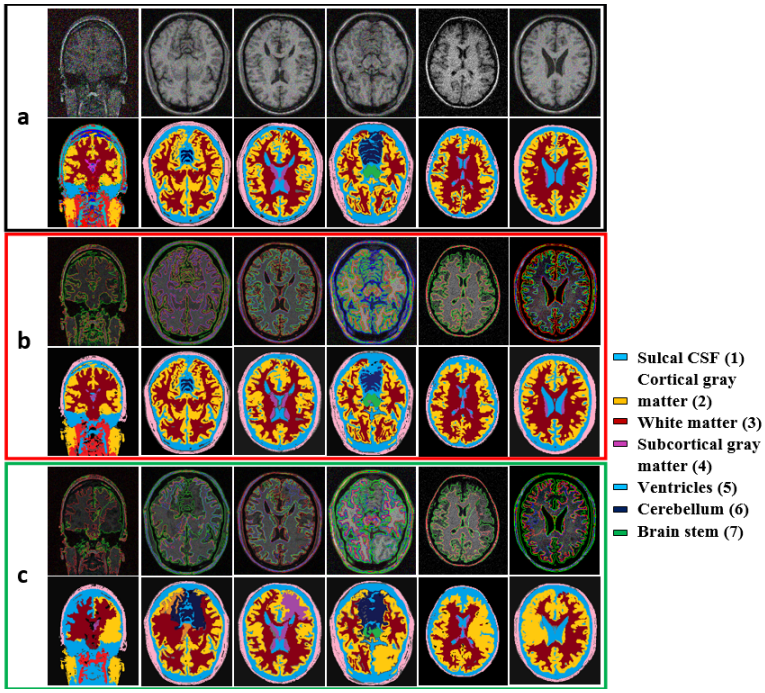


Figure 2.8: Examples of a multi-region labeling and segmentation. (a) are the input images and the ground truth, (b) are the segmentation results with topological graph and atlas information, (c) are the segmentation results with topological graph without atlas information.

2014b).

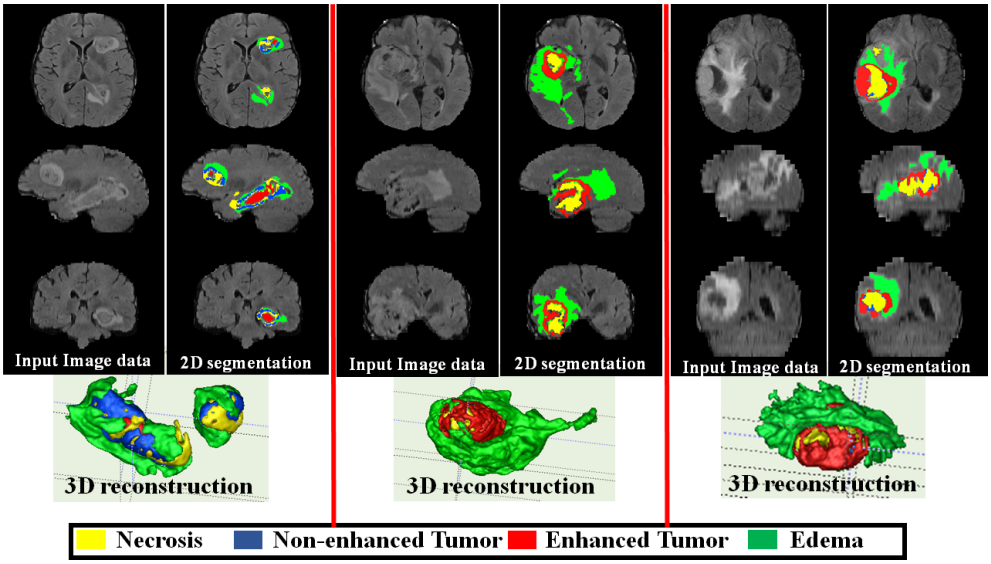


Figure 2.9: 3D Multi-label glioma segmentation of BraTS testing data. In each example, the first column is the input image data (axial, coronal, and sagittal planes), the second column is the 2D tumor segmentation, the fourth row is 3D tumor reconstruction.

Remote Sensing Image Classification

The third part of this monograph consists of chapters that discuss the integration of marked point process, Gaussian process and Markov random field for remote sensing image classification.

Combine Markov Random Fields and Marked Point Processes to Extract Building from Remotely Sensed Images

Automatic building extraction from remotely sensed images is a popular research topic. In spite of the research efforts of the past decades fully automatic extraction is still a challenging task. The key issue is representation of objects and images (Mayer, 1999). Statistical approaches provide a strong framework of modeling and estimation. Markov random fields and marked point processes represent context-dependent entities well. Based on Markov random fields (MRFs), low-level information referring to the single image pixels and interaction between neighboring pixels are represented concisely. However, high-level knowledge, such as free semantic structures and variable topology, can not be represented by MRFs conveniently. Based on spatial point process, high-level knowledge can be introduced via marks attached to the points and the relationships between neighboring points. While specific shapes can be represented by geometric marks, general shape can not be determined based on image content. This problem results from the weakness of representing low-level information.

Motivated by the complementary characteristics of MRFs and marked point processes, we combine them to represent both low-level information and high-level knowledge. Based on this representation, we propose an automatic approach for extracting buildings from single remotely sensed image, as illustrated in Figure 2.10. At high level, rectangles are used to represent buildings, and a marked point process is constructed to represent the buildings on ground scene. Interactions between buildings are introduced into the the model to represent their relationships. At the low level, a MRF is used to represent the statistics of the image appearance. Histograms of colors are adopted to represent the building's appearance. The high-level model and the low-level model are combined by establishing correspondences between marked points and nodes of the MRF. We adopt reversible jump Markov Chain Monte Carlo (rjMCMC) techniques to explore the configuration space at the high level, and adopt a Graph Cut algorithm to optimize configuration at the low level. We propose a top-down schema to use results from high level to guide the optimization at low level, and propose a bottom-up schema to use results from low level to drive the sampling at high level. The work appeared at the ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, ISPRS Congress (Chai *et al.*, 2012).

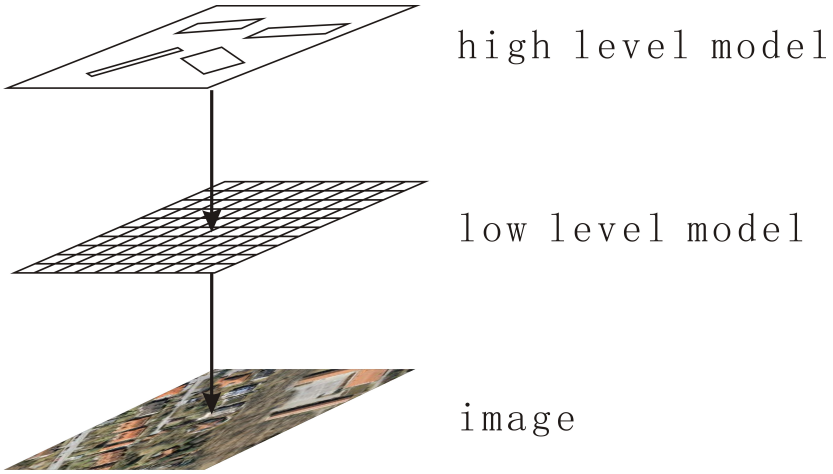


Figure 2.10: Hybrid representation for building extraction. Marked point process is adopted to represent the high-level knowledge, i.e. the buildings and their distribution. Markov random field is adopted to represent the low-level information, i.e. the properties of all pixels. Each marked point at the high level denotes one building, and it corresponds to one rectangular region in the Markov random field at the low level. The high-level model and the low-level model are combined together by establishing correspondences of marked points at the high level and regions at the low level. High-level knowledge is introduced as a priori term in the MRF and low-level information is introduced into data term in the marked point process.

Multi-Source Multi-Scale Hierarchical Conditional Random Field Model for Remote Sensing Image Classification

Fusion of remote sensing images and LiDAR data provides complimentary information for the remote sensing applications, such as object classification and recognition. This work presents a novel multi-source multi-scale hierarchical conditional random field (MSMSH-CRF) model to integrate features extracted from remote sensing images and LiDAR point cloud data for image classification. The main contribution of this work is a novel CRF-based modeling scheme exploiting the complementarity of multi-source data such as the texture in remote sensing images and the elevation in LiDAR data. To exploit different levels of contextual information in images, the multi-scale hierarchical potentials are proposed in our model, which is then enhanced by evidence aggregation from a local to global level. Considering the interrelation of the same objects in remote sensing images and LiDAR data, multi-source hierarchical potentials are proposed in the model to make full use of the category consistency of multi-source data. Figure 2.11 shows the example results of MSMSH-CRF classification method. The work appears at the ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Photogrammetric Image Analysis (PIA) (Zhang *et al.*, 2015).

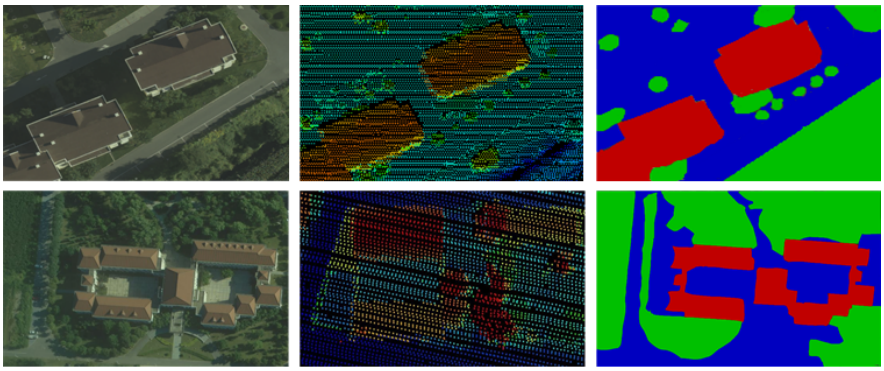


Figure 2.11: The classification result from the MSMSH-CRF model on the Beijing Airborne Data (Zhang *et al.*, 2013), *Left*: remote sensing image, *Middle*: LiDAR point cloud, *Right*: classification result (red - building, blue - road, green - vegetation).

Integration of Gaussian Process and Markov Random Field for Hyperspectral Image Classification

The abundant spectral information contained in hyperpsectral data enable the characterization, identification, and classification of the land-covers with improved accuracy and robustness. The kernel-based methods represented by SVMs have been proved as an excellent classification approach for HSI in terms of accuracy and robustness (Camps-Valls & Bruzzone, 2005; Melgani & Lorenzo, 2004). Gaussian processes (GPs) are another representative of potentially promising kernel-based methods. However, Bayesian GPs have not received much attention in remote sensing community.

This work presents a framework GP-MRF, which combines GPs and MRFs for accurate classification of hyperspectral remote sensing image data. This method exploits the relationship between adjacent pixels and integrates it into spectral information to obtain spectral-spatial classification. This framework consists of two steps. Firstly, a GP classifier yields pixelwise predictive probability for each class. Secondly, an MRF is applied to extract spatial contextual information in the label map from the first step. Then the classification results are inferred from the spectral-spatial information. By means of MRF regularization an enhanced classification result has been obtained, as illustrated in Figure 2.12. The work appears at the IEEE Joint Urban Remote Sensing Event (JURSE) (Liao *et al.*, 2015).

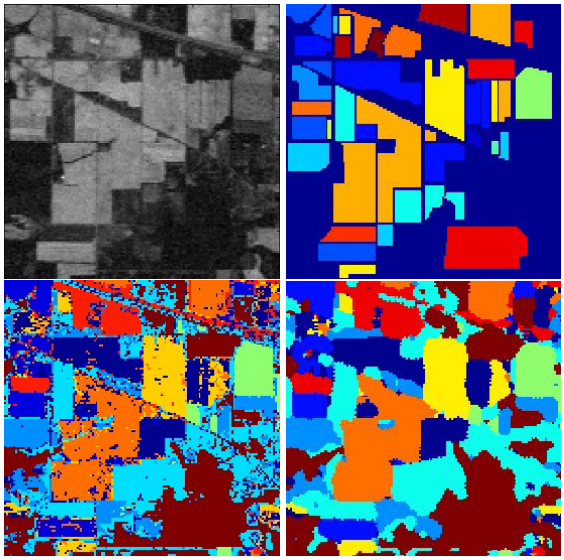


Figure 2.12: Hyperspectral image classification results. *Top Left*: Data of Indian Pines, *Top Right*: ground truth, *Bottom Left*: classification result of GP, *Bottom Right*: classification result of GP-MRF.

2. SUMMARY

Part I

OBJECT SEGMENTATION

Chapter 3

Image Segmentation by Bilayer Superpixel Grouping

The task of image segmentation is to group image pixels into visually meaningful objects. It has long been a challenging problem in computer vision and image processing. In this chapter we address the segmentation as a superpixel grouping problem. We propose a novel graph-based segmentation framework which is able to integrate different cues from bilayer superpixels simultaneously. The key idea is that segmentation is formulated as grouping a subset of superpixels that partitions a bilayer graph over superpixels, with graph edges encoding superpixel similarity. We first construct a bipartite graph incorporating superpixel cue and long-range cue. Furthermore, mid-range cue is also incorporated in a hybrid graph model. Segmentation is solved by spectral clustering. Our approach is fully automatic, bottom-up, and unsupervised. We evaluate our proposed framework by comparing it to other generic segmentation approaches on the state-of-the-art benchmark database. An earlier version of this chapter appeared at the Asian Conference on Pattern Recognition (ACPR) (Yang, 2013).

3.1 Introduction

Image segmentation is a fundamental low-level problem in computer vision and image processing. It provides the basis for high-level image understanding such as object recognition, image retrieval, activity recognition,etc.. Despite a variety of segmentation techniques have been proposed, it remains a challenging problem due to the broad diversity and ambiguity in an image. The task of segmentation is to group image pixels into visually meaningful objects, which are useful for further processing such as recognition.

In image segmentation, one has to consider a prohibitive number of possible pixel groupings. Using prior information about object appearance, or other scene content significantly simplifies the problem. For instance, many segmentation techniques are formulated as a Markov random field based energy minimization problems. However, the corresponding energy functions typically include terms that require prior object knowledge in terms of user interaction (Rother *et al.*, 2004) or knowledge about object appearance. Approaches to image segmentation in the literature include normalized cuts (Ncut) (Shi & Malik, 2000), mean shift (MS)

(Comaniciu & Meer, 2002), graph-based method (FH) (Felzenszwalb & Huttenlocher, 2004b), and ultrametric contour maps (UCM) (Arbelaez *et al.*, 2011). In recent years an increasingly popular way to solve image segmentation problem uses superpixels (Achanta *et al.*, 2012). This allows features to be computed over a larger spatial support. In most cases, they are used to initialize segmentation. Endres & Hoiem (2010) generated multiple proposals by varying the parameters of a conditional random field built over a superpixel graph. We think of segmentation as a bottom-up preprocessing step for high-level computer vision tasks such as indexing and recognition, providing substantial reduction in the computational complexity of these tasks. It is therefore unclear how segmentation methods that use strong prior knowledge are applicable for object recognition from large databases.

In this work we address the image segmentation as a superpixel grouping problem, based on the observation that object boundaries are often reasonably well approximated by superpixel boundaries. We propose a novel graph-based segmentation framework which is able to integrate cues from bilayer superpixels simultaneously. Our approach is fully automatic, bottom-up, and unsupervised. The key idea is that segmentation is formulated as grouping a subset of superpixels that partitions a bilayer graph over superpixels, with graph edges encoding superpixel similarity. We first construct a bipartite graph incorporating superpixel cue and long-range cue (neighboring superpixels in two layer). Segmentation is solved by spectral clustering. Furthermore, mid-range cue (neighboring superpixels within one layer) is also incorporated in a hybrid graph model. Given an image, we first compute two layer superpixel segmentation of the image. Based on two superpixel images, segmentation is performed as a superpixel grouping problem.

3.2 Problem Formulation

In this section, we propose a novel graph-based segmentation framework which is able to integrate different cues from bilayer superpixels simultaneously. We formulate segmentation as a superpixel grouping problem, based on the observation that object boundaries are often reasonably well approximated by superpixel boundaries. A bipartite graph is constructed to incorporate superpixel cue and long-range cue. Segmentation is then solved using spectral clustering. Furthermore, we propose a hybrid graph model that incorporates superpixel cue, mid-range cue, and long-range cue.

3.2.1 Bipartite Graph Construction

We construct a bipartite graph over two layer superpixels of one image I, as shown in Figure 3.1. Superpixels are generated by some unsupervised segmentation algorithms, such as NCut (Shi & Malik, 2000), UCM (Arbelaez *et al.*, 2011), SLIC (Achanta *et al.*, 2012), etc.. Formally, let $G_b = (U, V, E_{UV})$ be a bipartite graph with node set $U \cup V$ corresponding to two layers of superpixels and E_{UV} corresponding to graph edges between two layers, where $U = \{u_i\}_{i=1}^{n}$ and $V = \{v_j\}_{j=1}^{m}$. Given the above bipartite graph $G_b = (U, V, E_{UV})$, the task is to partition it into k groups. We further define an edge weight w_{ij} to encode the similarity between two superpixels u_i and v_j in two layers that are connected by an edge. The weight

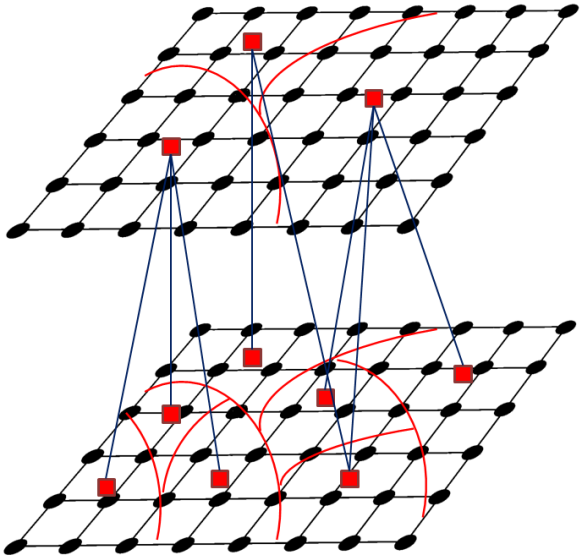


Figure 3.1: The proposed bipartite graph model of image segmentation. A black dot denotes a pixel while a red square denotes a superpixel.

matrix $W = (w_{ij})_{n\times m}$ is constructed as follows, which could also be seen as an across-affinity matrix between U and V,

$$w_{ij} = \begin{cases} \alpha & \text{if } |u_i \cap v_j| = \min(|u_i|, |v_j|) \\ e^{-\beta d_{ij}} & \text{if } u_i \sim v_j \\ 0 & \text{otherwise} \end{cases} \tag{3.1}$$

where $|u_i \cap v_j|$ is the number of pixels in the intersection between superpixels u_i, v_j, d_{ij} denotes the distance [1] between the features of superpixels u_i and v_j, $\sim$ denotes a certain neighborhood between superpixels [2], and α, β are free parameters controlling the balance between the superpixel cue and the long-range cue, respectively. By this construction, two neighboring superpixels are more likely to be grouped together if they are closer in feature space.

In Bagon *et al.* (2008), the easiness and difficulty of describing one superpixel u_i is evaluated by its description length in terms of visual codewords. Inspired by Bagon *et al.* (2008), we define the distance as the Kullback-Leibler divergence between two superpixels u_i and v_j. Specifically, given a dictionary of visual codewords, and the histogram of occurrence of the codewords in u_i, we define

$$d_{ij} = -\log \mathrm{KL}(u_i, v_j) \tag{3.2}$$

where KL denotes the Kullback-Leibler divergence. Below, we explain how to extract the

[1] For example, if color space is used as the feature space, and a superpixel u_i (v_j) is represented by the average color c_i (c_j) of the pixels within it, then $d_{ij} = \|c_i - c_j\|_2$.

[2] For example, $u_i \sim v_j$, if u_i is spatially adjacent to v_j or most similar to v_j in terms of (average) color. This neighborhood relationship is adopted in this work.

dictionary of codewords. First, SIFT descriptors (Lowe, 2004) are extracted for each pixel of the superpixel at a fixed scale and orientation, using the fast SIFT framework in Vedaldi & Fulkerson (2008). The pixel descriptors are then clustered using K-means (with $K = 100$). All pixels grouped within one cluster are labeled with a unique codeword id of that cluster. Then, the histogram of their occurrence in every superpixel is estimated.

3.2.2 Superpixel Spectral Clustering

To make spectral clustering method applicable to our problem, we can simply denote an expanded similarity matrix

$$S = \begin{bmatrix} O & W \\ W^T & O \end{bmatrix} \tag{3.3}$$

where W is the across-affinity matrix of the bipartite graph G_b. Note that this similarity matrix is sparse and symmetric. We denote by

$$L = I_{n+m} - H^{-1/2} S H^{-1/2} \tag{3.4}$$

the Laplacian matrix, where I_{n+m} is identity matrix and H the diagonal matrix composed of the row sums of S (Shi & Malik, 2000). It can be easily shown that for any S with nonnegative elements, the Laplacian matrix is symmetric positive semi-definite. Spectral clustering captures essential cluster structure of a graph using the spectrum of graph Laplacian matrix. Mathematically, it solves the generalized eigen-problem (Shi & Malik, 2000):

$$L\nu = \gamma H \nu \tag{3.5}$$

where γ and ν are corresponding eigen-values and eigen-vectors. The first k generalized eigenvectors $r_1, \cdots, r_k$ of the generalized eigen-problem Eq. (3.5) are computed by Lanczos method (Golub & Loan, 1996), where k is the cluster number. Let $R \in \Re^{(n+m)\times k}$ be the matrix containing the vectors $r_1, \cdots, r_k$ as columns. The $n + m$ rows of R can thus be easily clustered by k-means (Ng *et al.*, 2001) or certain discretization technique (Yu & Shi, 2003).

3.2.3 Hybrid Graph Model

In the above graph construction, our graph model incorporates both long-range grouping cues by bilayer graph construction and short-range superpixel cues by superpixel representation. In addition, mid-range smoothing cues could naturally be incorporated in this graph model, which we call hybrid graph model. let $G = (U, V, E_{UV}, E_U, E_V)$ be a expanded general graph from the bipartite graph G_b with E_U corresponding to graph edges within one layer of U and E_V corresponding to graph edges within the layer of V, as shown in Figure 3.2.

We define an edge weight $p_{ii'}$ ($q_{jj'}$) to encode the similarity between two spatially adjacent superpixels u_i (v_j) and $u_{i'}$ ($v_{j'}$) that are connected by an edge as follows

$$p_{ii'} = \text{TD}(u_i, u_{i'}) \tag{3.6}$$

where $\text{TD}(u_i, u_{i'}) = \|t_i - t_{i'}\|_1$. t_i is the histogram of Texton occurrence of superpixel u_i.

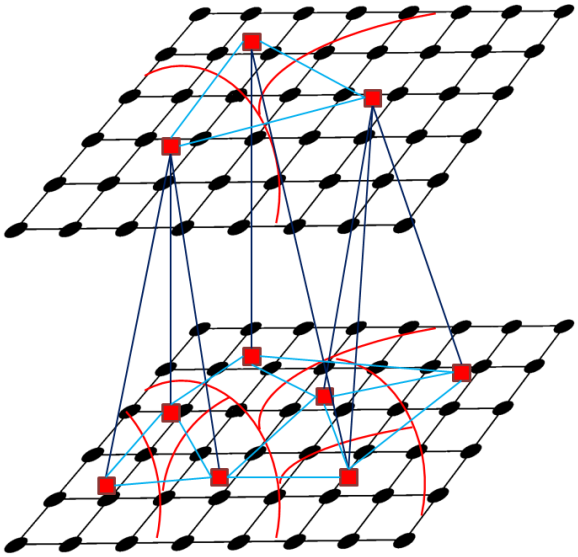


Figure 3.2: The proposed hybrid graph model of image segmentation. A black dot denotes a pixel while a red square denotes a superpixel.

The weight matrix for the layer of U is $P = (p_{ii'})_{n\times n}$. In the same way, the weight matrix for the layer of V is $Q = (q_{jj'})_{m\times m}$. The histogram of Texton occurrence is computed as follows. We first convert I to grayscale and convolves it with the set of 17 Gaussian derivative and center-surround filters (Arbelaez *et al.*, 2011), as shown in Figure 3.3. We use 8 oriented even- and odd-symmetric Gaussian derivative filters and a center-surround (difference of Gaussians) filter. Each pixel is associated with a 17-d vector of responses, containing one entry for each filter. These vectors are then clustered using K-means (with $K = 64$). The cluster centers define a set of image-specific textons and each pixel is assigned the integer id of the closest cluster center. Then, the histogram of their occurrence (t_i) in every superpixel (u_i) is estimated.

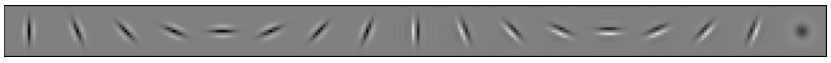


Figure 3.3: Filters for creating Textons (Arbelaez *et al.*, 2011).

Based on the across-affinity matrix W, and similarity matrices P and Q, we can denote an expanded similarity matrix

$$\tilde{S} = \begin{bmatrix} P & W \\ W^T & Q \end{bmatrix} \tag{3.7}$$

Then image segmentation using this hybrid graph model can be solved in a similar manner by spectral clustering in Section 3.2.2. Simply replace S by $\tilde{S}$ in Eq. (3.4) to compute the Laplacian matrix. The overall superpixel segmentation algorithm is summarized in Figure 3.4.

Input: Image I, k: number of clusters

1. Partition I into superpixels U and V by segmentation algorithm
2. Construct the graph $G = (U, V, E_{UV}, E_U, E_V)$
3. Compute across-affinity matrix W based on Eq. (3.1)
4. Compute affinity matrix P (Q) based on Eq. (3.6)
5. Build similarity matrix $\tilde{S}$ according to Eq. (3.7)
6. Compute the Laplacian matrix L
7. Compute the first k generalized eigenvectors $r_1, \cdots, r_k$ of Eq. (3.5)
8. Let $R \in \Re^{(n+m)\times k}$ be the matrix containing the vectors $r_1, \cdots, r_k$ as columns, use k-means algorithm to cluster $(n + m)$ rows of R into k groups

Output: k clusters

Figure 3.4: Image segmentation by bilayer superpixel grouping

3.3 Experimental Results

We evaluate the proposed image segmentation algorithm on some images from Berkeley Segmentation Data Set (BSDS), and compare it with state-of-the-art methods.

Implementation details Our framework builds a graph on superpixel nodes, which are generated by SLIC (Achanta *et al.*, 2012), though other choices are also possible. The main reason of choosing SLIC is that it is currently state-of-the-art superpixel segmentation algorithm and practically efficient. The SLIC parameters are the region size and the regularizer. For our experiments, we set region size proportional to the image size to make around 200 and 100 superpixels for two layers for every image. The regularizer is set as 0.15 for all the images. The parameters in the bipartite graph construction are set as follows $\alpha = 0.9$, and $\beta = 0.35$. The number of clustering k is set to 6 for all the experiments.

Results Figure 3.5 shows the segmentation results for an example image. The red boundary overlays with the superpixel segmentation image for visualization. By comparing with mean shift, normalized cut and UCM segmentation methods, our proposed bipartite and hybrid segmentation methods produce more reasonable results with respect to object boundaries and small objects.

Some more segmentation examples of BSDS images can be visualized in Figure 3.6. The top 4 rows are perceptually satisfactory results, and the bottom 2 rows show the typical failure cases. We report the results from the flat clustering with only local neighborhood information, bipartite segmentation results, and hybrid segmentation results. These results demonstrate the high performance of our methods on this dataset. Note that it is usually difficult for many segmentation algorithms, even the ones incorporating high-level shape priors, to segment highly textured objects from textured background. Our methods provides perceptually satisfactory results in the bear and lion images. For the typical failure cases, these images usually contain

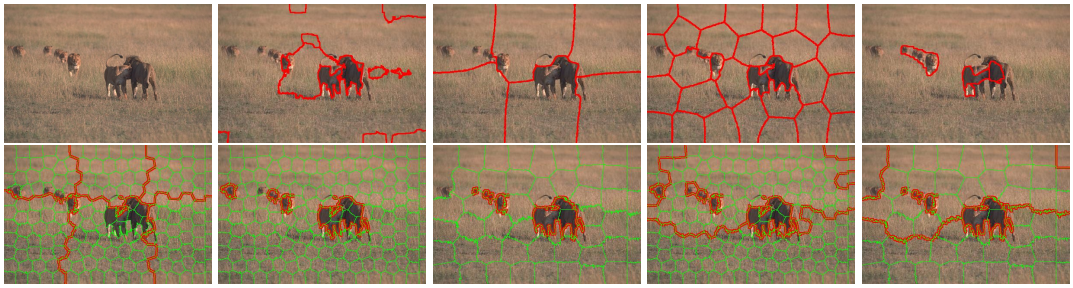


Figure 3.5: Segmentation example of lion image. Top row (from left to right): original image, mean shift (Comaniciu & Meer, 2002) segmentation result, normalized cut (Shi & Malik, 2000) result with 6 region partition, normalized cut result with 30 region partition, and UCM (Arbelaez *et al.*, 2011) result; Bottom row: flat clustering result with local neighborhood information, bipartite segmentation result (bottom layer), bipartite segmentation result (top layer), hybrid segmentation result (bottom layer), and hybrid segmentation result (top layer).

complex object appearance and texture background.

3.4 Conclusion

We have presented a novel graph-based framework for image segmentation, which is formulated as grouping a subset of superpixels that partitions a bilayer graph over superpixels, with graph edges encoding superpixel similarity. A bipartite graph is constructed to incorporate superpixel cue and long-range cue. Segmentation is then solved using spectral clustering. Furthermore, we propose a hybrid graph model that incorporates superpixel cue, mid-range cue, and long-range cue. The scheme is fully automatic, bottom-up, and unsupervised. The experiments demonstrate the high performance of our approach on the challenging dataset. Future work should study the incorporation of high-level cues.

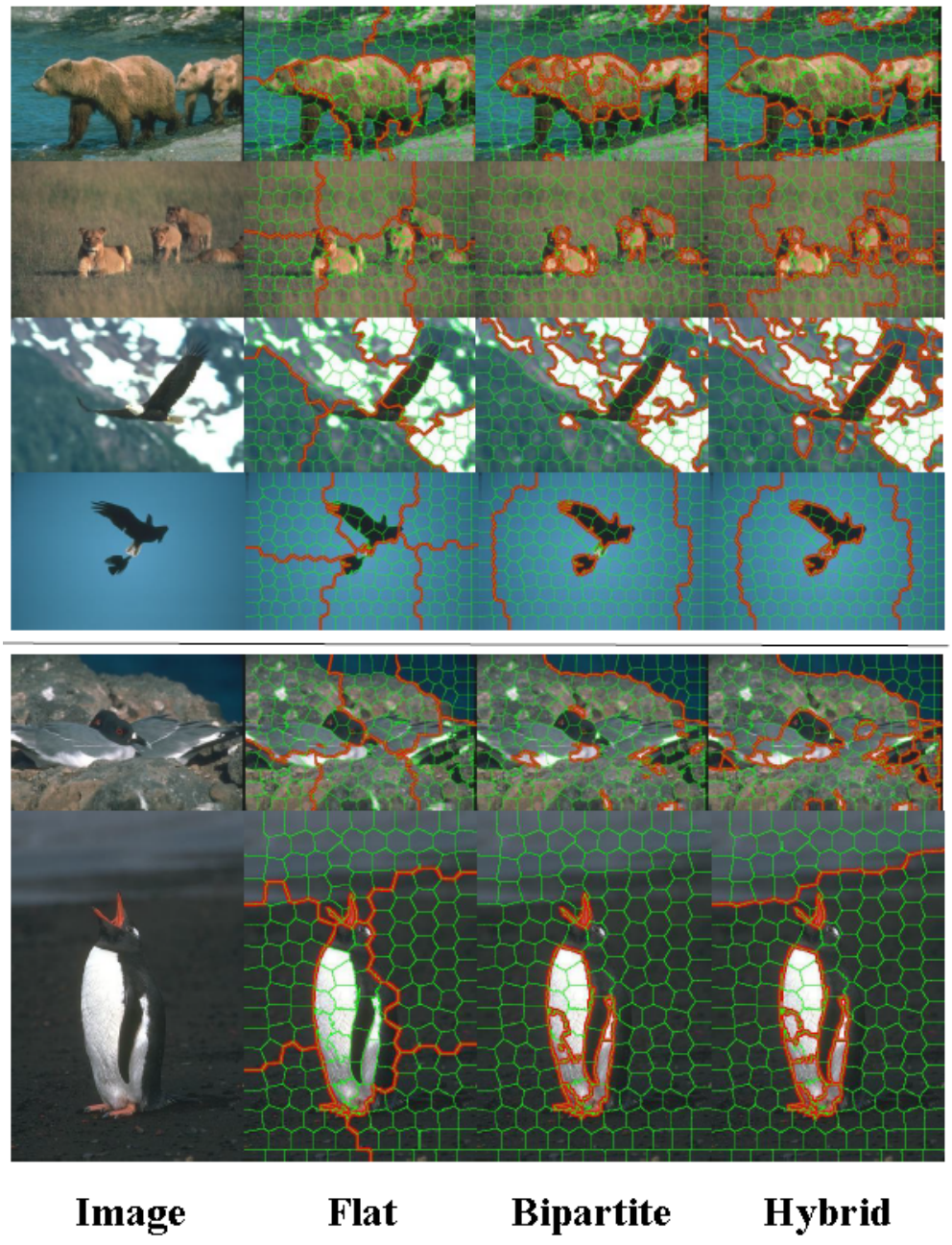


Figure 3.6: Segmentation example of BSDS images. Top 4 rows (from left to right): original image, flat clustering result with local neighborhood information, bipartite segmentation result, and hybrid segmentation result (the red boundary overlays with the superpixel segmentation image for visualization). Bottom 2 rows: typical failure cases.

Chapter 4

Estimating Layout of Cluttered Indoor Scenes Using Trajectory-based Priors

Given a surveillance video of a moving person, we present a novel method of estimating layout of a cluttered indoor scene. In this chapter, we propose an idea that trajectories of a moving person can be used to generate features to segment an indoor scene into different areas of interest. We assume a static uncalibrated camera. Using pixel-level color and perspective cues of the scene, each pixel is assigned to a particular class either a sitting place, the ground floor, or static background areas like walls and ceiling. The global topological order of classes, such as sitting objects and background areas are above ground, is locally integrated into a conditional random field by an ordering constraint. The proposed method yields very accurate segmentation results on challenging real world scenes. We focus on videos with people walking in the scene and show the effectiveness of our approach through quantitative and qualitative results. The proposed estimation method shows better estimation results as compared to the state of the art scene layout estimation methods. We are able to correctly segment 90.3% of background, 89.4% of sitting areas and 74.7% of the floor. An earlier version of this chapter appeared in Image Vision Computing (IVC) (Shoaib *et al.*, 2014).

4.1 Introduction

Estimating layout or structure of an indoor scene is important for many tasks, such as activity analysis (McKenna & Charif, 2004), robot navigation (Oriolo *et al.*, 1998; Thrun *et al.*, 2004), scene understanding (Saleemi *et al.*, 2010; Zhang *et al.*, 2011) or object placement (Jia *et al.*, 2013; Jiang *et al.*, 2012; Xu *et al.*, 2002). Specifically for the analysis of elderly activity, scene layout provides a semantic context knowledge that is necessary for long-term observation. With the help of scene context, we can localize a person and monitor his daily behavior. Semantic context also benefits the unusual event prediction, such as fall detection (Debard *et al.*, 2012). Lying on the sofa has a different interpretation from lying on the floor. With semantic context information usual lying on sofa can be taken as usual activity.

Keeping these important aspects and applications in mind, different mechanisms have been proposed for indoor scene layout estimation. State of the art methods either use spatial image

features (Hedau *et al.*, 2009; Lee *et al.*, 2010; Wang *et al.*, 2010) or use trajectory based temporal information (McKenna & Charif, 2004; Zhang *et al.*, 2011) for this purpose. However, either of the features are not enough to estimate the indoor scene layout. A major challenge for image features based techniques arises from the fact that most indoor scenes are cluttered by a lot of furniture and decorations (Wang *et al.*, 2013). They often obscure the geometric structure of the scene, and also occlude boundaries between walls and the floor. Appearances and layouts of clutters can vary drastically across different indoor scenes, so it is extremely difficult (if not impossible) to model them consistently. Similarly trajectory based techniques normally cluster the trajectory data and model only the paths (Hu *et al.*, 2004; Zhang *et al.*, 2011). They do not take care of the clutter or resting places in the scene. The modeling of resting areas (McKenna & Charif, 2004) has been done using stop points of a trajectory inside a resting place. This mechanism can not reliably used in case of noise in the trajectory data. A normal stop outside a resting area might be take as a resting place.

Though image features and trajectory data are not self sufficient for reliable scene layout estimation but they can be used together to achieve reliable indoor scene layout estimation. In this work, we propose a mechanism which learns the scene semantic context model using image segmentation mechanism in an unsupervised way. We do not use trajectories directly for scene layout estimation rather our segmentation mechanism used features both image and trajectory based features. We are also able to model the resting or sitting places in the scene. An overview of the approach is as follows. We assume that we have a static and uncalibrated surveillance camera in the scene. Given a moving person in the scene, we first model the trajectory of moving person using a set of key-points on his silhouette. We identify or cluster the regions corresponding to feet locations of moving person as floor. Given a potential floor area, we define the relative height of each point relative to the floor. Similarly using lines and trajectories we define the orientation of each point in the scene. We now incorporate height, orientation and color information into a conditional random field(CRF) to define relationship between different points in the scene. Figure 4.5 gives an overview of the CRF-based image segmentation for unsupervised scene layout estimation procedure. A graphcut based inference algorithm is run on our CRF to define the final scene segmentation or layout.

4.1.1 Contributions

The key contributions of this work are as follows:

1. Indoor scene layout estimation using both trajectory data of a moving person and image features. The estimation process is fully automatic and unsupervised. We do not use any training data. No assumptions are about the structure of the scene.

2. Efficiently estimation scene structure in the presence of scene clutter. We classify scene clutter either as sittable areas or scene background. Modeling resting areas as a separate class improves overall scene layout estimation process.

3. We show that using line segments instead of voting based straight lines we can obtain better orientation map or surface normals. Improvement in orientation map improve

overall scene layout estimation by providing correct orientations for resting places like sofa and bed.

4. Experiments are performed on a new dataset of scene videos with moving person along with publicly available videos of the indoor scenes and better segmentation results are achieved. We show using quantitative and qualitative results that by combining trajectory information of moving persons with image attributes, we can obtain an accurate indoor scene layout, superior to geometric methods. We will make the data and segmentation results publicly available for comparison.

The rest of the work is organized as follows. Related work and contributions of this work are described in Section 4.2. The proposed scene layout estimation mechanism is elaborated in Section 4.3. Image segmentation mechanism used for scene layout estimation is explained in Section 4.4. The performance of the approaches are evaluated in Section 4.5. The proposed approach is compared with other approaches in this section followed by a conclusion in Section 4.6.

4.2 Related Work

Layout of indoor scenes have been estimated in literature mainly using single-image segmentation (Hedau *et al.*, 2009; Lee *et al.*, 2010; Pero *et al.*, 2011, 2012; Schwing & Urtasun, 2012; Wang *et al.*, 2010). Such techniques use the image attributes like line segments, geometric context and color etc to define 3D structure or geometry of the scene. However, recognition of scene structure using only image features is challenging. In Hedau *et al.* (2009); Hoiem *et al.* (2007), different features are learned from a image database and then these learned features are used to train a classifier to segment a scene into different layers like ceiling, walls, clutter etc. Used features like color context are not discriminative enough for different classes. Due to color similarity a part of a wall might be detected as scene clutter or vice versa. Another set of approaches tries finding volumetric structures inside the scene to define different objects in the single images (Gupta *et al.*, 2011; Hedau *et al.*, 2010; Lee *et al.*, 2010; Pero *et al.*, 2012). They are able to find cubic objects like beds etc. They have high dependencies on straight lines in the scenes. In home environment it is difficult to detect all straights lines on objects due to cluttered scene and occlusions. Such approaches fail to detect objects like bed or sofa if they do not have enough straight lines and cubic constraint is not fulfilled.

Some other techniques use supplementary information like laser range data (Schuster *et al.*, 2010), or Kinect 3D data (Koppula *et al.*, 2011; Silberman *et al.*, 2012; Taylor & Cowley, 2012) for scene layout estimation. Similarly different areas in a scene are marked as sittable using 3D data by their ability to support a sitting action (Grabner *et al.*, 2011). Structure from camera motion has also been used in the layout estimation of the indoor scenes (Tsai *et al.*, 2011).

Tracking information has also been used in literature to couple different actions with certain regions in the scene. Resting areas were modeled as a Gaussian mixture using minimum description length by clustering image points where a person stopped in the scene in McKenna & Charif (2004). Simply using tracking information is not sufficient enough for scene layout estimation while a person stopping outside a resting area might be taken as a resting area.

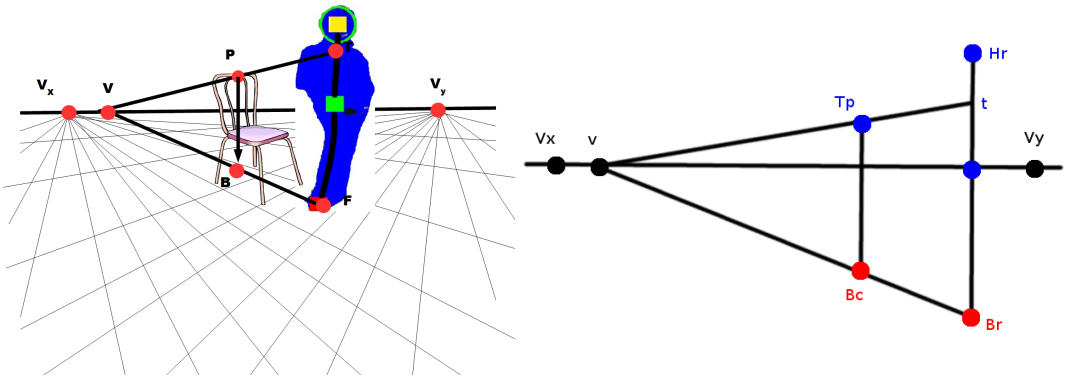


Figure 4.1: Height computation. (a) Process for finding the floor of a point p, (b) Process to compute the height of the point p.

Some other trajectory based approaches (Grabner *et al.*, 2011; Gupta *et al.*, 2011; Shoaib *et al.*, 2011; Wu & Aghajan, 2011) detect different areas in the scene by object interactions. For example a chair or a sofa is detected when some one sits in a particular scene area. Motion or person interactions alone are not reliable enough for scene layout estimation. Motion-based feature like speed are affected by the errors in moving object detection mechanism. Similarly user interactions based methods are dependent on user detection and posture classification. Any error in these two module will propagate in scene layout estimation process.

In our work we build on these efforts and take one step further to jointly segment scene and sitting places. We combine trajectory information of the moving persons along image attributes like color and perspective cues to segment indoor scene into activity areas like floor, inactivity areas or sittable places like bed, table, sofa and the remaining image area as background. We assume that sitting places are higher than floor and have orientation similar to floor. As objects like tables can also be used sitting and poses same attributes, we also consider them as sitting places.

4.3 Unsupervised Scene Layout Estimation

In order to estimate layout of a given scene, we use trajectory of coarse body motion as a reliable low level feature. In our case a trajectory T is a sequence of K correspondences

$$T = \{T_1, ..., T_t, ...T_K\}$$

where vector $T_t = [H(x,y), C(x,y), F(x,y)]$ compactly represents a person's location in terms of its corresponding key-points at time t. In order to find the key-points in each frame moving persons are segmented from the background scene using a combination of color and gradient-based background subtraction method (Shoaib *et al.*, 2009). We then use connected component analysis to find center of mass C(x,y) and ellipse fitting to define head location H(x,y). Assuming the person in standing posture, feet location F(x,y) is defined by projecting a medial axis from head to the silhouette bottom. Each key-point is a 2d location in the image

referred to as point (x,y) in coming text in this chapter.

The correspondence vectors T_t are monitored over time to record certain parameters from the movements of the persons. The scene areas through which feet or lower body centroid (in case of person to object occlusion) F pass are marked as floor areas B. Figure 4.2 shows the key-points extracted from a video and the unsupervised learning process for floor area B.

4.3.1 Features for Segmentation

For the layout estimation we use three types of information, namely color, height map H and orientation map O. We consider the layout estimation as a segmentation problem and define a conditional random field (CRF) (Lafferty *et al.*, 2001) using our three features. We will explain this procedure in Section 4.4.

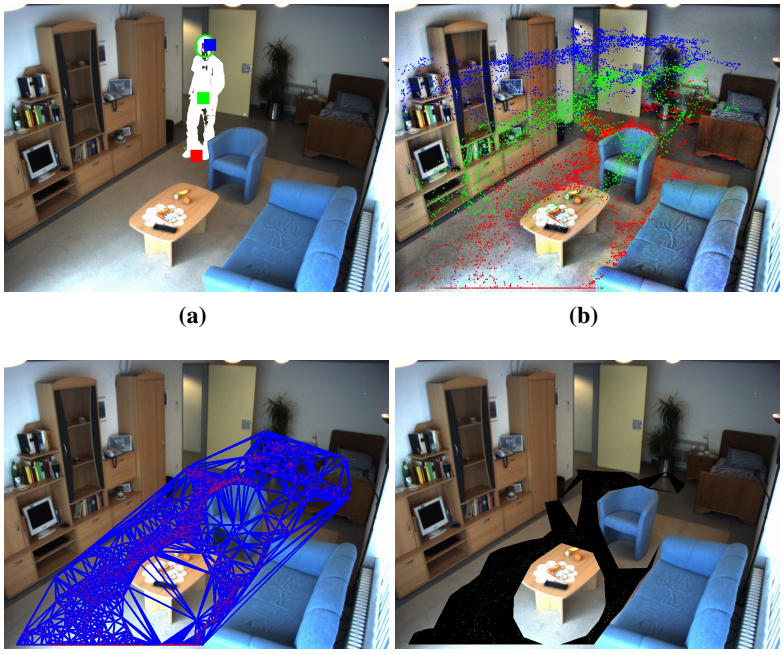
Height map Height map describes the relative homogeneous height h of each pixel with respect to the floor B. Highly probable floor pixels have height zero. In order to define the height for rest of the pixels in an image, we compute vanishing points $[v_x, v_y, v_z]$ in the scene. We use a combination of trajectory information (Junejo & Foroosh, 2006) and lines for this purpose. In order to minimize the effects of noisy trajectories we follow a RANSAC-based method to classify inliers and outliers and in turn to find three orthogonal vanishing points. The vanishing line of the floor plane can then be found by the two vanishing points v_x and v_y: $V_L = v_x \times v_y$. The key-points information in the form of head to feet $\overline{HF}$ and centroid to feet $\overline{CF}$ correspondences serve as a basis for height computation. For height computation we need the projection of every point p on the floor plane. Thus to define the floor of point p, first a nearest known correspondence is found. Then using nearest correspondence and vanishing line V_L a line is projected to the ground plane to define the floor. In order to find the floor for the current point p vanishing line V_L can be used. Figure 4.1(a) shows the process of finding the floor of a point. Initially a line connecting the top point T (on the medial axis of a nearest standing posture) and point p is intersected with the vanishing line V_L. From the point of intersection v a bottom line can be projected back to the known floor point F. The unknown floor point B is orthogonal to the current point p on the bottom line.

In order to find the homogeneous height of point p, we need a known standing posture H_r, B_r as reference and the vanishing line V_L (Criminisia *et al.*, 1999). We can then use basic trigonometry to find the height of a point p.

$$\frac{H}{R} = \frac{|p - B_r|\,|\infty - H_r|}{|H_r - B_r|\,|\infty - B_r|} \tag{4.1}$$

Orientation map An orientation map defines the major orientation of each pixel with respect to the world using vanishing points. In order to compute an orientation map, the orientation of each line in the image is defined. Orientation of the lines can be used to define the orientation of the surface lying between lines (Lee *et al.*, 2009). Orientation of a point is decided by the surface on which it is lying.

Orientation of a line segment can be computed by the vanishing point that lies on the extension of the line segment in the image. Common line detectors (Duda & Hart, 1972; Matas



(a) **(b)** **(c)** **(d)**

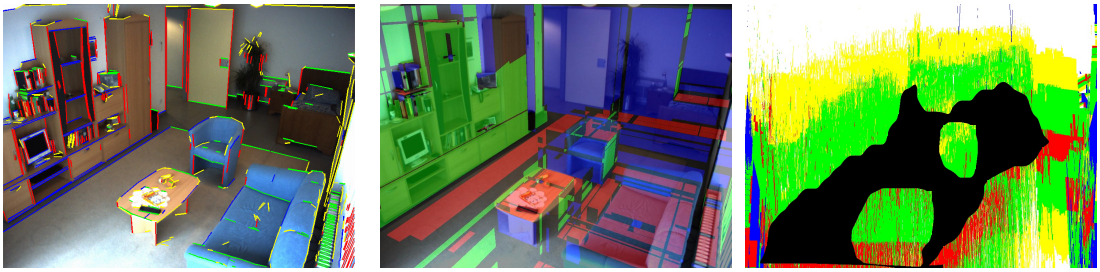
Figure 4.2: Unsupervised learning procedure for floor area. (a) Key-points, (b) Key-points correspondences, head points drawn in blue, body centroids in green and feet centroids in red, (c) candidate floor mesh for feet centroids, (d) floor area by filtering small disconnected areas.

et al., 2000) like hough transform try to find straight lines in an image using a voting-based strategy. If pixels more than a threshold vote for a line then this line is taken as valid in an image. We want to use the orientation map for modeling resting places. Resting places like a sofa or a bed may not have enough straight lines due to their irregular shape and irregular boundaries. Thus common line detectors fail to find sufficiently many lines. This fact can be seen in Figure 4.3(b) where a voting-based method (Matas *et al.*, 2000) fails to detect enough lines on the small sofa and the bed, which in turn resulted in wrong orientations in these areas. It can be observed in Figure 4.3(a) that resting places in the scene may not have enough straight lines instead they have many irregular line segments and curves that can be detected for the orientation map generation. Irregular lines do not follow the basic definition of a straight line and cannot be represented by the linear equation $y = mx + b$. Thus all the points on a irregular line might not have same slope m. As voting based techniques follow a fixed slope m, hence they have problems with irregular lines. Secondly, random alignments of pixels might generate wrong lines in cluttered indoor scene. Instead of searching for straight lines, we followed a method (Grompone von Gioi *et al.*, 2012) to identify line segments in the scene.

Let $L_o = l_{o,1}, l_{o,2}, ..., l_{o,n}$ be the set of line segments of orientation o, where $o \in [x, y, z]$ denotes one of the three orientations. Orientation o of a line is determined by a parallelism check of a line with known vanishing points (v_x, v_y, v_z). Figure 4.4(a) shows the orientations of lines, L_z drawn in red, L_y drawn in green, L_x drawn in blue while lines with unknown

(a) (b) (c)

Figure 4.3: Lines in the scene. (a) shows the edges in the scene, (b) shows the lines by a hough transform based voting mechanism (Matas *et al.*, 2000), (c) shows the lines by line growing mechanism (Grompone von Gioi *et al.*, 2012).



(a) (b) (c)

Figure 4.4: Orientation and height maps. (a) shows the lines membership for a particular orientation, (b) shows the orientation map using line segments, (c) shows the height map.

orientations are drawn in yellow.

For the pixel orientation a "sweep" of the lines in an image area towards the corresponding vanishing point is calculated. For example a sweep $S(l_{x,i}, v_y, \alpha)$ of a line $l_{x,i}$ towards a vanishing point v_y by amount α is the set of pixels that is supported by line $l_{x,i}$ to be orientation z (Lee *et al.*, 2009). Figure 4.4(c) shows the quantized height of each pixel. Floor points with lowest possible height are drawn in black. Pixels with very low height are drawn red. Pixels that have height less than the average body centroid height are in the probable areas of inactivity zones, and are drawn green. Pixels higher than body centroids are drawn in yellow, and pixels having height higher than head positions are drawn in white. The pixels whose height can not be estimated due to lack of neighbors are drawn in blue.

4.4 Joint Conditional Segmentation

Figure 4.5 gives an overview of the CRF-based image segmentation for unsupervised scene layout estimation procedure. First column shows the original surveillance scene image, keypoint trajectories of the moving persons and lines in the scene image. Second column shows the three features used to define unary class potentials i.e color priors areas are used to de-

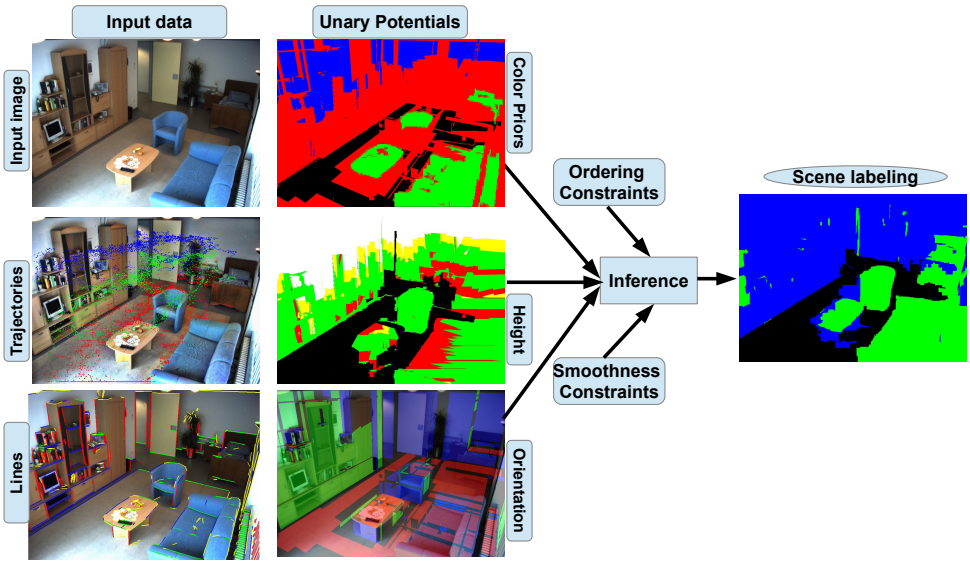


Figure 4.5: Unsupervised scene segmentation procedure for inactivity zones using CRF. First column shows the input data including image of the surveillance scene, key-point trajectories and lines in the scene image. Second column shows the three features used to define unary class potentials. i.e. color priors, relative quantized height and surface orientations. Binary potentials, i.e. ordering and smoothness constraints are added to define homogeneous regions neighborhood relationship. Inference procedure finds the optimal scene segmentation by minimizing energy on CRF.

fine Gaussian distributions of different classes. Quantized height defines the relative height of each pixel in the image, and surface orientations defines the normal orientations at each pixel. Color prior areas for different classes are selected using their known reliable properties. Binary potentials, i.e. ordering and smoothness constraints are added to define homogeneous regions neighborhood relationship. Graph cut based inference procedure is used to find the optimal scene segmentation by minimizing energy on CRF. Different areas in an indoor scene are modeled using joint conditional multi-class segmentation. Initially we segment a scene into homogeneous regions based on color and height information (Felzenszwalb & Huttenlocher, 2004b). Using prior knowledge of the potential floor B in the scene and pixel-level information then each homogeneous region is assigned to either ground floor GF, one of the sitting places ST or static background BG. The global topological order of the classes such as resting places are above ground and background is also above ground or at same level as resting places, is locally integrated in to a CRF (Barth *et al.*, 2010; Lafferty *et al.*, 2001) by ordering constraints. For a given image with N homogeneous regions, each homogeneous region and its corresponding pixels can take a unique label from a set L, where $L = l_1, ..., l_N$ represent the assignment of potential classes $C_1, ..., C_J$ to each homogeneous region. The optimum labeling is computed by finding the minimum energy configuration of the CRF. The energy function

characterizing the CRF used for the scene segmentation is as follows

$$E(L, f, \Theta) = \sum_{i=1}^{N} \Phi(l_i, f_i, \Theta) + \sum_{(a,b)\in B} \Psi(l_a, l_b, f_a, f_b, \Theta) \tag{4.2}$$

where Φ denotes set of unary potentials which are defined for all the classes using all the features. Ψ denotes a pairwise term which depends upon labels of neighboring homogeneous regions a and b. The feature vector f_i defines the color, height and orientation of the of each homogeneous region, i.e. $f_i = \{h_i, o_i, c_i\}$. The parameter set Θ includes the knowledge of color seed areas of the different classes and relative position of different classes in the scene. The color seed areas define prior colors for different sitting places $ST = \{ST_1, ..., ST_M\}$, floor GF and background BG in the scene. These areas are defined using some basic properties and feature set f_i. Sitting places ST are the areas or surfaces in the scene that have vertical orientation and they are higher than floor surface. Floor GF is the area with zero height and vertical orientation, while background BG are the areas with highest heights and orientations other than vertical. These prior areas are only used to define the color potentials for different classes and do not impact other potentials. They are also not involved in the segmentation process directly.

4.4.1 Unary Potentials

Unary potentials capture the labeling preference for a single class. Each potential function is defined for all candidate classes C_j. Each potential function has different discriminative criteria for different target classes. We use the following three unary potentials in our CRF model. Height potential encodes the relative quantized height H_i of each homogeneous region in the scene image w.r.t. the ground floor. Ground floor is considered to have zero height, the background class has regions that are higher than any other class. While sitting places or areas are considered to be near floor and have lower height than the background. The height potential is given by

$$\Phi_1^H(l_i = C_j, f_i, \Theta) = \begin{cases} -\log s(H_i - H_{min}, \lambda_{gs}, k_{gs}^j), & C_j = GF \\ -\log \Pi(H_i, \lambda_{st}^j, H_{min}^{'}, H_{max}^{'}, k_{st}^j), & C_j = ST_m \\ -\log s(H_i - H_{max}, \lambda_{bg}, k_{bg}^j), & C_j = BG \end{cases} \tag{4.3}$$

where minimum height H_{min}, $H_{min}^{'}$ and maximum height H_{max},$H_{max}^{'}$ values are decided using known average key-point heights. $s(x, \lambda, k)$ is one dimensional sigmoid function with width λ and turning point at $x = 0$, scaled to the range k with

$$s(x, \lambda, k) = (k_{max} - k_{min})/(1 + \exp(-x/\lambda)) + k_{min}$$

Similarly Π is a gating function for input x that is composed of two opposite sigmoid functions with slope λ

$$\Pi(x, x_{min}, x_{max}, \lambda, k) = (k_{max} - k_{min}) \cdot (s(x - x_{min}, \lambda, 0, 1) - s(x - x_{max}, \lambda, 0, 1)) + k_{min} \tag{4.4}$$

The orientation potential encodes the orientations or normals of homogeneous regions according to the orientation of the surfaces on which they lie. The ground surface GF have vertical orientation O_v. The sitting places ST have both horizontal and vertical parts. Hence they have either horizontal O_h or vertical orientation O_v. The background areas BG should always have one of the two horizontal orientation O_h. The orientation potential is given by

$$\Phi_2^O(l_i = C_j, f_i, \Theta) = \begin{cases} -\log \mathrm{d}(x, O_v, k^j), & C_j = GF \\ -\log \mathrm{d}(x, O_v, k^j) \vee -\log \mathrm{d}(x, O_h, k^j), & C_j = ST_m \\ -\log \mathrm{d}(x, O_h, k^j), & C_j = BG \end{cases} \tag{4.5}$$

where d is a customized delta function that returns higher potential in case of match and lower potential otherwise

$$\mathrm{d}(x, o, k) = (k_{max} - k_{min}) \cdot \delta(x, o) + k_{min}$$

Color potentials are modeled for the color similarity in the potential class areas using modified Gaussian mixture models(GMM). We define a separate GMM for each prior or seed area of a class. The parameters for each GMM are defined from color values in the seed area in the scene. The color potential is

$$\Phi_3^c(l_i = C_j, f_i, \Theta) = \begin{cases} -\log \mathrm{gmm}(c, \Theta_{gs}, k_{gs}^j), & C_j = GF \\ -\log \mathrm{gmm}(c, \Theta_{st}, k_{st}^j), & C_j = ST_m \\ -\log \mathrm{gmm}(c, \Theta_{bg}, k_{bg}^j), & C_j = BG \end{cases} \tag{4.6}$$

where modified gmm is

$$\mathrm{gmm}(x, \acute{C}_x^N, k) = (k_{max} - k_{min}) \cdot [\sum_{n=1}^{N} max(g(x, \acute{C}_x^n, k))] + k_{min} \tag{4.7}$$

g is a bell-shaped, zero-mean, multi-dimensional Gaussian function with covariance matrix $\acute{C}_x$, defined as

$$g(x, \acute{C}_x, k) = (k_{max} - k_{min}) \cdot \exp(-1/2 x^T \acute{C}_x^{-1} x) + k_{min}$$

where c is a mean RGB vector, x is the distance of vector c from mean vector of a particular Gaussian.

4.4.2 Binary Potentials

Binary or pairwise potentials Ψ define the preferences for labels over two neighboring homogeneous regions. Two neighboring regions with the same class should be assigned high likelihood τ_1 otherwise low likelihood τ_2. We also integrate the relative hierarchy of different classes within the binary potentials. For two neighboring homogeneous regions on rows v_a and v_b where $v_a \leq v_b$ the binary terms are given as follows

$$\Psi(l_a = C_{ja}, l_b = C_{jb}, f_a, f_b, \Theta) = -\log \begin{cases} \tau_1, & j_a = j_b \\ \tau_2, & j_a \neq j_b \wedge (j_a \prec j_b \vee (v_a = v_b \wedge o_a = o_h)) \\ \tau_3, & j_a \neq j_b \wedge j_a \not\prec j_b \wedge v_a < v_b \end{cases} \tag{4.8}$$

where j represent the relative ordering of two homogeneous regions. If we assume that the rows in image increase downward and the row above v_a have orientation o_h then we assign higher potential τ_2, because BG has only horizontal orientation and it should be above all.

4.4.3 Inference Using Graph Cut

The energy function $E(L, f, \Theta)$ in Eq. (4.2) is solved using a graph-cut based inference method. The inference method tries to find the optimal solution where total energy using potentials is minimum for labeling (Delong *et al.*, 2012). It is assumed that all the individual nodes in graph share the same state (label) space. For the pairwise potentials, it is achieved that the energy for two labels taking similar values should be less than the energy for them taking different values. At pixel-level either 4 or 8 neighborhood can be considered to define pair-wise relationships. This predefined neighborhood do not exist in the graph for homogeneous regions, as number of neighbors vary for each homogeneous region. Hence we define pair-wise relationships for all neighbors of a homogeneous region.

4.5 Scene Layout Estimation Results

We compared the proposed scene layout estimation method with state of the art scene layout estimation methods (Hedau *et al.*, 2009; Lee *et al.*, 2010) on several scenes. First we compare our layout estimation results with techniques that estimate indoor scene layout using single-image segmentation (Hedau *et al.*, 2009; Lee *et al.*, 2010). Later we shall show that the layout estimation results can be improved using depth information. Single-image segmentation techniques use the line segments to define 3D structure or geometry of the scene. We used the original softwares publicly provided by the authors on their websites (Hedau *et al.*, 2009; Lee *et al.*, 2010). Software from Lee *et al.* (2010) generates only the bounding box for the room layout. For comparison we colored the background area in the bounding box to blue. Similarly software from Hedau *et al.* (2009) generates the room layout estimate where each wall and roof has a different color. As we are interested in the background area as whole, thus we gave all background areas like walls and roof a uniform blue color. In order to evaluate the layout estimation, we captured five indoor scene videos using standard video surveillance net-

work cameras for our experiments. A person walked across the scene for about 2 to 5 minutes. Additionally we used 3 video sequences from publicly available standard datasets (Edgcomb & Vahid, 2012). We assume a perspective camera view of the scene with minimum possible tilt. For pixel-wise comparison of estimation results, we generated the ground-truth for all the scenes. We marked spatial layout of the room along the resting places. All the background areas are colored blue, all resting places as green and floor as black.

Figure 4.7, Figure 4.8, Figure 4.9 and Figure 4.10 show the original images and layout estimation results for qualitative evaluation of these scenes. Our dataset represents different possible indoor scenarios like living room, bed room, office, working computer lab, dining area etc. The videos in dataset are in different resolutions ranging from 352×288 to 1024×768. The scene images have been captured in varying lighting conditions like doors and windows open or closed, artificial light on or off.

CRF based segmentation normally depends on different parameters. We performed an offline parameters optimization using particle swarm optimization (Qian & Yasuda, 2008). The CRF parameters are not sensitive and follow the intuitive rules for different potentials. They can take a range of values between 0 and 1, while still giving the similar segmentation results. We use the following minimum and maximum unary potential levels $k^{j}_{gs,bg,st} = [0.1, 0.9]$. To accommodate the vertical parts in sitting areas in case of horizontal orientation we use $k^{O}_{st} = [0.1, 0.5]$ in all our experiments. For binary terms we used the values $\tau_1 = 0.85$, $\tau_2 = 0.15$ and $\tau_3 = 0.001$. Similarly $\lambda_{bg} > 0$ $\lambda_{gs} < 0$ and $\lambda_{st} > 0$.

4.5.1 RGB-based Results

Qualitative Results In first set of experiments we analyzed the role of different unary and binary potentials. Figure 4.6 illustrates the importance of different potentials introduced in our approach. Our trajectory based height feature plays an important role. It not only improves the scene layout but also benefits in the detection of resting places. Figure 4.6 shows the segmentation results for an indoor home scene for different configurations. Original scene image is shown in (a). The manually labeled ground truth image is shown in (b). Background BG is drawn in blue, sitting places ST are drawn in green, while floor is drawn in black. Final scene segmentation using all of the unary and binary potential for homogeneous regions is illustrated in (c). It can be observed that proposed mechanism correctly identified all the inactivity zones in the scene. Some false segmentations also appears on walls that can not be avoided at homogeneous regions level and shall be removed at object-level processing. In some areas vertical wall areas behind a sitting place got merged with the inactivity. Such vertical areas do not harm the objective of inactivity zones because they just extend the wall support area of an inactivity zone. In Figure 4.6(d-f) either a unary or binary potential is removed to judge its effect on the segmentation process. In the absence of orientation or color information background areas like wall are wrongly detected as inactivity zone. In the absence of height major parts of inactivity zones got detected either as background or floor. Figure 4.6(g) shows the segmentation using only the unary potentials. A number of wrong areas appear with in the actual classes. Figure 4.6(e-f) shows the segmentations using pixel-based unary and binary potentials or only unary potentials. Segmentation at pixel-level can not detect some resting areas properly. A large number of pixels belonging to resting areas are wrongly detected either

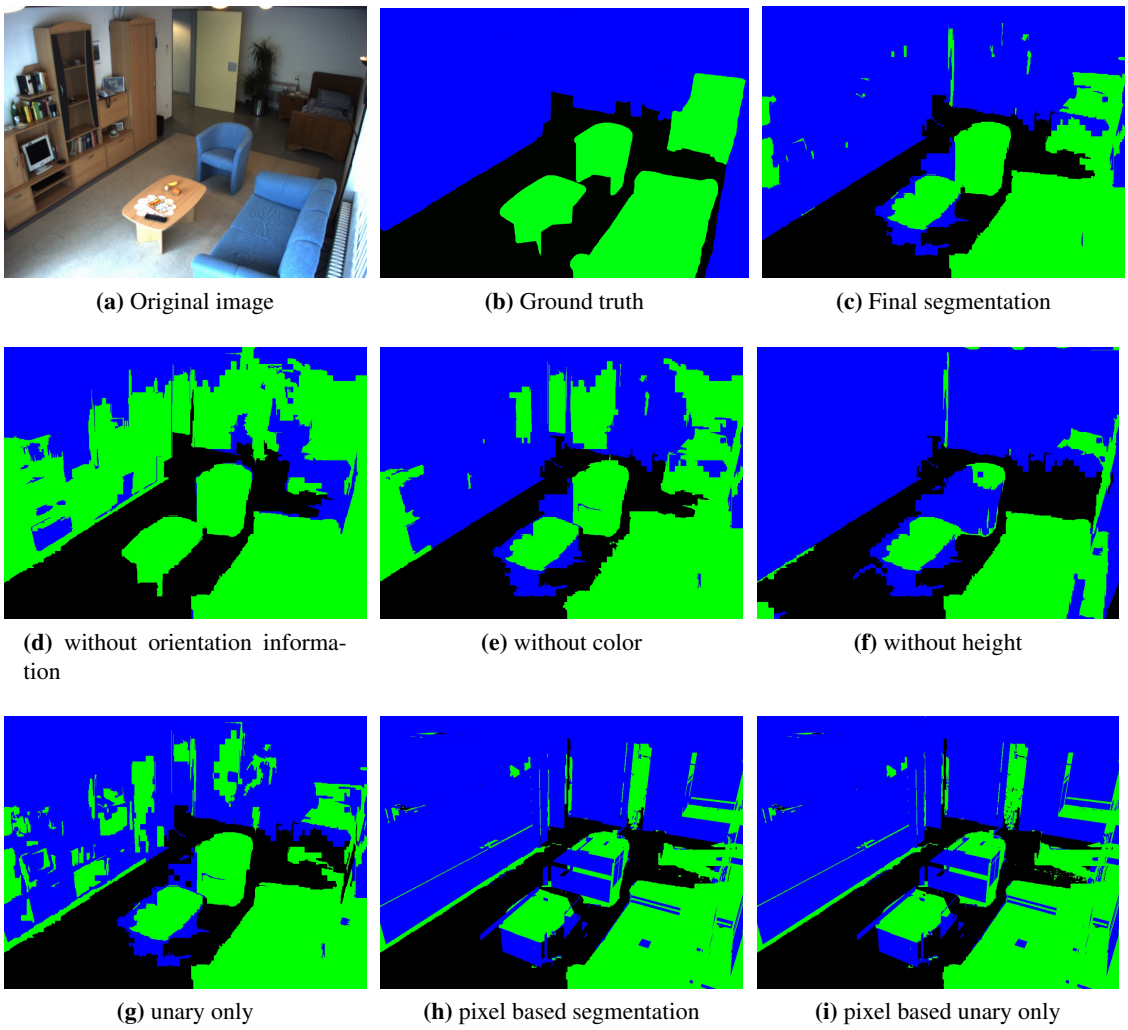
as background or ground floor.

Along internal comparison we also compare our results qualitatively with state of the art methods. Figure 4.7, Figure 4.8, Figure 4.9 and Figure 4.10 show the scene layout estimation results for different indoor scenes using the proposed and reference methods. First rows in Figure 4.7 and Figure 4.8 show the original scenes images LivingLab, StudentLab and LivingRoom, while the first rows in Figure 4.9 and Figure 4.10 show the original scenes images SurfaceClean, Pace and OfficeW. The second row in all Figures shows the corresponding ground truths. The third rows in Figure 4.7 and Figure 4.9 show the scene layouts estimated using proposed method. The third rows in Figure 4.8 and Figure 4.10 show the estimation results using Lee *et al.* (2010), while the last row in these figures show the results using Hedau *et al.* (2009). The color represents the most probable class at a specific homogeneous region. Blue represents background, black represents floor and green represents the sitting places or scene clutter. In results from Lee *et al.* (2010), background and floor areas are enclosed in a blue color box. Two upper portions in this box represent left and right walls, while lower portion represents the floor area. Sitting places are enclosed in cubic shapes. The method mainly use the line segments in the scene to define a geometry or 3D structure of the scene and assume that clutter in the scene follow a volumetric constraint. It finds the room structure in the form of walls and ceiling well but it is unable to detect the majority of the sitting areas because they do not obey a volumetric constraint due to the absence of straight lines. Different rest areas like sofa or bed might have curvy surface and lack straight lines. The third column shows the scene layout estimation results using Hedau *et al.* (2009). They use a training-based method and learn different features for different scene areas from a set of scene images. A large number of pixels belonging to sitting areas are wrongly detected either as background or ground floor due to their similarity in training data for wrong class.

Quantitative Results In order to compare the proposed method quantitatively we generated confusion matrices and pixel-wise accuracy graphs. Confusion matrix are used to analyze the percentage of correctly and wrongly classified pixels using standard parameter sets for the estimation methods. Pixel-wise accuracy graphs are used to analyze the accuracy of all the classes in a scene image while changing a particular parameter of the estimation method.

Table 4.1 shows a confusion matrix for the mean and standard deviation of segmentation results for different scenarios using the training based mechanism proposed by Hedau *et al.* (2009). We compared the layout estimation result image with its ground-truth for each scene. The mean and standard deviation values are calculated using comparison values for all the scenes. Both the ground floor GF and sitting places ST show lower number of true positives. This is mainly due to the similarity of features for different classes in the training data.

Table 4.2 shows a confusion matrix for the mean and standard deviation of segmentation results for different scenarios using the proposed method. As compared to reference method, we achieve better segmentation or estimation for background BG and sitting places ST. This is mainly due to the introduction of height feature in the segmentation process. Height classifies the different classes like resting places, floor and background even if they have very similar color and orientation. The ground floor GF shows lower number of true positives. This is mainly due to unavailability of trajectory information in some floor regions.



(a) Original image **(b)** Ground truth **(c)** Final segmentation

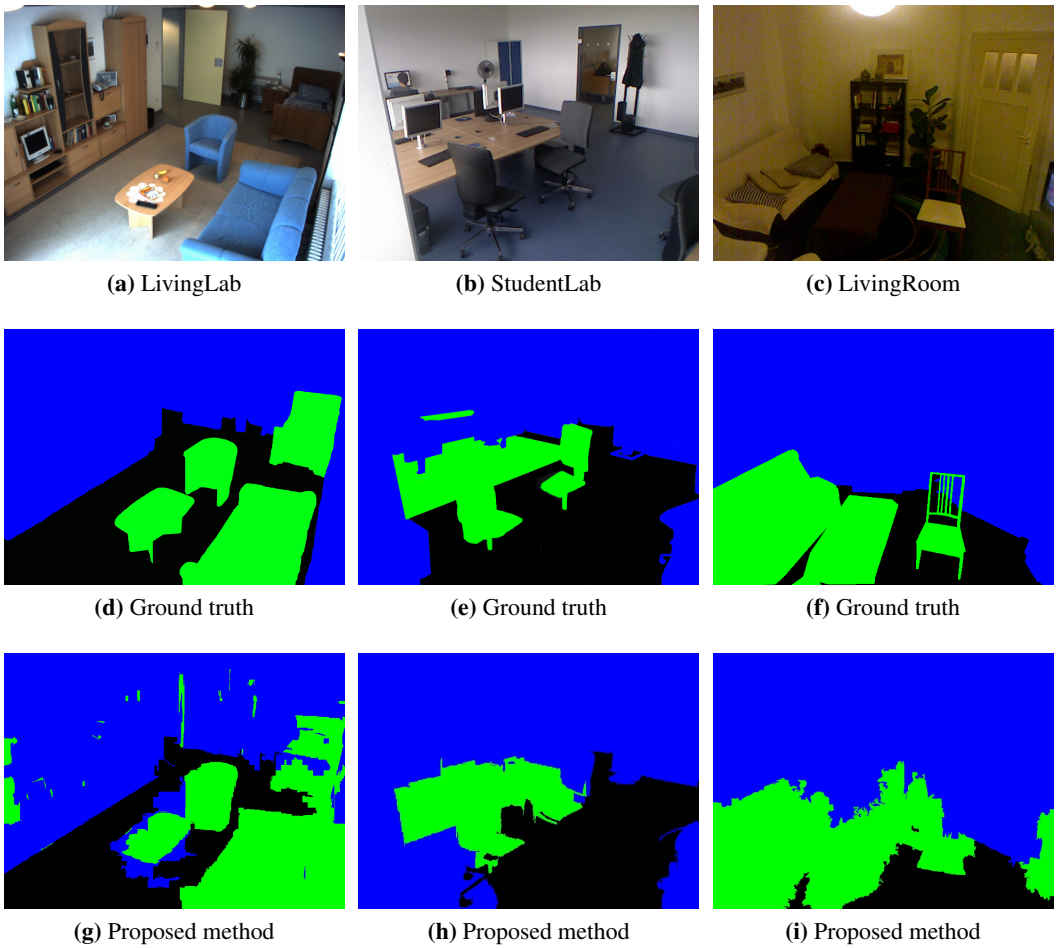
(d) without orientation information **(e)** without color **(f)** without height

(g) unary only **(h)** pixel based segmentation **(i)** pixel based unary only

Figure 4.6: Labeling results using different combinations of unary and binary potentials. A particular unary or binary potential is removed is identify its importance in the segmentation process. The color represents the most probable class at a specific pixel homogeneous region in the scene image. Blue represents background, black represents floor and green represents the sitting places.

Table 4.1: Confusion matrix for the segmentation results by Hedau *et al.* (2009) (mean and standard deviation).

GT	BG	GF	ST
BG	0.8339 ±0.0819	0.0188 ±0.0165	0.1473 ±0.0843
GF	0.0448 ±0.0783	0.7514 ±0.1595	0.2038 ±0.1219
ST	0.1281 ±0.2006	0.1262 ±0.1187	0.7457 ±0.2369



(a) LivingLab **(b)** StudentLab **(c)** LivingRoom

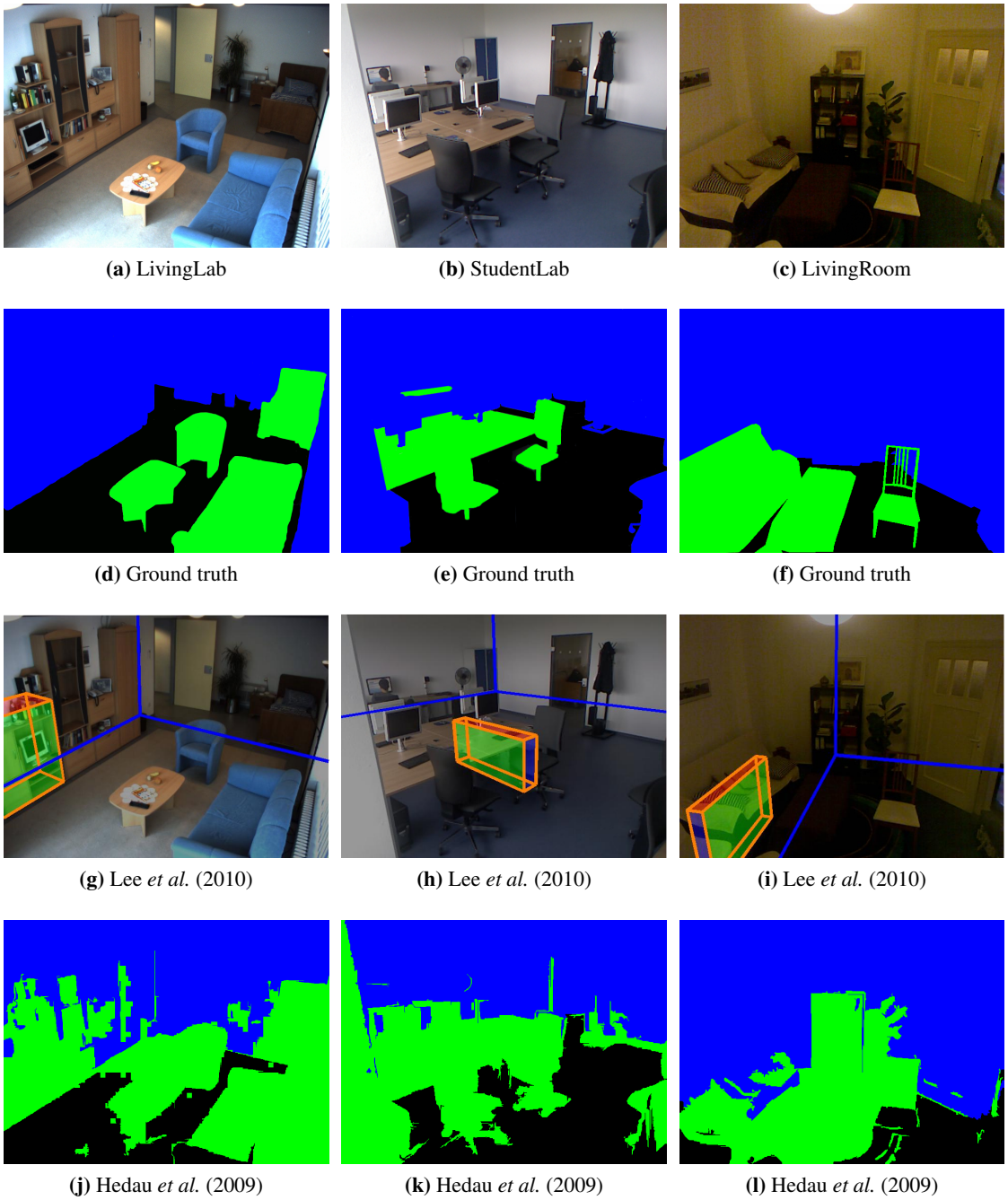
(d) Ground truth **(e)** Ground truth **(f)** Ground truth

(g) Proposed method **(h)** Proposed method **(i)** Proposed method

Figure 4.7: Scene Layout estimation results for different indoor scenes using proposed method. First row shows the original scenes images, LivingLab, StudentLab, LivingRoom. The second row shows the corresponding ground truths. The third row shows the scene layouts estimated using proposed method. The color represents the most probable class at a specific homogeneous region. Blue represents background, black represents floor and green represents the sitting places or scene clutter.

Table 4.2: Confusion matrix for the segmentation results by proposed mechanism (mean and standard deviation).

GT	BG	GF	ST
BG	0.903 ±0.0635	0.0028 ±0.0022	0.087 ±0.0627
GF	0.066 ±0.0671	0.7469 ±0.0797	0.182 ±0.1243
ST	0.064 ±0.0514	0.33 ±0.0265	0.894 ±0.0416



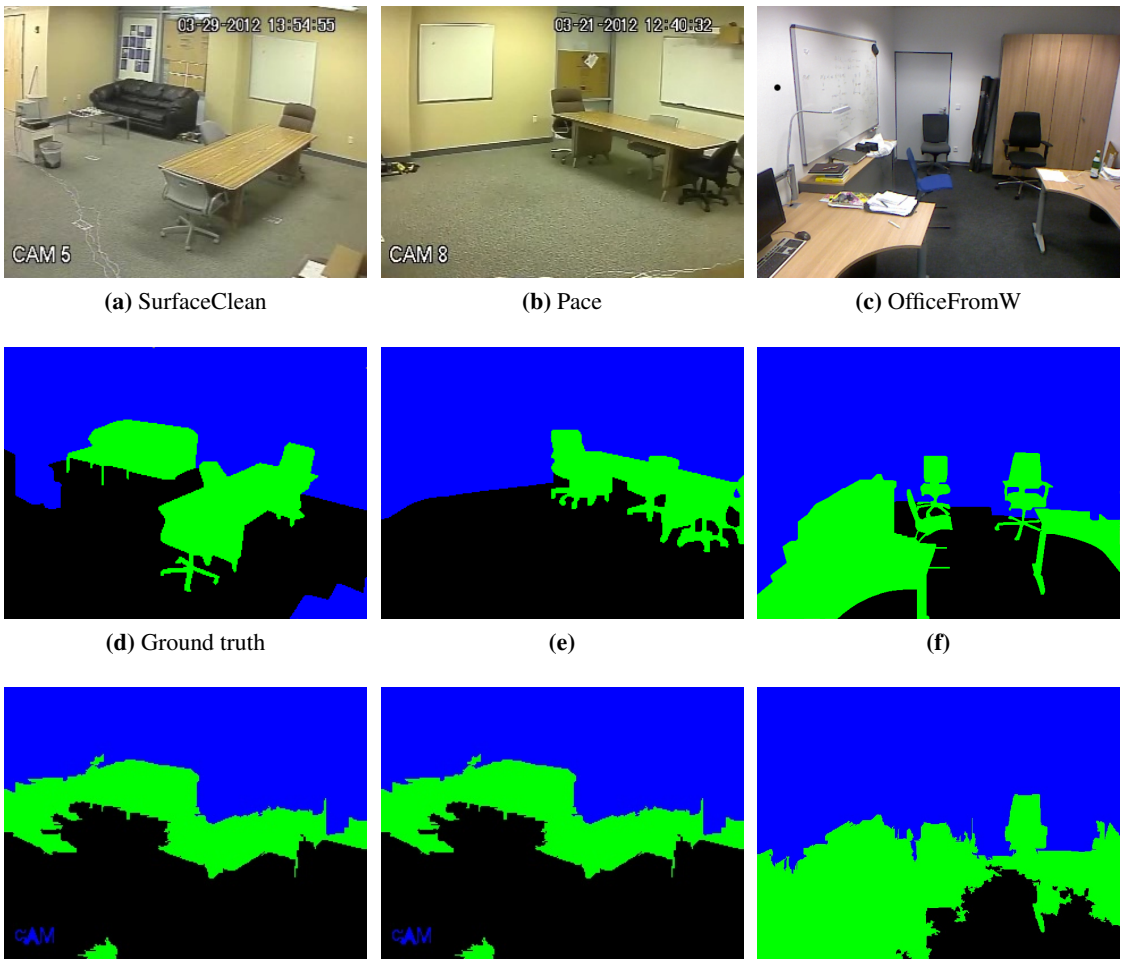
(a) LivingLab **(b)** StudentLab **(c)** LivingRoom

(d) Ground truth **(e)** Ground truth **(f)** Ground truth

(g) Lee *et al.* (2010) **(h)** Lee *et al.* (2010) **(i)** Lee *et al.* (2010)

(j) Hedau *et al.* (2009) **(k)** Hedau *et al.* (2009) **(l)** Hedau *et al.* (2009)

Figure 4.8: Scene Layout estimation results for different indoor scenes using reference methods. First row shows the original scenes images, LivingLab, StudentLab, LivingRoom. The second row shows the corresponding ground truths. The third row shows the estimation results using Lee *et al.* (2010), while the last row shows the estimation results using Hedau *et al.* (2009). The color represents the most probable class at a specific homogeneous region. Blue represents background, black represents floor and green represents the sitting places or scene clutter. In case of Lee *et al.* (2010) results background is enclosed in a blue color box, sitting places are enclosed in cubic shapes while rest of the scene is considered as floor.

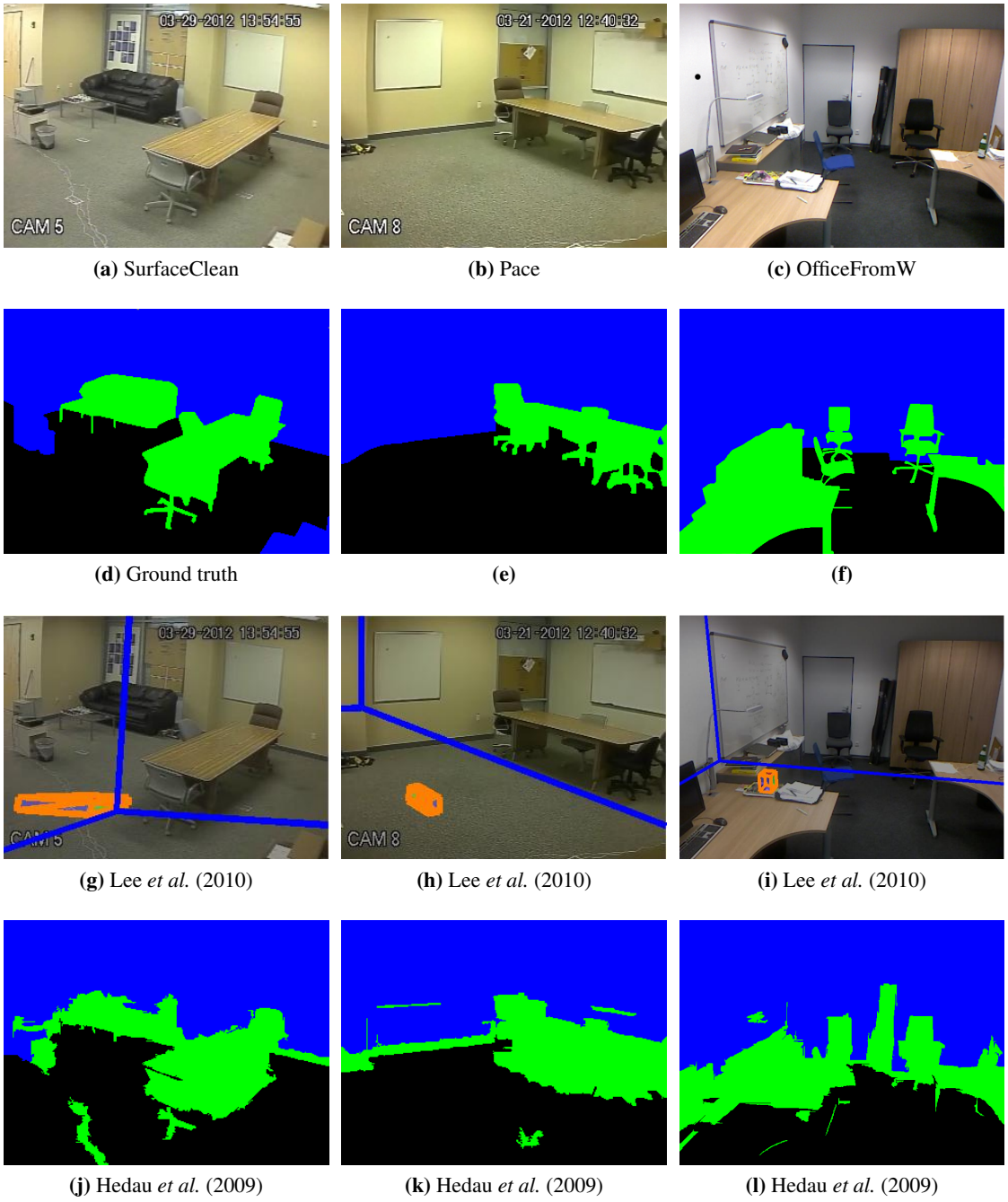


(a) SurfaceClean **(b)** Pace **(c)** OfficeFromW

(d) Ground truth **(e)** **(f)**

(g) proposed method **(h)** **(i)**

Figure 4.9: Scene Layout estimation results for different indoor scenes using proposed method. First row shows the original scenes images, SurfaceClean, Pace, OfficeW. The second row shows the corresponding ground truths. The third row shows the scene layouts estimated using proposed method.



(a) SurfaceClean **(b)** Pace **(c)** OfficeFromW

(d) Ground truth **(e)** **(f)**

(g) Lee *et al.* (2010) **(h)** Lee *et al.* (2010) **(i)** Lee *et al.* (2010)

(j) Hedau *et al.* (2009) **(k)** Hedau *et al.* (2009) **(l)** Hedau *et al.* (2009)

Figure 4.10: Scene Layout estimation results for different indoor scenes using reference methods. First row shows the original scenes images, SurfaceClean, Pace, OfficeW. The second row shows the corresponding ground truths. The third row shows the scene layout estimation results using Lee *et al.* (2010), while the last row shows the results using Hedau *et al.* (2009).

We perform a second quantitative evaluation for scene layout using pixel-wise accuracy comparison between the segmentation results and the ground truth (Gould *et al.*, 2008). Pixel-wise accuracy represents the sum of accuracy for all classes in a scene image.

$$Acc = \frac{TP_{BG} + TP_{GF} + TP_{ST}}{N} \tag{4.9}$$

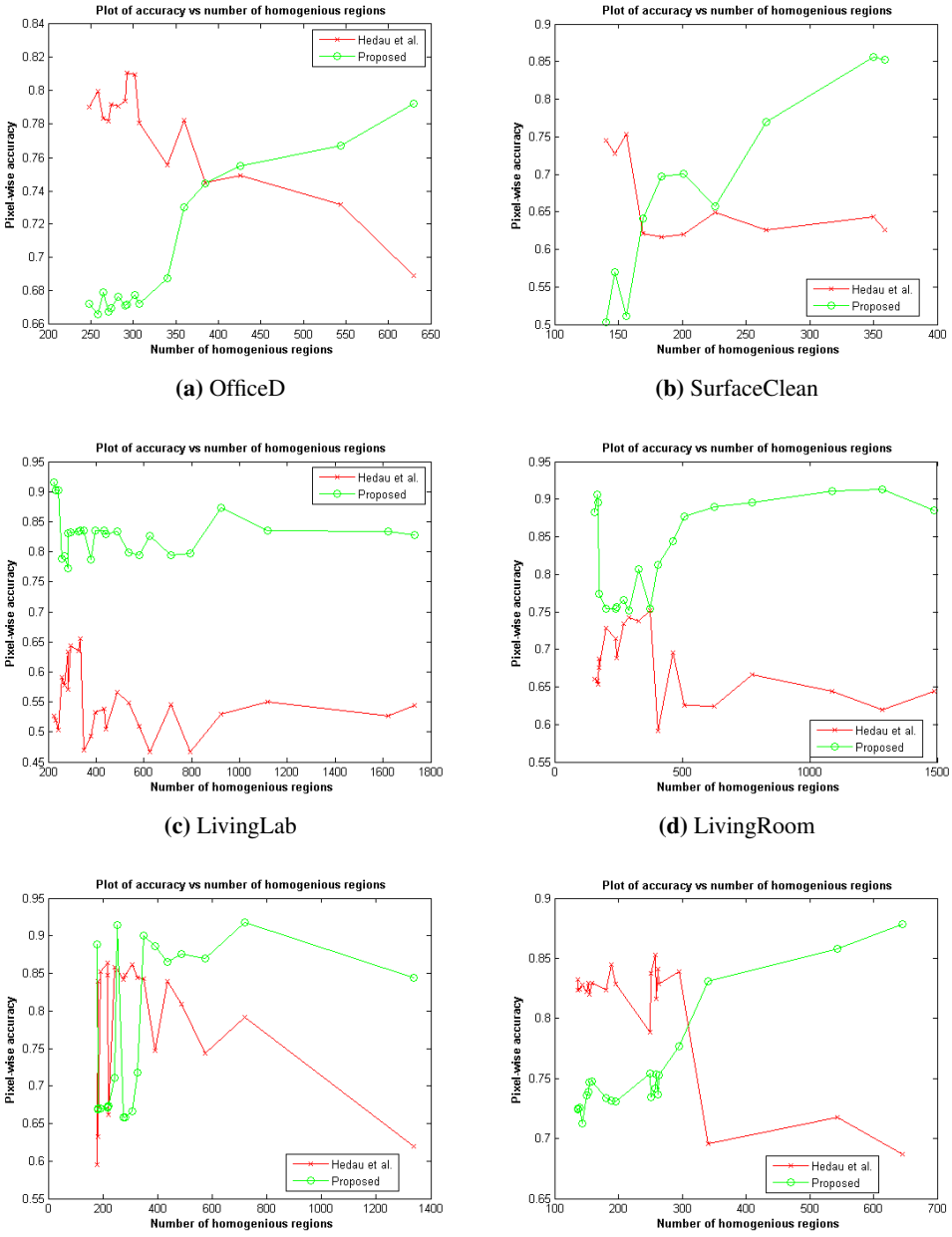
This accuracy term is calculated for different values of a parameter. In this experiment, the parameter is basically number of homogeneous regions that are used as an initial input for final scene segmentation.

Figure 4.11 shows pixel-wise accuracy curves for different indoor scenes using proposed and a reference method (Hedau *et al.*, 2009). Different accuracy values are derived by changing number of homogeneous regions in the scene image. As both proposed and and reference method (Hedau *et al.*, 2009) use homogeneous regions as an initial input to perform layout estimation. Hence, we used different number of homogeneous regions as a parameter to compare the performance of two methods. First row shows the graphs for OfficeD and SurfaceClean. The second row shows the curves for LivingLab and LivingRoom sequence, while the third row shows the graphs for StudentLab and OfficeW.

The proposed method shows better results for all the scenes except for the sequence OfficeD. For this particular sequence reference method achieves much better performance when number of homogeneous regions is low. This is mainly due to less amount of trajectory data available for this scene. Inaccurate height map is generated due to less trajectory information. As a result height information can not improve homogeneous regions and image segmentation process. Reference method (Hedau *et al.*, 2009) also showed better performance when size of homogeneous regions was too large. This fact can been seen in curves for OfficeD, OfficeW and SurfaceClean. For too large homogeneous regions the reference method performed better due to better similarity in their training data. As we use neighborhood relationships for homogeneous regions in our segmentation approach, thus we get better results when number of homogeneous regions is higher and they are smaller in size. Similarly feature values are uniform when we have smaller size of homogeneous regions.

4.5.2 Scene Context Model Using RGB-D Sensors

The scene layout estimations by the proposed method can be further improved if we use RGB-D sensors. Depth information from the sensor not only improves the homogeneous regions but depth also produces better orientation maps. These improved features are then used in the CRF-based segmentation process to produce better scene layouts. Commonly used RGB-D sensors like Kinect do not deliver a complete depth map. Kinect particularly face problems in the dark and shining areas in the scene. Similarly depth for the areas farther than 4 meter might not be accurate. Figure 4.12(b) shows that we are unable to get any depth information in different areas in StudentLab scene. In order to correct the depth information we used two mechanisms in all scenes. Kinect delivers slightly varying depth at different times. We performed temporal averaging to improve depth information from multiple images. Then we performed an interpolation or in-painting step using cross bilateral filter (Silberman *et al.*, 2012) to fill the rest of

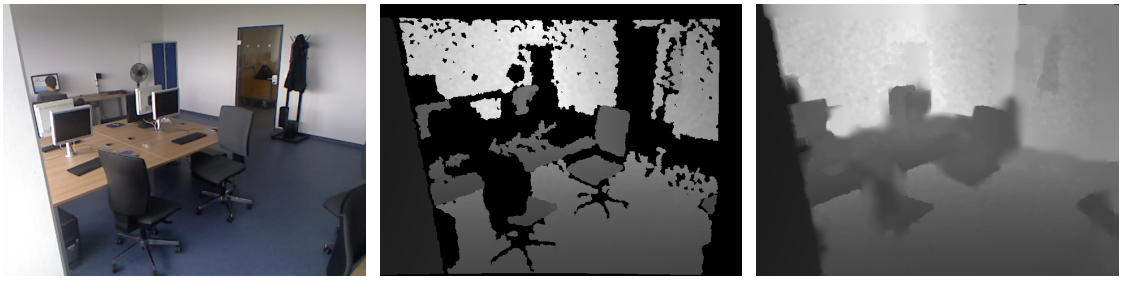


(a) OfficeD **(b)** SurfaceClean

(c) LivingLab **(d)** LivingRoom

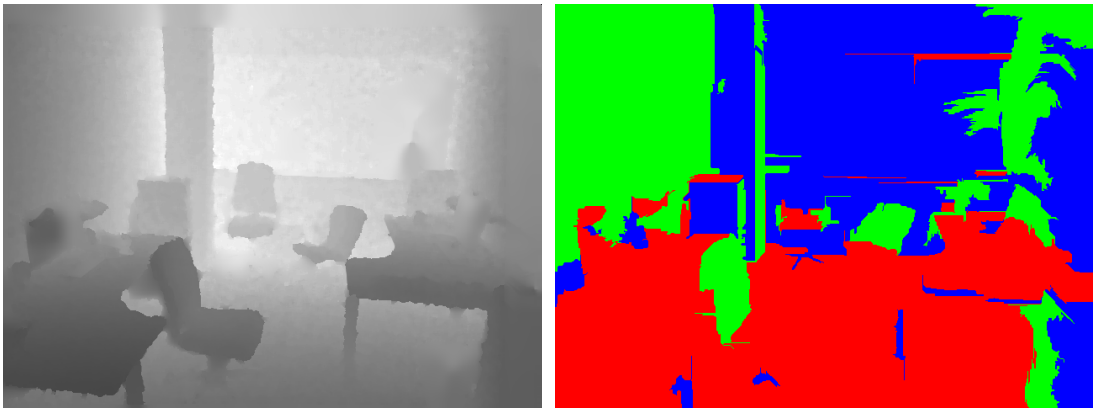
(e) StudentLab **(f)** OfficeW

Figure 4.11: Pixelwise accuracy for different indoor scenes using proposed and reference method (Hedau *et al.*, 2009) by changing number of homogeneous regions. First row shows the graphs for OfficeD and SurfaceClean. The second row shows the graphs for LivingLab and LivingRoom sequence, while the third row shows the graphs for StudentLab and OfficeW.



(a) StudentLab **(b)** StudentLab depth **(c)** Inpainted or corrected depth

Figure 4.12: Original and corrected depth for StudentLab scene.

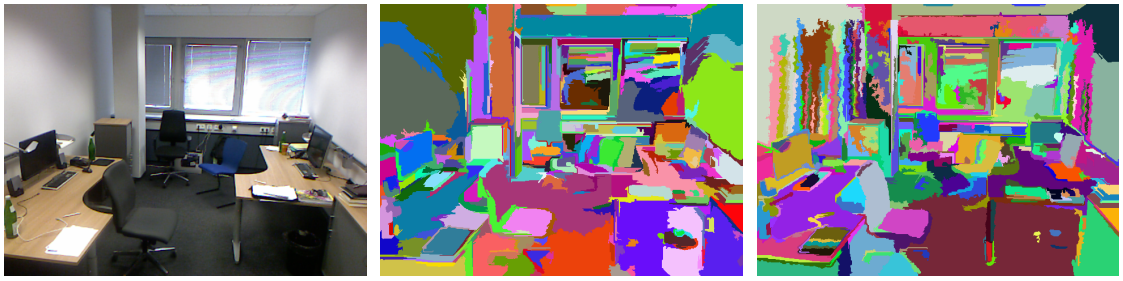


(a) OfficeD depth **(b)** Orientation map using depth

Figure 4.13: Depth and orientation map for OfficeD scene. Red represents vertical orientations Z while green and blue represent two horizontal orientations X and Y.

missing depth information from neighboring pixels. Figure 4.13(c) shows the inpainted depth for OfficeD (office from door) scene. The missing depth have been filled while maintaining the boundaries between different objects. In some severe cases depth inpainting mechanism fails to recover missing depth at object boundaries correctly (see Figure 4.12(c)). This may happen when major part of an object is missing in the original depth map.

We used depth information from RGB-D sensor to improve the quality of homogeneous regions. Homogeneous regions are used as initial input in our CRF-based segmentation algorithm. Errors in homogeneous regions propagate throughout segmentation process and result in inaccurate scene layout. Figure 4.14 shows homogeneous regions for OfficeD scene image. Color information is not always enough to generate good homogeneous regions. Using only color information results in homogeneous regions that contain multiple regions wrongly combined together due to color similarity. This can be observed in chairs areas in the scene shown in Figure 4.14(b). Part of one chair is wrongly combined with the floor, while in case of other chair a part is combined with the background. Figure 4.14(c) shows that using depth



(a) OfficeD

(b) Homogeneous regions using RGB data

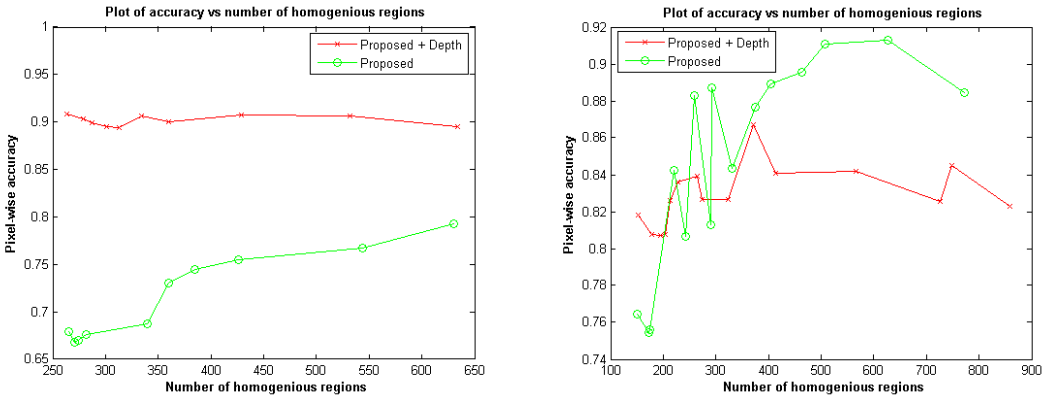
(c) Homogeneous regions using RGB and depth

Figure 4.14: Homogeneous regions for OfficeD scene image. Using color information only results in homogeneous regions that contain multiple regions wrongly combined together due to color similarity.

information we can solve these problems as chair and their nearby floor or background area are at different depths and lye at different distance from the camera. We also used depth and depth-based orientation map in our CRF based scene layout estimation mechanism. Figure 4.13(b) shows the orientation map generated using RGB-D information from Kinect sensor for OfficeD scene. Orientation map is based on the surface normals and defines the surface orientation for each point using its neighboring points (Rusu, 2009) in the point cloud. The surface normal is defined by the eigenvectors(Principal Component analysis) of the covariance matrix created from the nearest neighbors of a point. Orientation map being generated from inpainted depth might contain the areas where no single orientation is dominant. We refined the orientation map using homogeneous regions generated on the basis of color and depth information. We selected the dominant orientation for each homogeneous area.

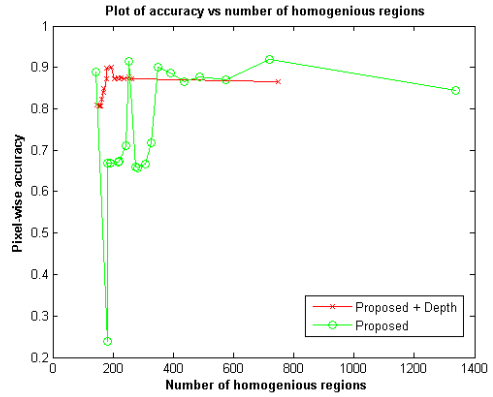
In CRF based segmentation mechanism depth information can not be used as a unary feature. Depth information is not a discriminatory feature for different scene classes. Different classes might have same depth in different areas of a scene. Floor near to camera might have similar depth as a chair near to the camera. Though depth is not a discriminatory feature in the whole image but it is an discriminatory feature in a local neighborhood in the scene. A chair should have different depth from its nearby floor and background. Keeping this point in view we integrated depth only as a binary or pairwise potential. We restricted the areas to have higher binary potential value only if they have similar depth. Depth based orientation map is used as replacement for the lines based orientation in unary potentials.

Figure 4.16 shows the scene Layout estimation results for different indoor scenes with RGB and RGB-D based features using proposed method. Depth information clearly improves the boundaries between different scene classes. Sitting areas are well separated from the background and floor. First row shows the scene layout results for OfficeD scene. Without depth parts of chair are segmented as floor due to strong color similarity with the nearby floor, similarly a number of background areas are merged with the sitting places to the similar height and color. These problems have been removed or minimized using depth information. The second and fourth row shows the scene layout results for OfficeW (office from window) and

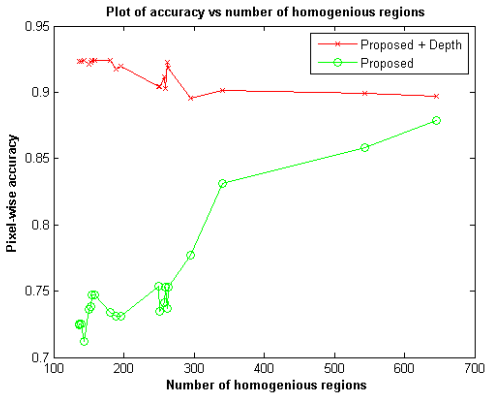


(a) OfficeD

(b) LivingRoom

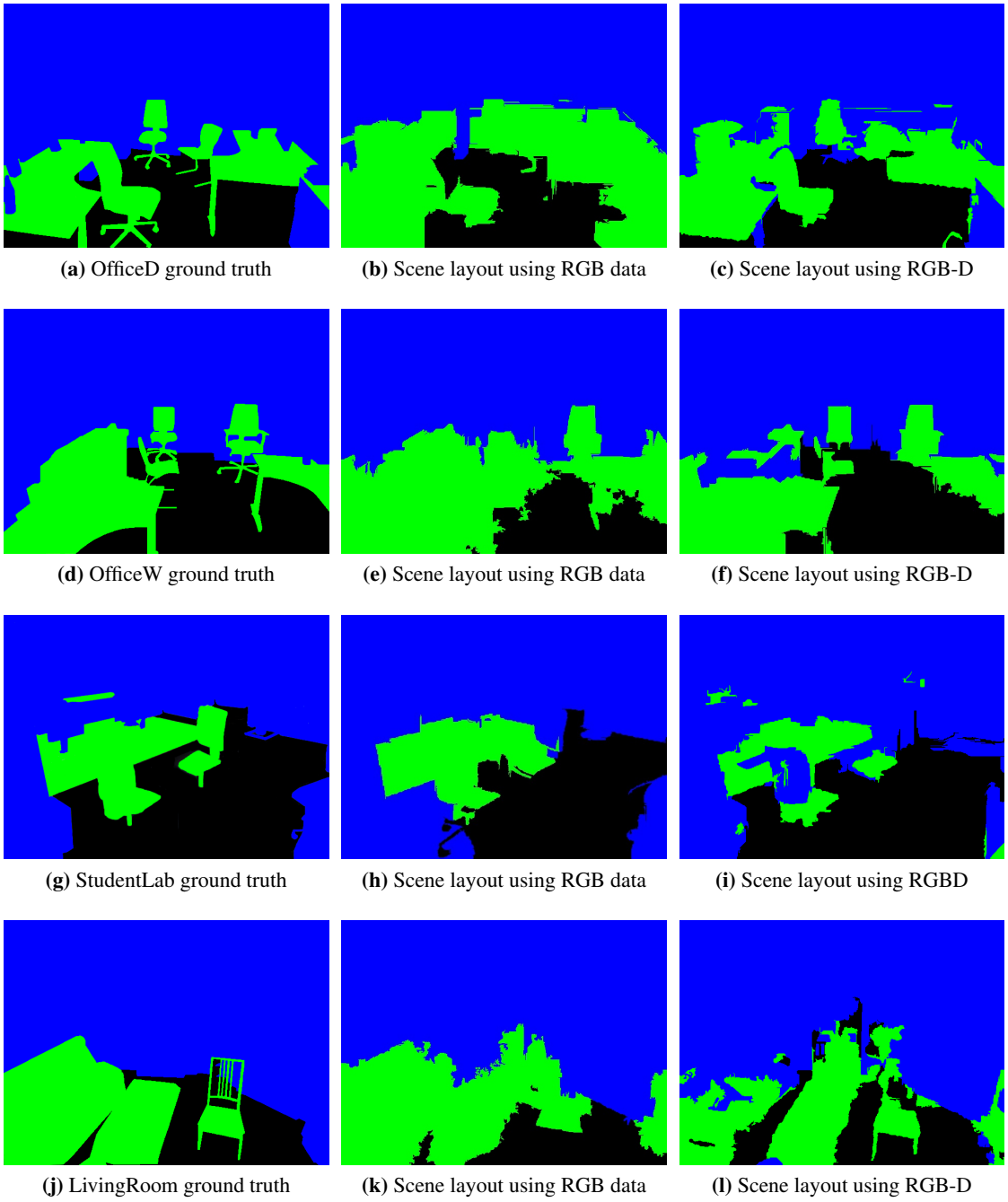


(c) StudentLab



(d) OfficeW

Figure 4.15: Pixelwise accuracy for different indoor scenes using proposed method with and without depth information by changing number of homogeneous regions. First row shows the graphs for OfficeD and LivingRoom sequence, while the second row shows the graphs for StudentLab and OfficeW.



(a) OfficeD ground truth **(b)** Scene layout using RGB data **(c)** Scene layout using RGB-D

(d) OfficeW ground truth **(e)** Scene layout using RGB data **(f)** Scene layout using RGB-D

(g) StudentLab ground truth **(h)** Scene layout using RGB data **(i)** Scene layout using RGBD

(j) LivingRoom ground truth **(k)** Scene layout using RGB data **(l)** Scene layout using RGB-D

Figure 4.16: Scene Layout estimation results for different indoor scenes with RGB and RGB-D based features using proposed method. First row shows the scene layout results for OfficeD scene, the second row shows the scene layout results for OfficeW scene, third row shows scene layout results for StudentLab scene, while last row shows results for LivingRoom.

LivingRoom scenes, a part of floor and background have been wrongly detected as sitting area in RGB only results. This problem has been reduced using the depth based features. Third row shows scene layout results for StudentLab scene. Depth based features do not show any improvement in estimation results. This is mainly due to the missing or wrong depth information from the Kinect sensor. Figure 4.12(c) shows that even interpolation cannot recover the missing depth information in some scene areas completely. We are unable to get any depth information in the chair area while it has some darker part. Boundaries in the chair are mixed with both floor and table in the unpainted depth image. Figure 4.15 shows pixel-wise accuracy for different indoor scenes by changing number of homogeneous regions using proposed method with and without depth information. Note different number of homogeneous regions were output with and without depth information. We plotted the graphs where similar homogeneous regions were available. First row shows the graphs for OfficeD and LivingRoom sequence. While the second row shows the graphs for StudentLab and OfficeW. It can be observed that introducing depth resulted in better pixel-wise accuracy. Specially depth improved performance when we have low number of homogeneous regions. Different objects or areas merge when less number of homogeneous regions are output. Errors in homogeneous regions is propagated in scene segmentation process. we can avoid or minimize this problem by using height and by maintaining the number of homogeneous regions high when no depth information is available. This fact is also evident from the graphs. Curves for higher of number of homogeneous regions are comparative. We can achieve approximate scene layout estimate with out depth, but for better accuracy we can also include depth information. Depth improves pixel-wise accuracy from 1 to 10%. Note in case of StudentLab sequence depth do not bring any advantage, while input depth image is highly erroneous. It shows that segmentation results are very much dependent on quality of depth information. Missing or wrongly interpolated depth information may not result in improvement rather it might lower the segmentation quality.

4.6 Conclusion

In this work, we presented an algorithm using the trajectories and image features to estimate the layout of indoor scenes captured with a static and uncalibrated 2-D surveillance camera. We develop a relationship between the moving person to the scene layout. By incorporating trajectory information along line segments into the same scene segmentation framework we showed that we can obtain a more accurate estimate of scene layout. The proposed method yields very accurate segmentation results on challenging real world scenes. We focus on videos with people walking in the scene and show the effectiveness of our approach through quantitative and qualitative results. We are able to correctly segment 90% of background, 89% of sitting areas and 75% of the floor. The ground floor shows lower true positive. This is mainly due to unavailability of information in some floor regions. Publicly available software from Hedau *et al.* (2009) was able to segment 83% of background, 75% of sitting areas and 75% of the floor. The publicly available software from Lee *et al.* (2010) finds the room structure well but is unable to detect the majority of the sitting areas because they do not obey a volumetric constraint due to the absence of straight lines. The scene layout information will be extremely helpful for activity analysis, navigation and other applications.

Chapter 5

Joint Object Segmentation and Depth Upsampling

With the advent of powerful ranging and visual sensors, nowadays, it is convenient to collect sparse 3D point clouds and aligned high-resolution images. Benefited from such convenience, in this chapter, we propose a joint method to perform both depth assisted object-level image segmentation and image guided depth upsampling. To this end, we formulate these two tasks together as a bi-task labeling problem, defined in a Markov random field. An alternating direction method (ADM) is adopted for the joint inference, solving each sub-problem alternatively. More specifically, the sub-problem of image segmentation is solved by Graph Cuts, which attains discrete object labels efficiently. Depth upsampling is addressed via solving a linear system that recovers continuous depth values. By this joint scheme, robust object segmentation results and high-quality dense depth maps are achieved. The proposed method is applied to the challenging KITTI vision benchmark suite, as well as the Leuven dataset for validation. Comparative experiments show that our method outperforms stand-alone approaches. This research appears in IEEE Signal Processing Letters (SPL) (Huang *et al.*, 2015).

5.1 Introduction

In recent years, the conjunctive use of ranging sensors and cameras has become more and more popular, which benefits many computer vision applications. For instance, on the one hand, high-quality range data produced by ranging sensors are often used complementary to visual information for better accomplishing tasks such as object-level image segmentation, scene parsing, and autonomous driving. On the other hand, the data generated by state-of-the-art ranging sensors, such as Velodyne HDL Lidars, are still low in resolution. High-resolution images are therefore employed as guidance to upsample depth information, making high-resolution and high-quality depth maps available.

This work focuses on two above-mentioned problems, which are depth assisted object-level image segmentation and image guided depth upsampling. The former takes advantage of depth information to segment an image into regions that correspond to objects. Previous works on this problem mainly rely upon depth maps inferred from dense stereo vision (Ladickỳ *et al.*, 2012;

Sengupta *et al.*, 2013). Recently, sparse 3D point clouds and the reconstructed corresponding dense depth maps are exploited as well in semantic segmentation for road scenarios (Chen *et al.*, 2014; Huang *et al.*, 2014). In these works, depth information is integrated either as geometric priors or as hard constraints within a Markov random field (MRF) framework to improve segmentation performance. The latter problem, aiming to generate high-resolution depth maps from sparse measurements, takes high-resolution visual images as guidance. Existing researches mainly use techniques such as bilateral filtering (Yang *et al.*, 2007), sparse representation (Gong *et al.*, 2014), or MRF (Diebel & Thrun, 2005; Park *et al.*, 2011; Zhu *et al.*, 2010). Visual information is incorporated as first or higher order constraints to guide the upsampling procedure. Particularly, higher order constraints, which are formulated according to image segmentation results, provide superior performance (Park *et al.*, 2011; Zhu *et al.*, 2010).

In contrast to conventional image segmentation or depth upsampling problems (Kang *et al.*, 2014; Liu *et al.*, 2012) that are conducted on single modality, the above two problems depend on bimodal data, and also the preprocessing results on their guidance modality. Their performance is hence improved. However, a common weakness shared by both is that they suffer from errors existing in their guidance. More specifically, the performance of image segmentation will be degenerated if the used depth map is noisy or overly smoothed on edges. Likewise, a segment that crosses over object boundaries may lead to wrong depth upsampling results. In order to prevent from such error propagation, we propose to solve these two problems jointly.

Therefore, we present a method to simultaneously deal with image segmentation and depth upsampling. The proposed method formulates both together as a bi-task labeling problem defined in a Markov random field. Although such a joint scheme has been exploited in some problems like joint image segmentation and stereo reconstruction (Bleyer *et al.*, 2011; Guillemaut & Hilton, 2011; Ladickỳ *et al.*, 2012; Tallón *et al.*, 2012), as well as joint object detection and semantic segmentation (Yao *et al.*, 2012), none of them works on depth upsampling. Moreover, most existing techniques for joint inference in MRFs obtain discrete labels. But in our case, segmentation takes discrete labels while upsampled depth is continuous. Therefore, we adopt an alternating direction method (ADM) to solve our joint problem, which alternatively uses Graph Cuts and a quadratic optimization algorithm to address each sub-problem. Experiments conducted on both the KITTI vision benchmark suite and the Leuven dataset demonstrate the superiority of our algorithm.

5.2 Problem Formulation

The objective of our work is to achieve reliable results for both image segmentation and depth upsampling when a sparse 3D point cloud and an aligned high-resolution image are given. To this end, we formulate both problems together as a multi-task labeling problem based on the MRF framework. Two types of random variables, which are respectively an object label and a depth label, are associated with each pixel. A graph $\mathcal{G} =< \mathcal{V}, \mathcal{E} >$ is hence constructed over the random variables, where the vertex set $\mathcal{V} = \{\mathcal{O}, \mathcal{D}\}$ contains the object labeling set $\mathcal{O}$ and the depth labeling $\mathcal{D}$. The set $\mathcal{E}$ refers to the edges between the vertices. With such a graphical

model, we can get an optimal result by minimizing the following energy function:

$$\begin{aligned} E(\mathcal{O},\mathcal{D}) &= \lambda_1 \sum_{i=1}^{N} \psi^{\mathcal{O}}(o_i) + \lambda_2 \sum_{i=1}^{N} \psi^{\mathcal{D}}(d_i) + \lambda_3 \sum_{i=1}^{N} \psi^{\mathcal{OD}}(o_i, d_i) \\ &+ \lambda_4 \sum_{i=1}^{N} \sum_{j \in \mathcal{N}(i)} \psi^{\mathcal{O}}(o_i, o_j) + \lambda_5 \sum_{i=1}^{N} \sum_{j \in \mathcal{N}(i)} \psi^{\mathcal{D}}(d_i, d_j) \\ &+ \lambda_6 \sum_{i=1}^{N} \sum_{j \in \mathcal{N}(i)} \psi^{\mathcal{OD}}(o_i, o_j, d_i, d_j). \end{aligned} \tag{5.1}$$

The function consists of six potential terms, which will be detailed below. $\lambda_1, \cdots, \lambda_6$ are the scalars balancing the importance of each term. N is the total number of pixels in the high-resolution image, and $\mathcal{N}(i)$ denotes the neighbors of pixel i. Besides, o_i denotes the i-th pixel's object label, taking a state from the label space $L = \{l_1, l_2, \cdots, l_n\}$, in which each state corresponds to an object instance. d_i is a continuous depth value within a perception range $D = [d_{min}, d_{max}]$.

5.2.1 Unary Potentials

Object Unary Potential The object unary potential $\psi^{\mathcal{O}}(o_i)$ evaluates the confidence for a pixel to be labeled as a particular object. Assume that the sparse 3D point cloud has been initially clustered into different object instances, as in Huang *et al.* (2014). Then, the pixels registered to these points, which are also referred to as *seeds*, are used to learn prior models of the objects. Let us denote the entire set of the seeds by $\mathcal{S}$ and the seeds belonging to the o_i-th object by $\mathcal{S}_{o_i}$. Also, we denote $\mathbf{f}_i$ as a feature extracted from pixel i, which might be the color and location (R, G, B, X, Y, Z) as in Huang *et al.* (2014) or some other more complicated features, and Θ_{o_i} as the model learned from $\mathcal{S}_{o_i}$. The potential is then designed to maximize the likelihood $p(\mathbf{f}_i|\Theta_{o_i})$, but also place a high confidence on the pre-labeled seeds. It is of the form:

$$\psi^{\mathcal{O}}(o_i) = \begin{cases} \alpha & i \in \mathcal{S}_{o_i} \\ \beta & i \in \mathcal{S}/\mathcal{S}_{o_i} \\ -\ln p(\mathbf{f}_i|\Theta_{o_i}) & \text{otherwise,} \end{cases} \tag{5.2}$$

where α and β are, respectively, a small and a large constant that are experimentally set to enforce the high confidence constraints. $\mathcal{S}/\mathcal{S}_{o_i}$ stands for the seeds except $\mathcal{S}_{o_i}$.

Depth Unary Potential The depth unary potential $\psi^{\mathcal{D}}(d_i)$ is designed to evaluate the difference between the recovered depth value and the measured one. As mentioned above, only the seeds in the image have depth measurements. Therefore, this term is defined as

$$\psi^{\mathcal{D}}(d_i) = \begin{cases} ||d_i - \bar{d}_i||_2^2 & i \in \mathcal{S} \\ 0 & \text{otherwise,} \end{cases} \tag{5.3}$$

where $\bar{d}_i$ denotes the measured depth of pixel i.

Joint Unary Potential The joint unary potential $\psi^{\mathcal{OD}}(o_i, d_i)$ penalizes the inconsistency between an object label and a depth value. This terms is commonly used when objects provide substantial information about the 3D location (Guillemaut & Hilton, 2011; Ladickỳ *et al.*, 2012). For instance, if an object is labeled as the sky, then it is of high confidence that the depth value is infinite. Such a constraint is employed in our work to prevent from recovering wrong depth values in the sky region. Thus, it is designed as:

$$\psi^{\mathcal{OD}}(o_i, d_i) = \begin{cases} \gamma & o_i = \text{sky} \cap d_i \neq d_{max} \\ 0 & \text{otherwise,} \end{cases} \tag{5.4}$$

where γ is a large constant value.

5.2.2 Pairwise Potentials

Object Pairwise Potential The object pairwise term $\psi^{\mathcal{O}}(o_i, o_j)$ encourages adjacent pixels to take the same object label if their features are similar to each other. It is defined in terms of a weighted Potts model (Boykov *et al.*, 2001):

$$\psi^{\mathcal{O}}(o_i, o_j) = w_{ij} T(o_i \neq o_j), \tag{5.5}$$

where $T()$ is an indicator, whose value is 1 when its parameter is true and 0 otherwise. The weight measures the similarity of two features and is defined by

$$w_{ij} = \exp\left(-||\mathbf{f}_i - \mathbf{f}_j||_2^2 / \sigma^2\right). \tag{5.6}$$

Depth Pairwise Potential Likewise, the depth pairwise potential $\psi^{\mathcal{D}}(d_i, d_j)$ encourages two neighbors to have close depth values if their features are similar. However, rather than using the weighted Potts model as defined above, we formulate this potential in the same form as (Diebel & Thrun, 2005) in order to achieve continuous values. It is thus defined by

$$\psi_{ij}^{\mathcal{D}}(d_i, d_j) = w_{ij} ||d_i - d_j||_2^2. \tag{5.7}$$

Joint Pairwise Potential The last term, that is the joint pairwise potential, enforces the local consistency of object labels and depth values between adjacent pixels. Based upon an observation that object boundaries and depth discontinuities are often co-occurrent, this term is designed as

$$\psi_{ij}^{\mathcal{OD}}(o_i, o_j, d_i, d_j) = w_{ij} ||d_i - d_j||_2^2 T(o_i \neq o_j). \tag{5.8}$$

It penalizes the case that two objects having similar features hold large depth difference.

5.3 Inference

The formulated problem involves two sets of variables: one takes discrete labels and the other is continuous. Therefore, inferring both together is extremely challenging. In this work, we employ an alternating direction method (ADM) (Boyd *et al.*, 2011) to iteratively solve the entire problem. In each iteration, a set of variables is solved alternatively by keeping the other set fixed. More specifically, under the assumption that depth values are known, the image segmentation sub-problem is addressed by Graph Cuts (Boykov *et al.*, 2001). When segmentation is determined, depth upsampling is performed via solving a linear system. More details of the inference are presented below. More details of the inference are presented below.

5.3.1 Inference for Object Segmentation

When depth values are fixed, the energy function defined in Eq. (5.1) degenerates to the following one:

$$\begin{aligned} E(\mathcal{O}) = \lambda_1 \sum_{i=1}^{N} \psi^{\mathcal{O}}(o_i) + \lambda_3 \sum_{i=1}^{N} \psi^{\mathcal{OD}}(o_i, d_i) \\ + \sum_{i=1}^{N} \sum_{j \in \mathcal{N}(i)} w_{ij} \left(\lambda_4 + \lambda_6 ||d_i - d_j||_2^2\right) T(o_i \neq o_j). \end{aligned} \tag{5.9}$$

We can validate that this energy function satisfies the submodularity restriction. Therefore, expansion moves of Graph Cuts (Boykov *et al.*, 2001) are applied to efficiently optimize this problem.

5.3.2 Inference for Depth Upsampling

Under the assumption that segmentation is determined, Eq. (5.1) is reduced to the form:

$$\begin{aligned} E(\mathcal{D}) = \lambda_2 \sum_{i=1}^{N} \psi^{\mathcal{D}}(d_i) + \lambda_3 \sum_{i=1}^{N} \psi^{\mathcal{OD}}(o_i, d_i) \\ + \sum_{i=1}^{N} \sum_{j \in \mathcal{N}(i)} \omega_{ij} \left(\lambda_5 + \lambda_6 T(o_i \neq o_j)\right) ||d_i - d_j||_2^2. \end{aligned} \tag{5.10}$$

It is a quadratic function defined with respect to the depth values. Therefore, efficient optimization techniques, such as a conjugate gradient algorithm (Diebel & Thrun, 2005) or even a closed-form solution, can be applied. In this work, we infer depth values via solving the linear system obtained from the closed-form solution.

5.4 Experiments

We have conducted two series of experiments, respectively, on the KITTI vision benchmark suite (Geiger *et al.*, 2012) and the Leuven dataset (Ladickỳ *et al.*, 2012). Throughout all the experiments, the involved parameters are empirically set as follows: $\lambda_1 = \lambda_2 = \lambda_3 = 0.5$, $\lambda_4 = \lambda_5 = \lambda_6 = 10$; in Eq. (5.2), the constants used for enforcing hard constraints are $\alpha = 1$ and $\beta = 500$; in Eq. (5.4), $\gamma = 500$; and in Eq. (5.6), $\sigma^2 = 3 \times 255^2$. Note that both the color and the depth values are scaled to $[0, 255]$ and in depth value 255 stands for the infinity for the sky. Therefore, 255^2 in σ^2 is used to normalize the features and 3 is a manually tuned scalar. Moreover, the iteration number of ADM is set to 4, at which most cases can get converged. Our algorithm is implemented in Matlab and has not been optimized for efficiency. Thus, each iteration takes about 25s for the KITTI and 5s for the Leuven dataset when running on our desktop with an Intel Core i5 2300 and 4 GB memory.

5.4.1 Experiments on KITTI

We first conduct our experiments on the 'City' category in KITTI (Geiger *et al.*, 2012). It has 28 different urban road scenarios and over 8,000 frames. Each frame contains color images in the resolution of 1242×375 and an aligned 360^o 3D point cloud. Preprocessing, which includes data registration and object hypothesis generation, is initially performed following the way in Huang *et al.* (2014). Then, the proposed algorithm is carried on to achieve the results of object segmentation and depth upsampling.

To validate the performance of our joint scheme, we compare the proposed method with a stand-alone version. More precisely, the stand-alone depth upsampling is obtained by optimizing the energy function containing only the potentials defined in Eq. (5.3) and Eq. (5.7). Also, the stand-alone object-level segmentation is achieved by minimizing the energy composed of Eq. (5.2) and Eq. (5.5), meanwhile taking the upsampled depth as a part of the feature $\mathbf{f}_i$. Figure 5.1 presents some typical results. It shows that, for depth upsampling, the joint approach prevents from over-smoothing object boundaries, especially on the boundaries between objects and the sky. Our approach also obtains better performance on segmentation. For example, in the stand-alone result in Column 1, segmentation errors exist on the top of the kiosk, which are resulted in due to incorrect depth values. Wrong depth information also leads to segmentation errors in the sky in Column 3 and 4, and on the top of car in Column 2, 4 and 5. Fortunately, such errors are all fixed by the joint approach.

In addition, we also conduct quantitative evaluation for the segmentation results. We randomly select 200 frames from all sequences and manually label them. The results are evaluated in terms of the global consistency error (GCE) and the local consistency error (LCE) (Martin *et al.*, 2001), both of which are ranged in $[0, 1]$ and 0 stands for the best performance. In our work, since the object hypothesis generation method produces multiple object instances, together with the sky and the ground, for each frame. Therefore, the pixel accuracy of the sky and the road is also calculated. All comparative evaluations are listed in Table 5.1, demonstrating the superiority of our joint method.

However, since no ground truth of dense depth maps is available in KITTI, we are not able to quantitatively evaluate depth upsampling results. This is also the reason that we perform the

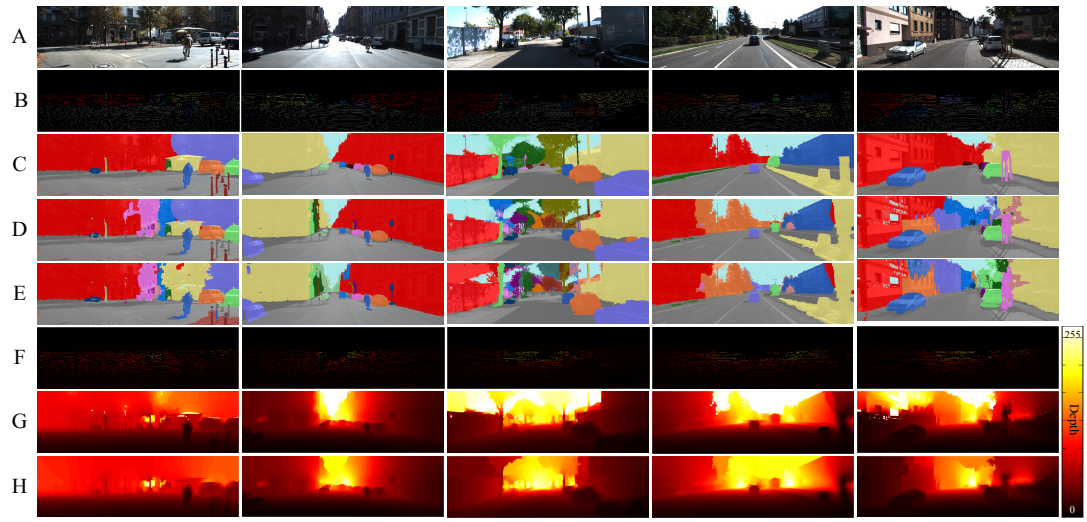


Figure 5.1: Object-level image segmentation and depth upsampling results on the KITTI dataset. (A) Original Image. (B) Object seeds. (C) Object-level image segmentation ground truth. (D) Joint segmentation result. (E) Stand-alone segmentation result. (F) Sparse depth map. (G) Joint depth upsampling result. (H) Stand-alone depth upsampling result.

Table 5.1: Quantitative evaluation results of object-level segmentation.

Label	Object		Sky	Road
Criterion	GCE	LCE	Accuracy(%)	Accuracy(%)
Stand-alone	0.12	0.11	86.84	91.03
Joint	**0.09**	**0.09**	**97.41**	**99.06**

next experiments.

5.4.2 Experiments on Leuven Dataset

The major objective of this experiment is to perform quantitative evaluation for depth upsampling. Therefore, we design a set of experiments based on the Leuven dataset (Ladickỳ *et al.*, 2012), which provides ground truth for both segmentation and dense disparity maps. In order to synthetically generate sparse depth maps, we randomly sample 10% points for each object class. This sampling rate is chosen to be consistent with the rate of sparse points in KITTI. In the experiment, we test our joint approach and compare it to the stand-alone methods. Moreover, two state-of-the-art color guided depth upsampling methods, one is based on the MRF with second order constraints (Harrison & Newman, 2010) and the other incorporates a L_0 sparse constraint into the MRF framework (Gong *et al.*, 2014), are also compared with. Meanwhile, the results of Ladickỳ *et al.* (2012) are also provided for reference.

Figure 5.2 presents two typical results. It demonstrates that our joint method corrects some object segmentation errors existing in the stand-alone approach, for instance, those on the pavement in Column D. For depth upsampling, our method avoids the over-smoothness on

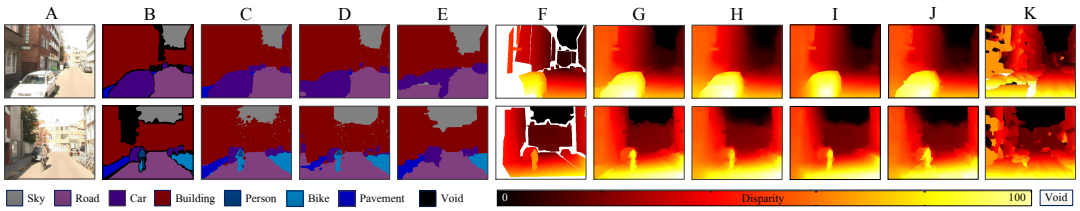


Figure 5.2: Object class segmentation and upsampled disparity results on the Leuven dataset. (A) Original Image. (B) Object class segmentation ground truth. (C) Joint segmentation result. (D) Stand-alone segmentation result. (E) The segmentation result in Ladickỳ *et al.* (2012). (F) Dense disparity ground truth. (G) Joint disparity upsampling result. (H) Stand-alone disparity upsampling result. (I) The disparity upsampling result of method in Gong *et al.* (2014). (J) The disparity upsampling result of method in Harrison & Newman (2010). (K) The reconstructed disparity map in Ladickỳ *et al.* (2012).

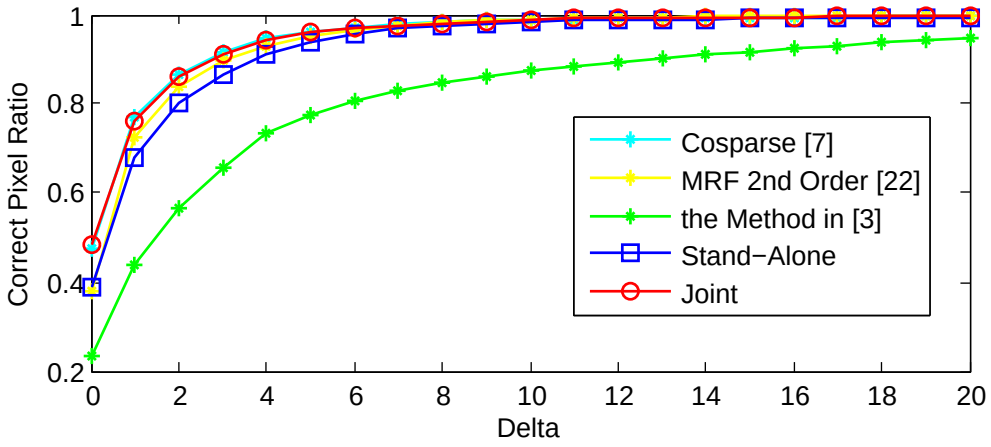


Figure 5.3: Quantitative comparison of the performance of dense disparity generation. The correct pixel ratio is the proportion of pixels which satisfy $|d_i - \bar{d}_i| \leq \delta$, where d_i is the disparity label of ith pixel, $\bar{d}_i$ is corresponding ground truth label and δ is the allowed error.

object boundaries on the disparity map when comparing to the stand-alone approach, as shown in Column G and H. Table 5.2 lists the quantitative evaluation of segmentation results and Figure 5.3 illustrates the evaluation of depth upsampling. Both show that the proposed method outperforms almost all the others.

5.5 Conclusion

This work has presented a joint method for both object-level image segmentation and depth upsampling. It is constructed over a MRF and inferred alternatively with Graph Cuts and a quadratic optimization algorithm, so that discrete segmentation labels and continuous depth values are effectively obtained. The effectiveness of our approach has been validated on the KITTI vision benchmark suite and the Leuven dataset. Experiments show that it outperforms

Table 5.2: The pixel accuracy (%) of different object classes obtained by the joint and stand-alone approaches on the Leuven dataset. The results of Ladickỳ *et al.* (2012), although it is for joint segmentation and stereo reconstruction, are also provided for reference.

Label	Sky	Road	Car	Building	Person	Bike	Sidewalk	Global
Ladickỳ *et al.*	**99.7**	99.1	88.7	96.9	$--$	59.0	54.2	95.4
Stand-alone	87.3	96.9	95.9	96.3	22.4	54.3	54.6	93.1
Joint	89.9	**99.1**	**98.9**	**98.5**	**65.5**	**90.3**	**96.2**	**97.1**

the stand-alone approaches on both segmentation and upsampling. In the future, we will explore the parameter learning scheme to obtain better performance and the promising results can further applied to achieve better performance on holistic scene understanding or other computer vision applications.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China via grants 61001171 and the Fundamental Research Funds for the Central Universities.

Chapter 6

A Generic Probabilistic Graphical Model for Region-based Scene Interpretation

The task of semantic scene interpretation is to label the regions of an image and their relations into meaningful classes. Such task is a key ingredient to many computer vision applications, including object recognition, 3D reconstruction and robotic perception. The images of man-made scenes exhibit strong contextual dependencies in the form of the spatial and hierarchical structures. Modeling these structures is central for such interpretation task. Graphical models provide a consistent framework for the statistical modeling. Bayesian networks and random fields are two popular types of the graphical models, which are frequently used for capturing such contextual information. Our key contribution is the development of a generic statistical graphical model for scene interpretation, which seamlessly integrates different types of the image features, and the spatial structural information and the hierarchical structural information defined over the multi-scale image segmentation. It unifies the ideas of existing approaches, e. g. conditional random field and Bayesian network, which has a clear statistical interpretation as the MAP estimate of a multi-class labeling problem. We demonstrate experimentally the application of the proposed graphical model on the task of multi-class classification of building facade image regions. This research appears at the International Conference on Computer Vision Theory and Applications (VISAPP) (Yang, 2015).

6.1 Introduction

The task of semantic scene interpretation is to label the regions of an image and their relations into semantically meaningful classes. Such task is a key ingredient to many computer vision applications, including object recognition, 3D reconstruction and robotic perception. The problem of scene interpretation in terms of classifying various image components in the images is a challenging task partially due to the ambiguities in the appearance of the image data (Tsotsos, 1988). These ambiguities may arise either due to the physical conditions such as the illumina-

tion and the pose of the scene components with respect to the camera, or due to the intrinsic nature of the data itself. Images of man-made scenes, e. g. building facade images, exhibit strong contextual dependencies in the form of spatial and hierarchical interactions among the components. Neighboring pixels tend to have similar class labels, and different regions appear in restricted spatial configurations. Modeling these spatial and hierarchical structures is crucial to achieve good classification accuracy, and help alleviate the ambiguities.

Graphical models, either directed models or undirected models, provide consistent frameworks for the statistical modeling. Two types of graphical models are frequently used for capturing such contextual information, i. e. Bayesian networks (BNs) (Sarkar & Boyer, 1993) and random fields (RFs) (Besag, 1974), corresponding to directed and undirected graphs. RFs mainly capture the mutually dependent relationships such as the spatial correlation. Attempts were made to exploit the spatial structure for semantic image interpretation by using RFs. Early since nineties, Markov random fields (MRFs) have been used for image interpretation (Modestino & Zhang, 1992); the limiting factor that MRFs only allow for local features has been overcome by conditional random fields (CRFs) (Kumar & Hebert, 2003a; Lafferty *et al.*, 2001), where arbitrary features can be used for classification, at the expense of a purely discriminative approach. On the other side, BNs usually model the causal relationships among random variables. Early in nineties, Sarkar & Boyer (1993) have proposed the perceptual inference network with the formalism based on Bayesian networks for geometric knowledge-base representation. Both have been used to solve computer vision problems, yet they have their own limitations in representing the relationships between random variables. BNs are not suitable to represent symmetric relationships that mutually relate random variables. RFs are natural methods to model symmetric relationships, but they are not suitable to model causal or part-of relationships.

Spatial and hierarchical relationships are two valuable cues for image interpretation of man-made scenes. In this work we will develop a consistent graphical model representation for image interpretation that includes both information about the spatial structure and the hierarchical structure. We assume some preprocessing leads to regions, either as a partitioning of the image area or as a set of overlapping or non-overlapping segments. The key idea for integrating the spatial and the hierarchical structural information into the interpretation process is to combine them with the low-level region class probabilities in a classification process by constructing the graphical model on the multi-scale image regions.

The following sections are organized as follows. The related works are discussed in Section 6.2. In Section 6.3, the statistical model for the interpretation problem is formulated. Then, the relations to previous models is discussed in Section 6.4. In Section 6.5, experimental results are presented. Finally, this work is concluded in Section 6.6.

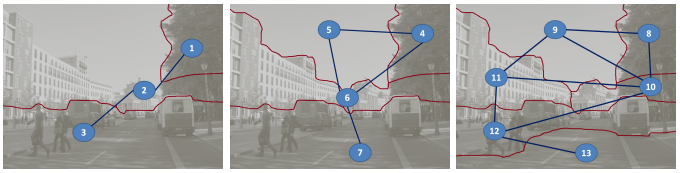
6.2 Related Work

There are many recent works on contextual models that exploit the spatial structures in the image. Meanwhile, the use of multiple different over-segmented images as a preprocessing step is not new to computer vision. For example, multiple over-segmentations for finding objects in the images is used in Russell *et al.* (2006). In the context of multi-class image classification, the

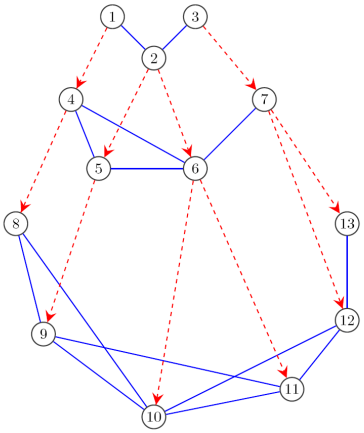
work of Plath *et al.* (2009) comprises two aspects for coupling local and global evidences both by constructing a tree-structured CRF on image regions on multiple scales and using global image classification information. Thereby, Plath *et al.* (2009) neglect direct local neighborhood dependencies. The work of Schnitzspan *et al.* (2008) extends classical one-layer CRF to a multi-layer CRF by restricting the pairwise potentials to a regular 4-neighborhood model and introducing higher-order potentials between different layers.

Although not as popular as CRFs, BNs have also been used to solve computer vision problems (Mortensen & Jia, 2006; Sarkar & Boyer, 1993). BNs provide a systematic way to model the causal relationships among the entities. By explicitly exploiting the conditional independence relationships (known as prior knowledge) encoded in the structure, BNs could simplify the modelling of joint probability distributions. Based on the BN structure, the joint probability is decomposed into the product of a set of local conditional probabilities, which is much easier to specify because of their semantic meanings (Zhang & Ji, 2010).

Graphical models have reached a state where both hierarchical and spatial neighborhood structures can be efficiently handled. RFs and BNs are suitable for representing different types of statistical relationships among the random variables. Yet only a few previous works focus on integrating RFs with BNs. In Kumar & Hebert (2003b), the authors present a generative model based approach to man-made structure detection in 2D natural images. They use a causal multiscale random field as a prior model on the class labels. Labels over an image are generated using Markov chains defined over coarse to fine scales. However, the spatial neighborhood relationships are only considered at the bottom scale. So, essentially, this model is a tree-structured belief network plus a flat Markov random field. In Liu *et al.* (2006), the authors propose an integration of a BN with an MRF for image segmentation. A naive Bayes model is used to transform the image features into a probability map in the image domain. The MRF enforces the spatial relationships of the labels. The use of a naive Bayes model greatly limits the capability of this method. Recently, a unified graphical model that can represent both the causal and noncausal relationships among the random variables is proposed in Zhang & Ji (2010). They first employ a CRF to model the spatial relationships among the image regions and their measurements. Then, they introduce a multilayer BN to model the causal dependencies. The CRF model and the BN model are then combined through the theories of the factor graphs to form a unified probabilistic graphical model. Their graphical model is too complex in general. Although their model improves state of the art results on the Weizmann horse dataset and the MSRC dataset, they need a lot of domain expert knowledge to design the local constraints. Also, they use a combination of supervised parameter learning and manual parameter setting for the model parameterization. Simultaneously learn the BN and CRF parameters automatically from the training data is not a trivial task. Compared to the graphical models in Kumar & Hebert (2003b) and Liu *et al.* (2006), which are too simple, the graphical models in Zhang & Ji (2010) are too complex in general. Our graphical model lies in between, cf. Figure 6.1. We try to construct our graphical model that is not too simple in order to model the rich relationships among the neighborhood of pixels and image regions in the scene, yet not too complex in order to make parameter learning and probabilistic inference efficiently. Furthermore, our model underlies a clear semantic meaning. If the undirected edges are ignored, meaning no spatial relationships are considered, the graph is a tree representing the hierarchy



(a) Multi-scale segmentation



(b) The graphical model

Figure 6.1: Illustration of the graphical model architecture. The blue edges between the nodes represent the neighborhoods at one scale (undirected edges), and the red dashed edges represent the hierarchical relation between regions (undirected or directed edges).

of the partonomy among the scales. Within each scale, the spatial regions are connected by the pairwise edges.

6.3 Model

6.3.1 The Graphical Model Construction

By constructing the graphical model, we can flexibly choose either directed edges or undirected edges to model the relationships between the random variables based on the semantic meaning of these relationships.

We use an example image to explain this model construction process. Given a test image, Figure 6.1 shows the corresponding multi-scale segmentation of the image, and the corresponding graphical model for image interpretation. Three layers are connected via a region hierarchy (Drauschke & Förstner, 2011). The development of the regions over several scales is used to model the region hierarchy. Furthermore, the relation is defined over the maximal overlap of the regions. Nodes connection and numbers correspond to the multi-scale segmentation. The pairwise interactions between the spatial neighboring regions can be modeled by the undirected edges. The pairwise potential functions can be defined to capture the similarity between the

neighboring regions. The hierarchical relation between regions of the scene partonomy representing parent-child relations or part-of relations can be modeled by either the undirected edges or the directed edges.

6.3.2 Multi-class Labeling Representation

We present the scene interpretation problem as a multi-class labeling problem. Given the observed data $\boldsymbol{d}$, the distribution P over a set of the variables $\boldsymbol{x}$ can be expressed as a product of the factors

$$P(\boldsymbol{x} \mid \boldsymbol{d}) = \frac{1}{Z} \prod_{i \in \mathcal{V}} \boldsymbol{f}_i(\boldsymbol{x}_i \mid \boldsymbol{d}) \prod_{\{i,j\} \in \mathcal{E}} \boldsymbol{f}_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j \mid \boldsymbol{d}) \prod_{\langle i,k \rangle \in \mathcal{S}} \boldsymbol{f}_{ik}(\boldsymbol{x}_i, \boldsymbol{x}_k \mid \boldsymbol{d}) \tag{6.1}$$

where the factors $\boldsymbol{f}_i, \boldsymbol{f}_{ij}, \boldsymbol{f}_{ik}$ are the functions of the corresponding sets of the nodes, and Z is the normalization factor. The set $\mathcal{V}$ is the set of the nodes in the complete graph, and the set $\mathcal{E}$ is the set of pairs collecting the neighboring nodes within each scale. $\mathcal{S}$ is the set of pairs collecting the parent-child relations between regions with the neighboring scales, where $\langle i, k \rangle$ denotes nodes i and k are connected by either a undirected edge or a directed edge. Note that this model only exploits up to second-order cliques, which makes learning and inference much faster than the model involving high-order cliques.

By simple algebra calculation, the probability distribution given in Eq. (6.1) can be written in the form of a *Gibbs* distribution

$$P(\boldsymbol{x} \mid \boldsymbol{d}) = \frac{1}{Z} \exp\left(-E(\boldsymbol{x} \mid \boldsymbol{d})\right) \tag{6.2}$$

with the energy function $E(\boldsymbol{x} \mid \boldsymbol{d})$ as

$$E(\boldsymbol{x} \mid \boldsymbol{d}) = \sum_{i \in \mathcal{V}} E_1(\boldsymbol{x}_i \mid \boldsymbol{d}) + \alpha \sum_{\{i,j\} \in \mathcal{E}} E_2(\boldsymbol{x}_i, \boldsymbol{x}_j \mid \boldsymbol{d}) + \beta \sum_{\langle i,k \rangle \in \mathcal{S}} E_3(\boldsymbol{x}_i, \boldsymbol{x}_k \mid \boldsymbol{d}) \tag{6.3}$$

where α and β are the weighting coefficients in the model. E_1 is the unary potential, E_2 is the pairwise potential, and E_3 is either the hierarchical pairwise potential or the conditional probability energy. This graphical model is illustrated in Figure 6.1. The most probable or maximum a posteriori (MAP) labeling $\boldsymbol{x}^*$ is defined as

$$\boldsymbol{x}^* = \arg\max_{\boldsymbol{x} \in \mathcal{L}^n} P(\boldsymbol{x} \mid \boldsymbol{d}) \tag{6.4}$$

and can be found by minimizing the energy function $E(\boldsymbol{x} \mid \boldsymbol{d})$.

6.4 Relation to Previous Models

In this section, we draw comparisons with the previous models for image interpretation (Drauschke & Förstner, 2011; Fulkerson *et al.*, 2009; Plath *et al.*, 2009; Yang *et al.*, 2010) and show that at certain choices of the parameters of our framework, these methods fall out as the special cases. We will now show that our model is not only a generalization of the standard flat CRF over the image regions, but also of the hierarchical CRF and the conditional Bayesian network.

6.4.1 Equivalence to Flat CRFs over Regions

Let us consider the case with only one layer segmentation of the image (the bottom layer of the graphical model in Figure 6.1). In this case, the weight β is set to be zero, the set $\mathcal{V}^1$ is the set of nodes in the graph of the bottom layer, and the set $\mathcal{E}^1$ is the set of pairs collecting the neighboring nodes in the bottom layer. This allows us to rewrite Eq. (6.3) as

$$E(\boldsymbol{x} \mid \boldsymbol{d}) = \sum_{i \in \mathcal{V}^1} E_1(\boldsymbol{x}_i \mid \boldsymbol{d}) + \alpha \sum_{\{i,j\} \in \mathcal{E}^1} E_2(\boldsymbol{x}_i, \boldsymbol{x}_j \mid \boldsymbol{d}) \tag{6.5}$$

which is exactly the same as the energy function associated with the flat CRF defined over the image regions with E_1 as the unary potential and E_2 as the pairwise potential. In this case, our model becomes equivalent to the flat CRF models defined over the image regions (Fulkerson *et al.*, 2009; Gould *et al.*, 2008).

6.4.2 Equivalence to Hierarchical CRFs

Let us now consider the case with the multi-scale segmentation of the image. If we choose E_3 as a pairwise potential in Eq. (6.3), the energy function reads

$$\begin{aligned} E(\boldsymbol{x} \mid \boldsymbol{d}) = & \sum_{i \in \mathcal{V}} E_1(\boldsymbol{x}_i \mid \boldsymbol{d}) + \alpha \sum_{\{i,j\} \in \mathcal{E}} E_2(\boldsymbol{x}_i, \boldsymbol{x}_j \mid \boldsymbol{d}) \\ & + \beta \sum_{\{i,k\} \in \mathcal{S}} E_3(\boldsymbol{x}_i, \boldsymbol{x}_k \mid \boldsymbol{d}) \end{aligned} \tag{6.6}$$

which is exactly the same as the energy function associated with the hierarchical CRF defined over the multi-scale of the image regions with E_1 as the unary potential, E_2 as the pairwise potential within each scale, and E_3 as the hierarchical pairwise potential with the neighboring scales. In this case, our model becomes equivalent to the hierarchical CRF models defined over multi-scale of image regions (He *et al.*, 2004; Yang *et al.*, 2010).

If we set α to be zero, and choose E_3 as a pairwise potential in Eq. (6.3), the energy function reads

$$E(\boldsymbol{x} \mid \boldsymbol{d}) = \sum_{i \in \mathcal{V}} E_1(\boldsymbol{x}_i \mid \boldsymbol{d}) + \beta \sum_{\{i,k\} \in \mathcal{S}} E_3(\boldsymbol{x}_i, \boldsymbol{x}_k \mid \boldsymbol{d}) \tag{6.7}$$

which is the same as the energy function associated with the tree-structured CRF by neglecting the direct local neighborhood dependencies on the image regions on multiple scales. In this

case, our model becomes equivalent to the tree-structured CRF models defined over multi-scale of the image regions (Plath *et al.*, 2009; Reynolds & Murphy, 2007).

6.4.3 Equivalence to Conditional Bayesian Networks

If we set α to be zero, and choose E_3 as the conditional probability energy in Eq. (6.3), the energy function reads

$$E(\boldsymbol{x} \mid \boldsymbol{d}) = \sum_{i \in \mathcal{V}} E_1(\boldsymbol{x}_i \mid \boldsymbol{d}) + \beta \sum_{\langle i,k \rangle \in \mathcal{S}} E_3(\boldsymbol{x}_i, \boldsymbol{x}_k \mid \boldsymbol{d}) \tag{6.8}$$

which is the same as the energy function associated with the tree-structured conditional Bayesian network defined over the multi-scale of the image regions. In the tree-structured conditional Bayesian network, the classification of a region is based on the unary features derived from the region and the binary features derived from the relations of the region hierarchy graph. In this case, our model becomes equivalent to the tree-structured conditional Bayesian network defined over multi-scale of the image regions (Drauschke & Förstner, 2011).

6.5 Experiments

We conduct the experiments to evaluate the performance of the proposed model on eTRIMS dataset (Korč & Förstner, 2009). The dataset consists of 60 building facade images, labeled with 8 classes: *building*, *car*, *door*, *pavement*, *road*, *sky*, *vegetation*, *window*. We randomly divide the images into a training set with 40 images and a testing set with 20 images. In all experiments, we take the ground truth label of a region to be the majority vote of the ground truth pixel labels. At the test stage we compute our accuracy at the pixel level.

The hierarchical mixed graphical model is defined over the multi-scale of the image regions when we choose E_3 as the conditional probability energy in Eq. (6.3). We present the experimental results for the hierarchical mixed graphical model with multi-scale mean shift segmentation (Comaniciu & Meer, 2002) and watershed segmentation (Vincent & Soille, 1991), and the comparison with the baseline region classifier, the flat CRF, and the hierarchical CRF classification results.

6.5.1 Results with Multi-scale Mean Shift and the Hierarchical Mixed Graphical Model

The overall classification accuracy is 68.9%. The weighting parameters are $\alpha = 0.8$, $\beta = 1$. For comparison, the RDF region classifier gives an overall accuracy of 58.8%, the flat CRF gives an overall accuracy of 65.8%, and the hierarchical CRF gives an overall accuracy of 69.0%.

Qualitative results of the hierarchical mixed graphical model with the multi-scale mean shift on the eTRIMS dataset (Korč & Förstner, 2009) are presented in Figure 6.2. The qualitative inspection of the results in these images shows that the hierarchical mixed graphical model yields significant improvement. The hierarchical mixed graphical model yields more

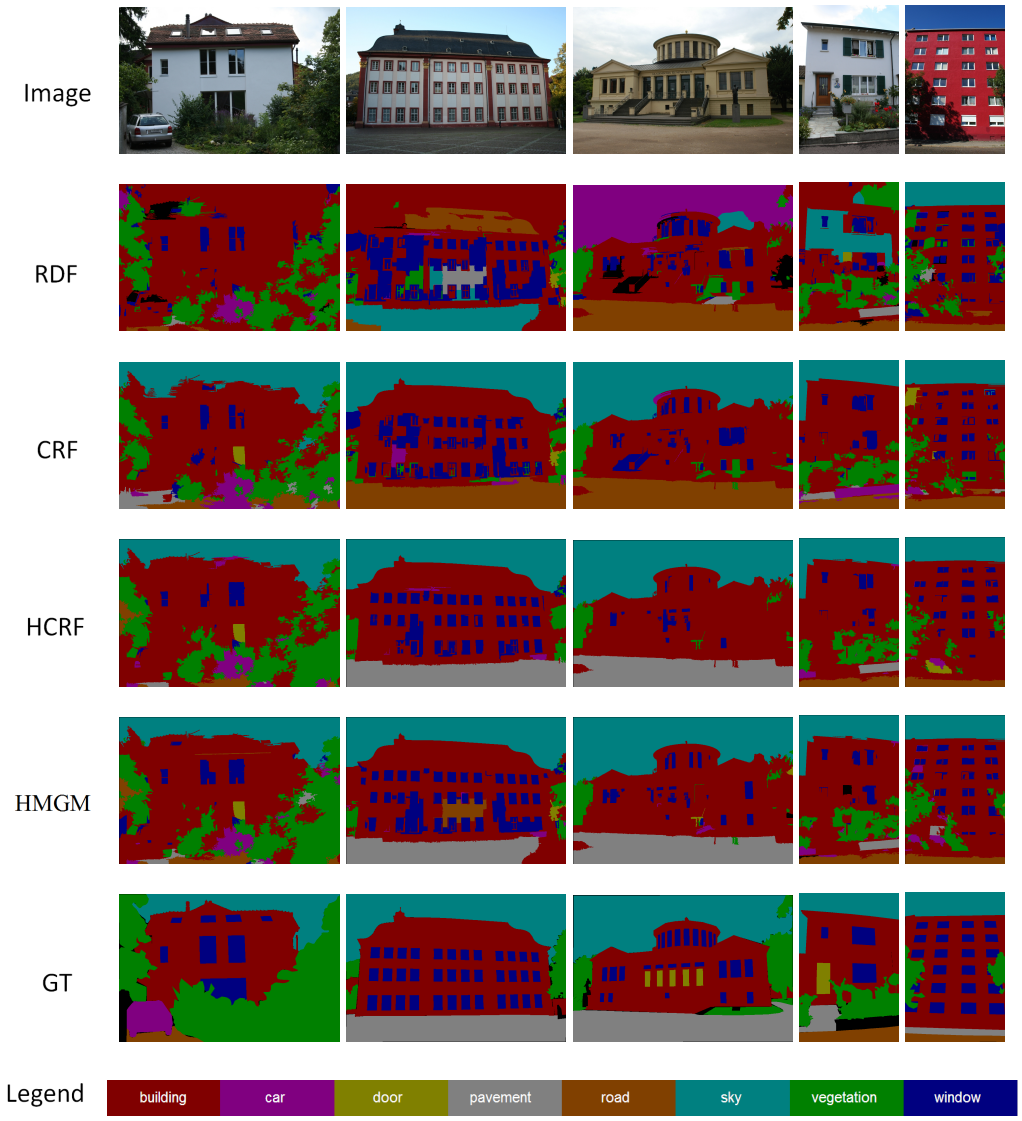


Figure 6.2: Qualitative classification results of the hierarchical mixed graphical model with the multi-scale mean shift segmentation on the testing images from the eTRIMS dataset (Korč & Förstner, 2009).

accurate and cleaner results than the flat CRF and the RDF region classifier, and comparable to the hierarchical CRF model. The greatest accuracies are for classes which have low visual variability and many training examples (such as window, vegetation, building, and sky) whilst the lowest accuracies are for classes with high visual variability or few training examples (for example door, car, and pavement). We expect more training data and the use of features with better invariance properties will improve the classification accuracy. Objects such as car, door, pavement, and window are sometimes incorrectly classified as *building*, due to the dominant presence of the building in the image. Detecting windows, cars, and doors should resolve some of such ambiguities.

6.5.2 Results with Multi-scale Watershed and the Hierarchical Mixed Graphical Model

The overall classification accuracy is 68.0%. The weighting parameters are $\alpha = 1.08$, $\beta = 1$. For comparison, the RDF region classifier gives an overall accuracy of 55.4%, the flat CRF gives an overall accuracy of 61.8%, and the hierarchical CRF gives an overall accuracy of 65.3%. Qualitative results of the hierarchical mixed graphical model on the eTRIMS dataset are presented in Figure 6.3.

6.6 Conclusion

In this work, we have addressed the problem of incorporating two different types of the contextual information, namely the spatial structure and the hierarchical structure for image interpretation of man-made scenes. We propose a statistically motivated, generic probabilistic graphical model framework for scene interpretation, which seamlessly integrates different types of the image features, and the spatial structural information and the hierarchical structural information defined over the multi-scale image segmentation. We demonstrate the application of the proposed model on the building facade image classification task.

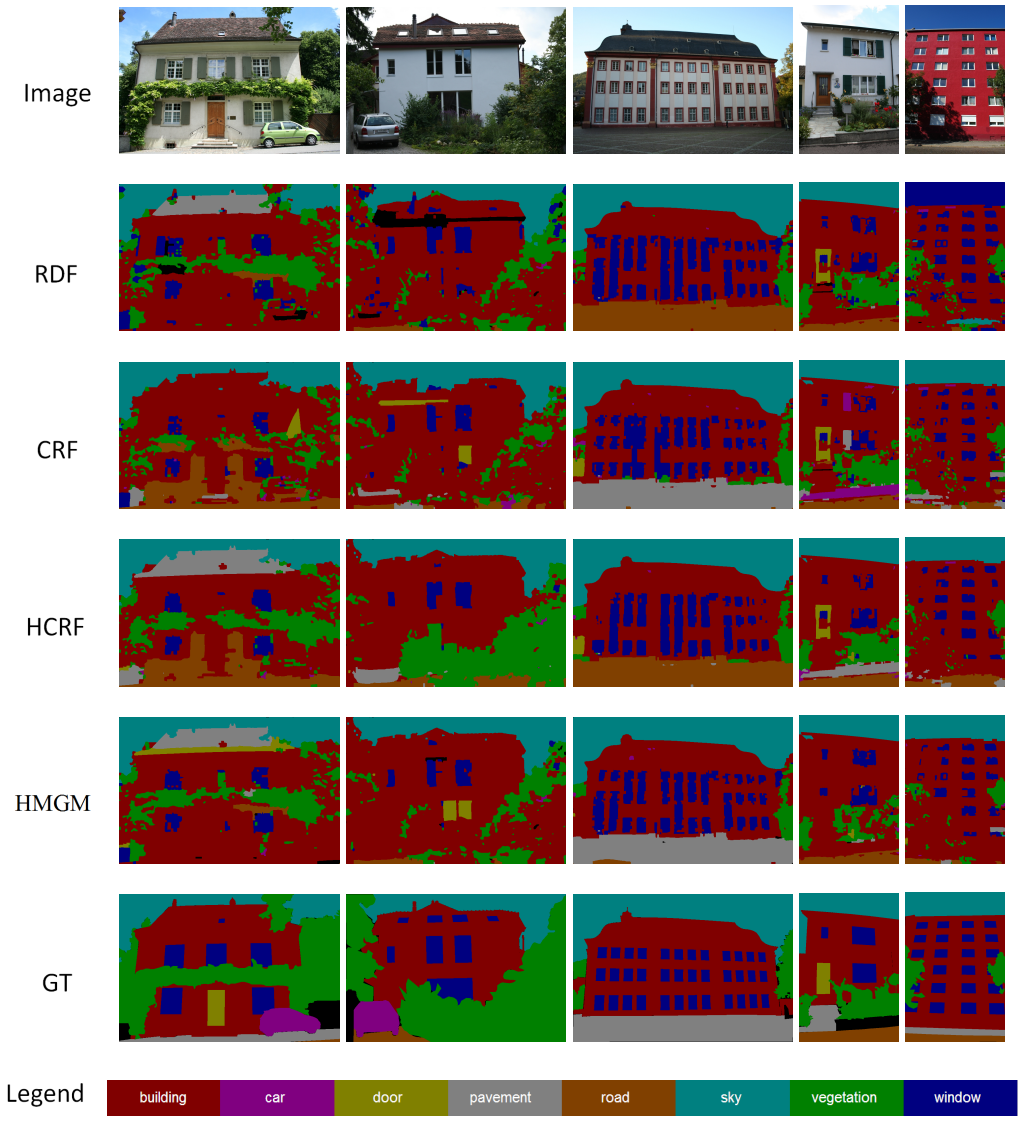


Figure 6.3: Qualitative classification results of the hierarchical mixed graphical model with the multi-scale watershed segmentation on the testing images from the eTRIMS dataset (Korč & Förstner, 2009).

Chapter 7

Video Segmentation with Joint Object and Trajectory Labeling

Unsupervised video object segmentation is a challenging problem because it involves a large amount of data and object appearance may significantly change over time. In this chapter, we propose a bottom-up approach for the combination of object segmentation and motion segmentation using a novel graphical model, which is formulated as inference in a conditional random field (CRF) model. This model combines object labeling and trajectory clustering in a unified probabilistic framework. The CRF contains binary variables representing the class labels of image pixels as well as binary variables indicating the correctness of trajectory clustering, which integrates dense local interaction and sparse global constraint. An optimization scheme based on a coordinate ascent style procedure is proposed to solve the inference problem. We evaluate our proposed framework by comparing it to other video and motion segmentation algorithms. Our method achieves improved performance on state-of-the-art benchmark datasets. An earlier version of this chapter appeared at the IEEE Winter Conference on Applications of Computer Vision (WACV) (Yang & Rosenhahn, 2014).

7.1 Introduction

One of the great challenges in computer vision is automatic segmentation of complex dynamic content in videos, so called object segmentation, which is to produce a binary segmentation, separating foreground objects from their background in an unannotated video. This is a challenging task, as local image measurements often provide only a weak cue. Object appearance may significantly change over the frames of the video due to changes in the camera viewpoint, scene illumination or object deformation. In general, segmentation must capture both short range correlations (within a frame and between successive frames) and long range correlations (across many frames) in the video. Object segmentation is the basis for many potential applications including object tracking, object recognition, 3D reconstruction, activity recognition, and video retrieval. Due to its potential applications, there is increasing number of works (Grundmann *et al.*, 2010; Lee *et al.*, 2011) addressing the problem of video object segmentation in recent years. Many approaches extend single image segmentation techniques to

multiple frames, exploiting the fact that there is redundancy along the time axis and that the motion field is smooth. The problems associated with these methods include drift, occlusion, and appearance adaption. Integrating long-term cues in the segmentation process might help solve these problems. In fact, video provides rich additional cues beyond a single image. These cues include object motion, temporal continuity, and long-range temporal object interactions, etc. Motion segmentation exploits these cues, which formulates clustering objectives to group pixels from all frames. However, motion segmentation results are only in discrete and sparse positions available (Brox & Malik, 2010).

We overcome aforementioned problems by merging image segmentation and motion segmentation. We propose a method to obtain a spatio-temporal foreground segmentation of a video that respects object boundaries, as shown in Figure 7.1, and at the same time perform trajectory labeling. Different from previous approaches, we address the foreground segmen-

Figure 7.1: Video object segmentation. Input: unannotated video. Output: Foreground object in each frame.

tation by partitioning frames using a novel graphical model on pixel level, which is dense in spatial domain, yet sparse in temporal domain. We formulate the problem as inference in a conditional random field (CRF). We make use of point trajectories, which have rich grouping information in their motion differences. The CRF contains binary variables representing the class labels of image pixels as well as binary variables indicating the correctness of trajectory clustering. Joint object and trajectory segmentation is formulated as a pixel and trajectory labeling problem of assigning each pixel and trajectory with either foreground or background. An overview of our proposed method is given in Figure 7.2.

Contributions Our main contribution is a fully automatic and unsupervised bottom-up approach for the combination of object segmentation and motion segmentation, which is for-

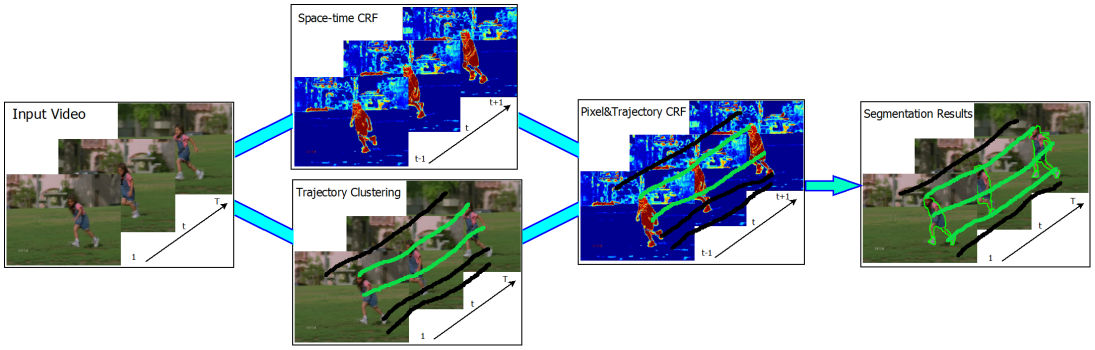


Figure 7.2: Video segmentation overview. Input: unannotated video. Output: Foreground object segments for all frames (the green boundary overlays with each frame for visualization), and trajectory labeling results. We optimize over pixels and trajectories in the joint space via a space-time CRF: both foreground estimation and trajectory clustering are modeled as energy potentials in the model. Here, the black trajectories are classified as background while the green ones are foreground.

mulated as inference in a unified CRF model. The CRF contains pixel labeling and trajectory clustering in a single energy function, which integrates dense local interaction and sparse global constraints. We optimize over pixels and trajectories in the joint space via a space-time CRF: both foreground estimation and trajectory clustering are modeled as energy potentials. An optimization scheme based on a coordinate ascent style procedure is proposed to solve the inference problem. To the best of our knowledge, this work is the first one to combine object labeling and trajectory clustering in a unified probabilistic framework.

The following sections are organized as follows. The related works are discussed in Section 7.2. Section 7.3 introduces the CRF model for video segmentation and the trajectory clustering. Our proposed approach is described in detail in Section 7.4. In Section 7.5, experimental results are presented. Finally, this work is concluded and future work is discussed in Section 7.6.

7.2 Related Work

Video object segmentation is often performed in an interactive or supervised manner. Interactive methods require a user to perform object boundary annotation in some key frames, which are then propagated to other frames (Price *et al.*, 2009; Vijayanarasimhan & Grauman, 2012; Yuen *et al.*, 2009). Tracking-based methods attempt to reduce the supervision to a manual segmentation on only the first frame (Chockalingam *et al.*, 2009; Ren & Malik, 2007; Tsai *et al.*, 2012). However, all such methods demand user input of drawing regions of interest, therefore not fully automatic, and may suffer from sensitivity to a user's annotation experience.

On the other hand, bottom-up approaches can segment videos in a fully automatic manner, based on cues like motion and appearance. Motion segmentation methods cluster pixels in video using bottom-up motion cues. Recent methods perform pixel-level segmentation in

a spatio-temporal video volume from scratch (Grundmann *et al.*, 2010), begin with an image segmentation per frame and then match segments across nearby frames (Reina *et al.*, 2010). Without any top-down notion of objects, however, such methods tend to over-segment, yielding regions that may lack semantic meaning. Brendel & Todorovic (2009) attempt to segment objects in video by tracking and splitting/merging image regions. Reina *et al.* (2010) extract multiple segmentation hypotheses in each frame, and then search for a segmentation consistent over multiple frames. Spatio-temporal segmentation of video sequences into segments with coherent local properties has been also addressed by graph-based approaches (Grundmann *et al.*, 2010). However, these methods are limited by the analysis performed at a local level. Lee *et al.* (2011) first discover key-segments and group them to predict the foreground objects in a video. Ma & Latecki (2012) introduce maximum weight cliques with mutex constraints in the region graph to obtain reliable segmentations of foreground object. In this work, we also conduct graph-based segmentation. But additionally, we incorporate long-range motion cues into the segmentation.

Similar to video segmentation, grouping point trajectories in video sequences based on independent motions, so called motion segmentation, has received significant attention. Recently, impressive results in grouping point trajectories were shown by Brox & Malik (2010) who carefully analyze motion differences between pairs of tracks and cluster the resulting affinity matrix using normalized cuts (Shi & Malik, 2000). These sparse trajectory clusters are used in Ochs & Brox (2011) to obtain dense object segmentation. Strong shape priors are derived from a multi-level super-pixel segmentation (Arbelaez *et al.*, 2011), which preserve the main borders between objects. Super-pixels are labeled and merged using the motion segmentation tracks and a multi-level variational approach. A tracking framework for segmenting objects in crowded scenes is proposed in Fragkiadaki *et al.* (2012a), which mediates grouping cues from two levels of tracking granularities, detection tracklets and point trajectories. Fragkiadaki *et al.* (2012b) propose detecting discontinuities of embedding density between spatially neighboring trajectories. Then Gabriel graph is used for converting trajectory clustering to dense image segmentation. Dragon *et al.* (2012) present an approach for motion segmentation using multi-scale clustering of frame-to-frame keypoint correspondences instead of trajectories. Another class of spatio-temporal techniques take advantage of all the frames in a video. They treat the video as a 3D space-time volume (Klein *et al.*, 2002; Rubio *et al.*, 2012). Such large amount of data usually results in expensive computational time. Instead of processing all the frames simultaneously, we make use of point trajectories to segment the successive frames, which all together is dense in space, yet sparse in time. As will be shown in this work, video segmentation benefits from motion segmentation, and vice versa.

7.3 Preliminaries

We begin by describing the CRF model for video segmentation. We then introduce the clustering technique for point trajectories.

7.3.1 Video Object Segmentation

Given a video sequence $\boldsymbol{I} = \{I_t\}$, we formulate video segmentation as a pixel labeling problem of assigning each pixel in frame I_t with either foreground or background. Consider a set of the random variables $\{X_i, i \in \mathcal{V}\}$ defined over an undirected graph $\mathcal{H} = (\mathcal{V}, \mathcal{E})$, where X_i is associated with a node $i \in \mathcal{V} = \{1, \ldots, n\}$. The CRF is defined over $\mathcal{H}$, so that each node i corresponds to a pixel p_i and an edge between two nodes corresponds to the cost of a cut between two pixels. Let $\boldsymbol{x} = \{x_i\}$ denote the labeling of the CRF which refers to any possible assignment of labels to the random variables, and takes values from the set $\boldsymbol{L} = \{0, 1\}^n$, where 0 corresponds to background and 1 corresponds to foreground. Its energy function $E(\boldsymbol{x})$ can be written as

$$E(\boldsymbol{x}) = \sum_{i \in \mathcal{V}} \phi_i(x_i) + \alpha \sum_{\{i,j\} \in \mathcal{E}} \phi_{ij}(x_i, x_j) \tag{7.1}$$

where ϕ_i and ϕ_{ij} are the unary and pairwise potentials respectively, which both depend on the observed data $\boldsymbol{I}$. α is the weighting coefficient in the model. The edge set $\mathcal{E}$ is commonly chosen to define a 6 neighborhood (Lee *et al.*, 2011; Reina *et al.*, 2010), which consists of 4 spatially neighboring pixels in the same frame, and two temporally neighboring pixels in adjacent frames. We assign a pixel's temporal neighbor in the next frame by its optical flow vector displacement (Brox & Malik, 2011). This energy function, Eq. (7.1), encourages spatial homogeneity of contrast within each frame and temporal consistency between frames.

The most probable or MAP labeling $\boldsymbol{x}^*$ of the random field can be found by minimizing the energy function $\boldsymbol{x}^* = \arg\min_{\boldsymbol{x} \in \boldsymbol{L}} E(\boldsymbol{x})$. While the exact minimization is generally intractable on general CRF, a good approximation can be found efficiently using graph cut based methods (Boykov *et al.*, 2001) or belief propagation (Murphy *et al.*, 1999).

7.3.2 Trajectory Clustering

Long-term motion can provide strong low-level cues for many vision tasks. For example, two static objects can be separated based on their past or future independent motion if this motion evidence is propagated over time. Video segmentation approaches segment objects following the Gestalt principle of common fate, often enhanced by large temporal context of point trajectories. We define a trajectory tr_r to be a sequence of space-time points: $\text{tr}_r = \left\{(lx_r^t, ly_r^t), t \in T_r\right\}$, where T_r is the frame span of tr_r, and (lx, ly) is the pixel location. We obtain point trajectory by tracking pixels across frames using the optical flow (Brox & Malik, 2011). Point trajectories are dense in space and can have various lengths.

Trajectories have rich grouping information in their motion differences. We define pairwise affinities between all trajectories that share at least one frame, yielding the affinity matrix W for the whole sequence. We set affinities $W(\text{tr}_r, \text{tr}_s)$ between trajectories tr_r and tr_s according to the maximum velocity difference v_{rs} computed during their time overlap

$$W(\text{tr}_r, \text{tr}_s) = \exp[-\text{dst}_{rs}(d_{sp}\frac{v_{rs}^2}{\sigma_v^2})] \tag{7.2}$$

where dst_{rs} denotes the maximum Euclidean distance between tr_r and tr_s, and σ_v is the normal-

ization factor. Penalizing maximum velocity difference takes advantage of the most informative frames in the time overlap between tr_r and tr_s (Brox & Malik, 2010). d_{sp} denotes the average spatial Euclidean distance of tr_r and tr_s in the common time window. Multiplying with the spatial distance ensures that only proximate points can generate high affinities. We then classify trajectories as foreground or background by performing spectral clustering on the affinity matrix W (Brox & Malik, 2010). An example is shown in Figure 7.3.

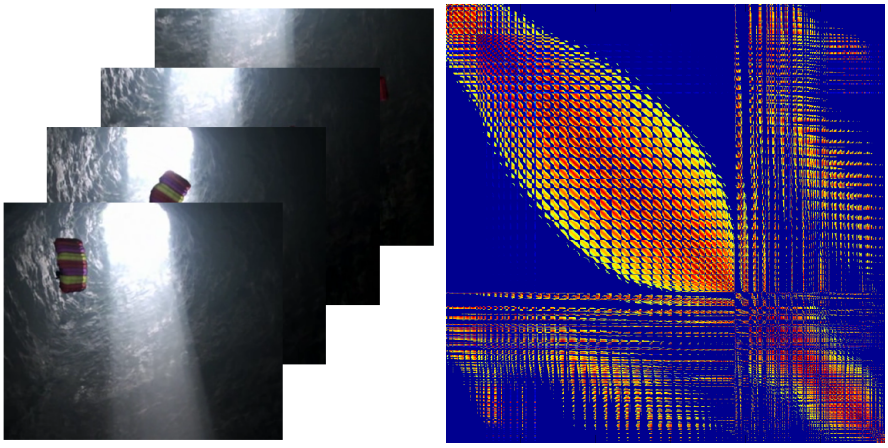


Figure 7.3: *Left*: an example video sequence. *Right*: corresponding affinity matrix W.

7.4 Joint object and trajectory segmentation

In this section, we describe our approach to video segmentation. We formulate the problem as inference in a CRF. The random field contains binary variables representing the class labels of image pixels as well as binary variables indicating the correctness of trajectory clustering. The illustration in Figure 7.2 gives an overview of our model.

7.4.1 Formulation

Joint object and trajectory segmentation is formulated as a pixel and trajectory labeling problem of assigning each pixel and trajectory with either foreground or background. Formally, let $x_i \in \{0, 1\}$ be a random variable representing the class label of the i-th pixel, while $y_r \in \{0, 1\}$ is a random variable associated with the class label of the r-th trajectory. Similar to Eq. (7.1), the total energy function $E(\boldsymbol{x}, \boldsymbol{y})$ for joint segmentation can be written as

$$\begin{aligned} E(\boldsymbol{x}, \boldsymbol{y}) = &\sum_{i \in \mathcal{V}} \phi_i(x_i) + \alpha \sum_{\{i,j\} \in \mathcal{E}} \phi_{ij}(x_i, x_j) \\ &+ \beta \sum_{\{i,r\} \in \eta} \phi_{ir}(x_i, y_r) + \gamma \sum_{\{r,s\} \in \delta} \phi_{rs}(y_r, y_s) \end{aligned} \tag{7.3}$$

where $\mathcal{V}$ and $\mathcal{E}$ are the sets of nodes and edges in the video frames respectively. η contains all pixel and trajectory pairs that are in correspondence, while the set δ contains all the pairs of trajectories. α, β, γ are the weighting coefficients in the model. ϕ_i is the unary potential encoding the likelihood of pixels belonging to foreground or background. ϕ_{ij} is the pairwise potential, which enforces spatial and temporal consistency between pixels. ϕ_{ir} is the pixel-trajectory compatibility potential, which ensures the corresponding pixel and trajectory take the same label. ϕ_{rs} is the trajectory clustering potential, which encourages foreground and background separation between trajectories. The formulation of these terms will be presented in the remainder of this section.

7.4.2 Potentials

Unary potentials The unary potential $\phi_i(x_i)$ independently predicts the label x_i based on the frame I_t. The label distribution $\phi_i(x_i)$ is usually calculated by using a classifier. In this work, we use the Gaussian mixture model (GMM) (i.e. Boykov-Jolly model (Boykov & Jolly, 2001; Rother *et al.*, 2004)). GMM is a popular appearance model in object segmentation (Batra *et al.*, 2011; Greenspan *et al.*, 2004). The GMM distributions are constructed with a set of simple features, which is a set of pixel colors. Assume a Gaussian mixture with C components, the parameters $\boldsymbol{\theta} = \left\{\pi_c^f, \mu_c^f, \sigma_c^f, \pi_c^b, \mu_c^b, \sigma_c^b\right\}_{c=1}^{C}$ are the prior probability, mean, and covariance of the model. Foreground and background trajectories are used for learning these parameters. We set $\phi_i(x_i)$ to be the pixel likelihoods computed from the learned GMM. A pixel that has similar color to the foreground object will have high cost if labeled as background.

Pairwise potentials In segmentation algorithms, spatial and temporal consistencies are usually enforced using pairwise terms based on color difference (Lee *et al.*, 2011; Rother *et al.*, 2004). ϕ_{ij} is modeled by a standard contrast-dependent function defined in Boykov & Jolly (2001); Rother *et al.* (2004), which favors assigning the same label to neighboring pixels with similar color. The edge set $\mathcal{E}$ consists of 4 spatially neighboring pixels in the same frame, and two temporally neighboring pixels in adjacent frames.

Pixel-trajectory compatibility potentials We introduce this pixel-trajectory compatibility term, which imposes a penalty on corresponding pixel and trajectory with different labels. It can be written as

$$\phi_{ir} = 1 - \delta(x_i, y_r) \tag{7.4}$$

The corresponding pixel and trajectory pair is determined by whether pixel p_i belongs to tr_r, which defines the set η.

Trajectory clustering potentials We define the trajectory clustering potentials ϕ_{rs} between two trajectories tr_r, tr_s as

$$\phi_{rs}(y_r, y_s) = y_r y_s L_{rs} \tag{7.5}$$

where L is the Laplacian matrix $L = H^{-1/2} W H^{-1/2}$ (Shi & Malik, 2000). W is the affinity matrix for trajectories defined in Section 7.3.2. H is the diagonal matrix composed of the

row sums of W. This term encourages coherent labeling of trajectories. This is equivalent to spectral clustering for all the trajectories in the sequence. Spectral clustering captures essential cluster structure of a graph using the spectrum of the graph Laplacian matrix (Shi & Malik, 2000).

7.4.3 Optimization

The video segmentation problem can be solved by finding the least energy configuration of the CRF defined in Eq. (7.3). In general, exact minimization of the energy function E is NP-hard. It is instead solved using approximate algorithms. In our case, minimizing the complex energy function given in Eq. (7.3), which involves two sets of random variables, is also difficult to approximate. In this work, we present an optimization scheme based on a coordinate ascent style procedure, alternating between minimizing $E(\boldsymbol{x}, \boldsymbol{y})$ with respect to $\boldsymbol{x}$ for fixed $\boldsymbol{y}$ (1-step) and with respect to $\boldsymbol{y}$ for fixed $\boldsymbol{x}$ (2-step). Convergence to a strong local optimum is usually achieved in 3-4 cycles of iterations. The algorithm is initialized by GMM for pixel labeling and trajectory clustering for trajectory labeling.

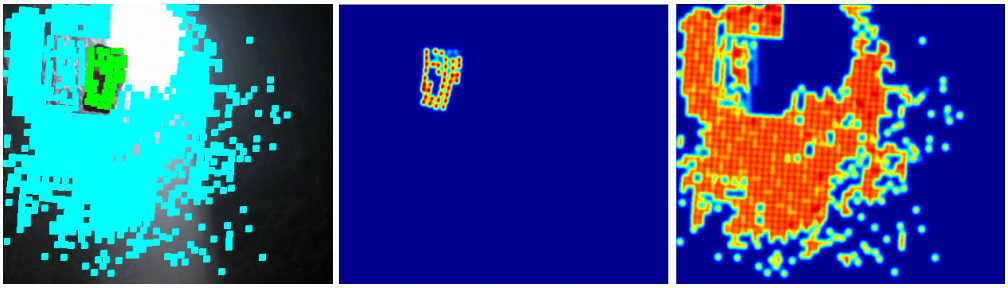


Figure 7.4: Shape-location prior likelihood. *Left*: sparse label from trajectory clustering (Brox & Malik, 2010), *Middle*: foreground confidence map, *Right*: background confidence map.

1-step For a given binary trajectory labeling $\widehat{\boldsymbol{y}}$, minimizing the total energy function $E(\boldsymbol{x}, \boldsymbol{y})$ in terms of $\boldsymbol{x}$ leads to

$$\min_{\boldsymbol{x}} E(\boldsymbol{x}, \widehat{\boldsymbol{y}}) = \sum_{i \in \mathcal{V}} \phi_i(x_i) + \alpha \sum_{\{i,j\} \in \mathcal{E}} \phi_{ij}(x_i, x_j) + \beta \sum_{\{i,r\} \in \eta} \phi_{ir}(x_i, \widehat{y}_r) \tag{7.6}$$

When a trajectory labeling is given, the trajectory clustering potentials become constant, and therefore do not affect energy minimization. Furthermore, pixel-trajectory compatibility potentials can effectively be merged to unary potentials. As the pairwise potentials of the energy function in Eq. (7.6) is a Potts model, it can be minimized using graph cuts (Boykov & Kolmogorov, 2004; Boykov *et al.*, 2001).

In order to be robust to outliers that may occur due to trajectory clustering errors, we map sparse trajectory points to dense shape-location priors in the pixel-trajectory compatibility

potentials. An estimate of the shape, location and scale of the foreground is computed in every frame using a kernel density estimation (KDE) (Hwang *et al.*, 1994) based on the sparse foreground points output by the binary trajectory labeling (Ellis & Zografos, 2012). The 2D spatial distribution is estimated from the sparse points labeled as foreground (background). The KDE for the object is defined as

$$\hat{f}_h(\boldsymbol{l}) = \frac{1}{\Omega} \sum_{k \in \Omega} K_h(\boldsymbol{l} - \boldsymbol{l}_k) \tag{7.7}$$

where $\boldsymbol{l}_k$ is the pixel location, Ω is the set of points belonging to the object in that frame, and h is the bandwidth parameter. We use a Gaussian kernel with an automatically adapted bandwidth parameter (Botev *et al.*, 2010). This KDE is estimated on sparse points and can be sampled densely to obtain a dense confidence map φ as shown in Figure 7.4. This model is highly computationally efficient, similar to the shape priors in Lee *et al.* (2011). Integrating the confidence map into the energy function in Eq. (7.6) leads to

$$\min_{\boldsymbol{x}} E(\boldsymbol{x}, \widehat{\boldsymbol{y}}) = \sum_{i \in \mathcal{V}} (\phi_i(x_i) + \beta \varphi_i(x_i)) + \alpha \sum_{\{i,j\} \in \mathcal{E}} \phi_{ij}(x_i, x_j) \tag{7.8}$$

2-step For a given pixel labeling $\widehat{\boldsymbol{x}}$, minimizing the total energy function $E(\boldsymbol{x}, \boldsymbol{y})$ in terms of $\boldsymbol{y}$ leads to

$$\begin{aligned} \min_{\boldsymbol{y}} E(\widehat{\boldsymbol{x}}, \boldsymbol{y}) &= \beta \sum_{\{i,r\} \in \eta} \phi_{ir}(\widehat{x}_i, y_r) + \gamma \sum_{\{r,s\} \in \delta} \phi_{rs}(y_r, y_s) \\ &= \beta \sum_{r \in R} \phi_r(y_r) + \gamma \sum_{\{r,s\} \in \delta} \phi_{rs}(y_r, y_s) \end{aligned} \tag{7.9}$$

where R is the set of nodes for the point trajectories. When a pixel labeling is given, the unary and pairwise potentials (first 2 terms in Eq. (7.3)) become constant. Note that it sometimes happens that the pixel labels $\boldsymbol{x}_k$ along the trajectory tr_r are not consistent. For example, a trajectory consisting of 8 pixel points, which the first 6 are labeled as foreground (1) and the last 2 as background (0). The simple Potts model in Eq. (7.4) is not a good representative model anymore. We propose the following potentials instead

$$\phi_r(y_r) = \begin{cases} \frac{\mathrm{N}_{\boldsymbol{x}_k=\mathbf{1}}}{|\boldsymbol{x}_k|}, & \text{when } y_r = 1 \\ 1 - \frac{\mathrm{N}_{\boldsymbol{x}_k=\mathbf{1}}}{|\boldsymbol{x}_k|}, & \text{otherwise} \end{cases}$$

where $\mathrm{N}_{\boldsymbol{x}_k=\mathbf{1}}$ is the number of times that the element of $\boldsymbol{x}_k$ is labeled as 1, and $|\boldsymbol{x}_k|$ is the number of elements in $\boldsymbol{x}_k$. As the trajectory clustering potentials ϕ_{rs} are in the forms of Eq. (7.5), Eq. (7.9) can also be minimized using graph cuts (Boykov & Kolmogorov, 2004; Boykov *et al.*, 2001).

7.5 Experimental Results

7.5.1 Datasets and Implementation Details

We present experiments on a number of benchmark sequences, from SegTrack dataset (Tsai *et al.*, 2012) and Berkeley Motion Segmentation Dataset (Brox & Malik, 2010), with focus on the *parachute* and *marple3* sequences. The *parachute* sequence from Tsai *et al.* (2012) has a spatial resolution of 414×352, consists of 51 frames, and per frame pixel-level ground-truth for the primary foreground object. The *marple3* sequence from Brox & Malik (2010) has a spatial resolution of 350×288, consists of 323 frames, and sparse pixel-level ground-truth for the foreground object. The videos span a wide degree of difficulty with challenges such as illumination changes, fg/bg color overlap, large shape deformation, and large camera motion.

Implementation Details We use Lab color space histograms with 23 bins per channel, and $C = 5$ component GMMs. To describe motion, we use optical flow histograms with 61 bins per x and y direction, using Brox & Malik (2011). For all sequences, point trajectories are obtained by Brox & Malik (2010), for which there is binary code available. Brox & Malik (2010) also yields trajectory clusters that look very appealing but are sparse (see Figure 7.8 bottom row), for which we use for learning the GMM parameters. For the optimization, we set $\alpha = 5$ for pairwise potentials, $\beta = 0.5$ for pixel-trajectory compatibility potentials, and $\gamma = 5$ for the trajectory clustering potentials. These parameters are fixed for the inference of all sequences. The optimization typically converges in 3 to 4 iterations.

7.5.2 Results

To quantify segmentation accuracy, we use the average per-frame pixel error rate (Tsai *et al.*, 2012), $\epsilon(S) = \frac{XOR(S,GT)}{F}$, where S is each method's foreground labeling, GT is the ground-truth foreground segmentation, and F is the total number of frames. This score penalizes both over- and under-segmentation. We compare against three state-of-the-art methods: (1) the motion coherence segmentation method (Tsai *et al.*, 2012), (2) the level-set based tracker (Chockalingam *et al.*, 2009), and (3) the multi-level variational method (Ochs & Brox, 2011). First two methods require human labeling of the object boundary in the first frame. Last method requires multi-level superpixel extraction. In contrast, our method requires no hand drawn supervision and no superpixel to guide the segmentation. Table 7.1 shows the results. Note that segmentation error for the *marple3* sequence is evaluated on the first 50 frames and calculated using the frames where pixel-level ground-truths are available. Our method achieves state-of-the-art results on these sequences. Per-10th-frame pixel label error rate is shown for the *marple3* sequence in Figure 7.5. When the parameters (α, β, γ) are set as $(5, 0.7, 5)$, the segmentation error is 1962 for the *marple3* sequence. We also test other parameter combination for the *parachute* sequence, e.g. the segmentation errors are 308 $(1, 0.7, 5)$, 247 $(5, 0.7, 5)$, 270 $(5, 0.6, 5)$, 263 $(5, 0.3, 5)$ respectively, where (α, β, γ) are the different parameter setting. As we use iterative optimization, parameter selection is not critical for the final segmentation results.

Table 7.1: Segmentation error as measured by the average number of incorrect pixels per frame. Lower values are better. We compare our method with three state-of-the-art methods (Chockalingam *et al.*, 2009; Ochs & Brox, 2011; Tsai *et al.*, 2012).

	Our method	Tsai *et al.*	Chockalingam *et al.*	Ochs & Brox
parachute	238	235	502	463
marple3	1610	-	-	2092
Manual seg	no	yes	yes	no

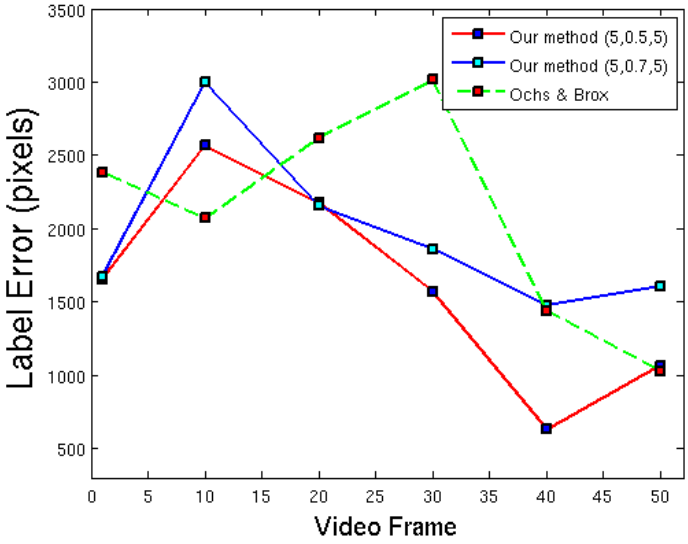


Figure 7.5: Per-10th-frame pixel label error rate of our approach and Ochs & Brox (2011) for the *marple3* sequence.

Figure 7.6 and Figure 7.7 show qualitative segmentation examples. Our method produces high quality segmentations of the foreground object. Fine details and object boundaries are comparable to Ochs & Brox (2011). Furthermore, the stability of the joint object and trajectory segmentation is demonstrated by the improved segmentation over Ochs & Brox (2011). Ochs & Brox (2011) produce only part of the parachute segment from frames 45 to 50 in Figure 7.6. While Ochs & Brox (2011) sometimes results in an over-segmentation of an object, our method produces a foreground segmentation at the object-level.

As our method jointly optimizes over object pixels and trajectories, we also present the comparison of our trajectory labeling and the trajectory clustering approach (Brox & Malik, 2010) in Table 7.2 in terms of overall clustering error (Brox & Malik, 2010). The overall clustering error is the number of bad labels over the total number of labels on a per-pixel basis. The tool provided by Brox & Malik (2010) optimally assigns clusters to ground truth regions. The results of our method are consistently better. Motion segmentation on sample frames of the *parachute* sequence is illustrated in Figure 7.8. Note that skater was assigned as foreground

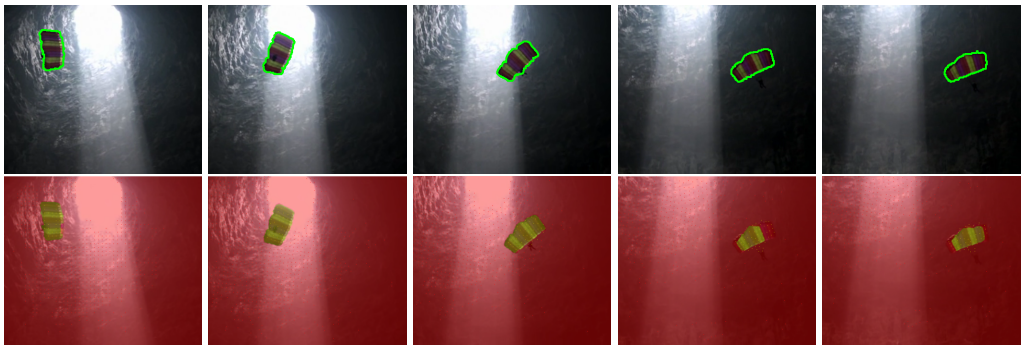


Figure 7.6: Comparison of our approach and the variational approach (Ochs & Brox, 2011) on frames 1, 15, 30, 45 and 50 of the *parachute* sequence from the SegTrack dataset (Tsai *et al.*, 2012) (The green boundary overlays with the original image for visualization.). *Top row*: our results, *Bottom row*: Ochs & Brox (2011).

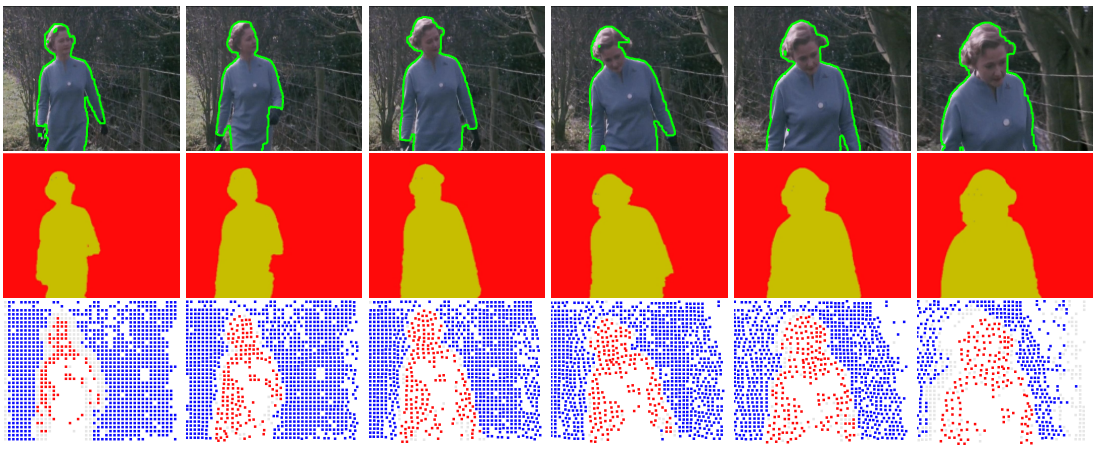


Figure 7.7: Comparison of our approach and the variational approach (Ochs & Brox, 2011) on frames 1, 10, 20, 30, 40 and 50 of the *marple3* sequence from the Berkeley Motion Segmentation Dataset (Brox & Malik, 2010) (The green boundary overlays with the original image for visualization.). *Top row*: our results, *Middle row*: Ochs & Brox (2011), *Bottom row*: motion segmentation results (Brox & Malik, 2010).

Table 7.2: Overall clustering error. We compare our method with Brox & Malik (2010). Note that we randomly sample ground truth frames of the *parachute* sequence.

	#GT frames	Our method	Brox & Malik
marple3	6	1.14	1.18
parachute	6	0.70	0.86
	12	0.70	0.88
	18	0.67	0.85
	24	0.67	0.86

in trajectory clustering results from Brox & Malik (2010) (see skater in Figure 7.8 2nd and 3rd columns). For our method, during optimization iteration, point trajectories which do not belong to the foreground object has been reassigned as background.

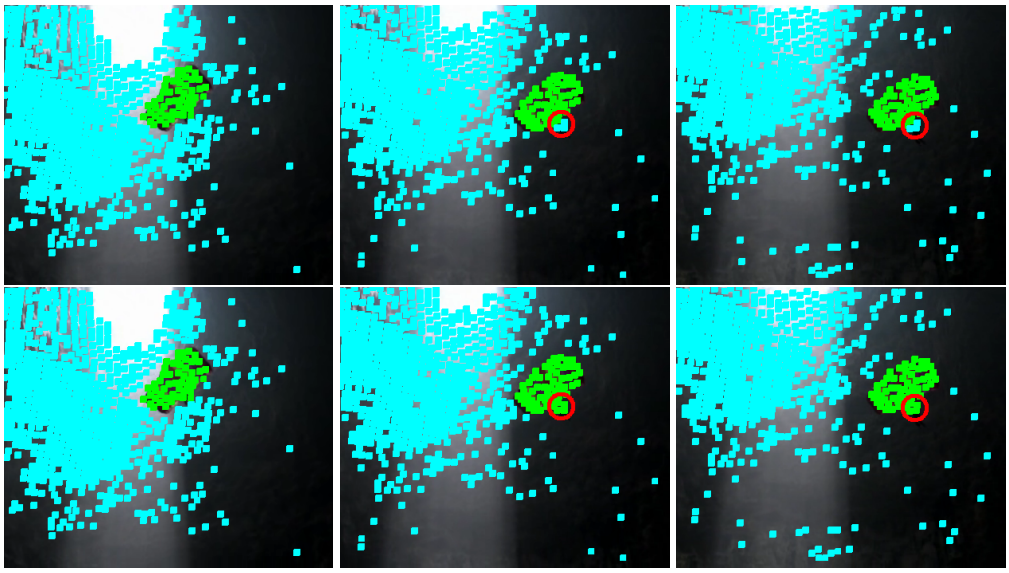


Figure 7.8: Comparison of our trajectory labeling and the trajectory clustering approach (Brox & Malik, 2010) on sample frames of the *parachute* sequence from the SegTrack dataset (Tsai *et al.*, 2012) (see the differences in red circles). *Top row*: our trajectory labeling results, *Bottom row*: trajectory labeling results from Brox & Malik (2010).

Figure 7.9 shows some additional examples that illustrate the final segmentation results of our method on video sequences. The typical failure cases are shown in Figure 7.9 bottom row. The failure is usually caused by very bad sparse labeling for GMM initialization. The limitation of our current method is that it relies on good point trajectory clustering results from Brox & Malik (2010). This could be alleviated by using *objectness measure* (Alexe *et al.*, 2012) or *key-segments* (Lee *et al.*, 2011) for GMM initialization.

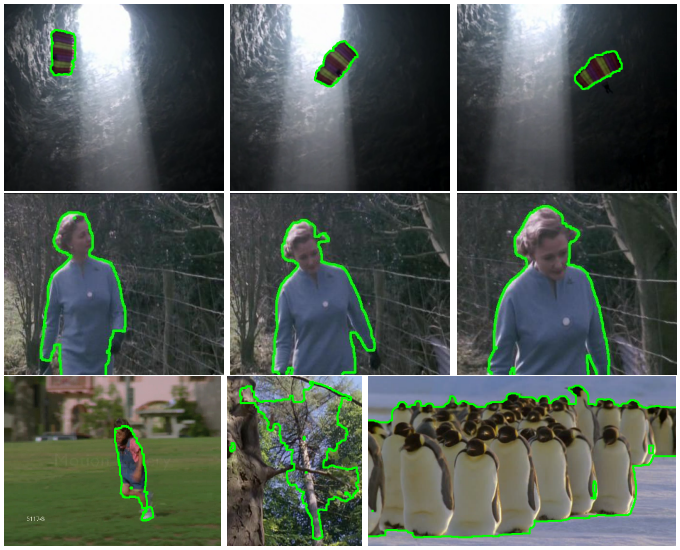


Figure 7.9: Additional segmentation results.

7.6 Conclusion

We presented a bottom-up approach for the combination of object segmentation and motion segmentation using a novel CRF model. The CRF contains binary variables representing the class labels of image pixels as well as binary variables indicating the correctness of trajectory clustering, which integrates dense local interaction and sparse global constraints. Hereby, we overcome the limitations of previous bottom-up unsupervised methods that often over-segment an object, and is, to the best of our knowledge, the first approach to combine object labeling and trajectory clustering in a unified probabilistic framework. Our method is fully automatic and unsupervised. The experiments demonstrate the high performance of our approach on benchmark datasets. In our ongoing work, we aim to integrate the proposed model into a system for multi-modal video cosegmentation.

Acknowledgments

The work is funded by the ERC-Starting Grant (DYNAMIC MINVIP). The authors gratefully acknowledge the support.

Chapter 8

Slice Sampling Particle Belief Propagation

Inference in continuous label Markov random fields is a challenging task. We use particle belief propagation (PBP) for solving the inference problem in continuous label space. Sampling particles from the belief distribution is typically done by using Metropolis-Hastings (MH) Markov chain Monte Carlo (MCMC) methods which involves sampling from a proposal distribution. This proposal distribution has to be carefully designed depending on the particular model and input data to achieve fast convergence. In this chapter, we propose to avoid dependence on a proposal distribution by introducing a slice sampling based PBP algorithm. The proposed approach shows superior convergence performance on an image denoising toy example. Our findings are validated on a challenging relational 2D feature tracking application. An earlier version of this chapter appeared at the IEEE International Conference on Computer Vision (ICCV) (Müller *et al.*, 2013).

8.1 Introduction

Markov Random Fields (MRFs) are a powerful tool for modeling relational dependencies among observations. Inference in such models is an inherent problem which has been widely addressed in the past. MRFs, and hence its inference methods, can be classified in two categories: discretely and continuously labeled problems. Numerous optimization approaches for discrete labels have been proposed, from binary labeled Graph Cuts (Boykov *et al.*, 2001), to multi-label tree reweighted message passing (Kolmogorov, 2006; Wainwright *et al.*, 2005). In this work, we deal with continuous labeled MRFs where we use a particle belief propagation (PBP) approach (Ihler & McAllester, 2009). The efficiency of such particle based approaches highly depends on the sampling scheme used to explore the label space. Previous approaches use Metropolis-Hastings (MH) Markov chain Monte Carlo (MCMC) methods for particle sampling. The performance of these methods depends on a carefully designed proposal distribution.

Contributions. We propose a novel sampling technique for PBP based on slice sampling (Neal, 2003). This method exploits the structure of the PBP message passing equations for direct sampling from the target distribution and does not depend on a proposal distribution

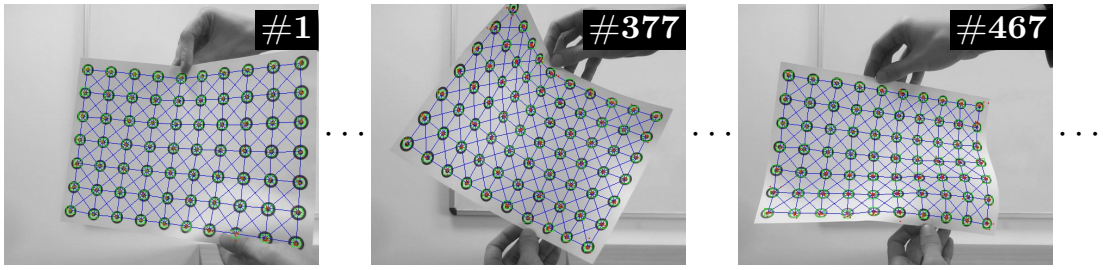


Figure 8.1: Relational 2D feature tracking example.

which is difficult to tune. We show the superiority of our method theoretically on a simplified toy application on image denoising. Our findings are then verified on a complex 2D relational feature tracking application as shown in Figure 8.1. We furthermore provide a publicly available database of image sequences for feature tracking applications including manually labeled groundtruth data (Müller *et al.*, 2013).

The rest of the work is organized as follows. Section 8.2 provides an overview of related work. Section 8.3 introduces notations and definitions used throughout this chapter and gives a short introduction to slice sampling. Our proposed approach is described in detail in Section 8.4. The proposed approach shows superior convergence performance on an image denoising example in Section 8.5. In Section 8.6 we present a thorough evaluation of our method compared to the state-of-the-art and propose a 2D relational feature tracking application. We conclude our findings in Section 8.7.

8.2 Related Work

Most works on MRF optimization specialize on a discrete label space (Boykov *et al.*, 2001; Kolmogorov, 2006; Wainwright *et al.*, 2005). Often such approaches are hard to apply on tasks where a continuous label space would be a more natural choice, such as feature tracking with relational constraints (Lin & Liu, 2006; Salzmann & Urtasun, 2012).

Loopy belief propagation is a prominent method using a local message passing mechanism for coordinating the optimal labeling of neighboring nodes. These methods work on discrete label spaces. The computational complexity is $\mathcal{O}(n^2)$ over the number of discrete labels n, making computations with many labels for approximating near-continuous models intractable (Sudderth *et al.*, 2010).

Recently, message passing approaches working in continuous rather than discrete label space were proposed (Ihler & McAllester, 2009; Kothapa *et al.*, 2011; Peng *et al.*, 2011; Sudderth *et al.*, 2010). These approaches use MCMC methods to approximate the message distributions. To the best of our knowledge, all previously proposed MCMC based belief propagation methods use Metropolis-Hastings (MH) sampling. This sampling strategy consists of two steps: (a) sampling a candidate particle from an easy to sample *proposal distribution*, and (b) accept or reject the candidate depending on a transition probability (Walsh, 2004). Applying this sampling technique involves a careful design of the proposal distribution, which is a compromise

between exploring the label space (using a broad proposal distribution) and maximizing the transition acceptance ratio (minimize sample moves) at the same time.

Throughout this work we show that considering alternative sampling techniques can be advantageous. We propose to use slice sampling (Neal, 2003) instead of MH, rendering proposal distribution selection obsolete in the context of PBP.

To demonstrate superior performance of our method on a real world problem we propose a relational feature tracking application inspired by (Lin & Liu, 2006; Salzmann & Urtasun, 2012) in the experiment section in this chapter. Some related works such as Duan *et al.* (2012); Shitrit *et al.* (2011) propose to formulate feature tracking as a discrete labeling problem and use global optimization algorithms (i.e. linear programming or dynamic programming). Such approaches need some sort of label pruning in order to keep computational complexity low. Closely related methods use belief propagation combined with particle filtering (Lin & Liu, 2006; Salzmann & Urtasun, 2012; Xue *et al.*, 2008), but still use proposal distributions for particle perturbation which introduces sensible optimization parameter tuning.

8.3 Definitions and Notation

8.3.1 Markov Random Field

Let $\mathcal{V}$ be a set of nodes and $\mathcal{N}_s \subset \mathcal{V}$ the set of neighboring nodes to $s \in \mathcal{V}$. For every node s there is a label x_s from the *label space* $\mathcal{L}_s$. The product $\mathcal{L} = \prod_{s\in\mathcal{V}} \mathcal{L}_s$ is the space of configurations $\mathbf{x} = \{x_s\}_{s\in\mathcal{V}}$. A Markov random field potential energy is given by:

$$E(\mathbf{x}) = \sum_{s\in\mathcal{V}} \psi_s(x_s) + \sum_{s\in\mathcal{V}} \sum_{t\in\mathcal{N}_s} \psi_{s,t}(x_s, x_t) \tag{8.1}$$

with a *unary potential* function $\psi_s(x_s)$ and a *binary potential* function $\psi_{s,t}(x_s, x_t)$. Then $p(\mathbf{x}) \propto \exp\left[-E(\mathbf{x})\right]$ defines a Markov random field (MRF).

We consider the problem of computing the maximum marginals: $\mu(x_s) = \max_{\mathbf{x}'|x'_s=x_s} p(\mathbf{x}')$[1].

8.3.2 Max-Product Particle Belief Propagation

In the following we summarize the max-product particle belief propagation algorithm (Besse *et al.*, 2012; Kothapa *et al.*, 2011). The energy term $E(\mathbf{x})$ is approximated by particles such that the label space $\mathcal{L}_s$ of each node s in the MRF is represented by a set of particles $P_s = \{x_s^{(1)}, \ldots, x_s^{(p)}\}$, where p is the number of particles per node. Then the estimated *belief* $b_s^n(x_s^{(i)})$ or *log disbelief* $B_s^n(x_s^{(i)}) = -\log(b_s^n(x_s^{(i)}))$ of node s at iteration n is calculated as follows (Besse *et al.*, 2012):

$$B_s^n(x_s^{(i)}) = \psi_s(x_s^{(i)}) + \textstyle\sum_{t\in\mathcal{N}_s} M_{t\to s}^n(x_s^{(i)}), \tag{8.2}$$

[1] Backtracking can be used to compute the MAP-configuration $\mathbf{x}^* = \arg\max_{\mathbf{x}} p(\mathbf{x})$ from the max-marginals (Kothapa *et al.*, 2011).

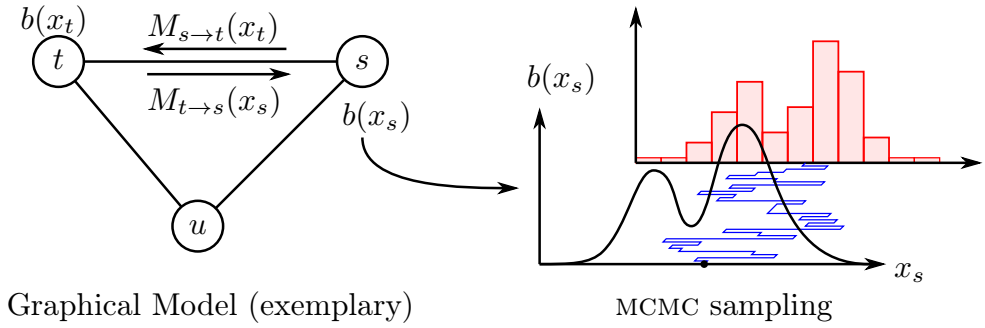


Figure 8.2: Particle Belief Propagation framework. Left: Message passing mechanism. Right: MCMC particle sampling of the belief $b(x_s)$ with an exemplary MCMC sampling chain of one particle (blue) and its corresponding histogram (red).

where the *messages* $M^n_{t\to s}(x_s)$ for $x_s \in P_s$ from node t to node s are:

$$M^n_{t\to s}(x_s) = \min_{x_t \in P_t} [\psi_{s,t}(x_s, x_t) + B^{n-1}_t(x_t) - M^{n-1}_{s\to t}(x_t)]. \tag{8.3}$$

Note that the log disbelief $B^n_s(x_s)$ and the messages $M^n_{t\to s}(x_s)$ can be calculated for all continuous values $x_s \in \mathcal{L}_s$ rather than only on the particle set P_s. On the other hand, the messages from node s to node t are approximated only using the particles x_t from the particle set $P_t = \{x^{(1)}_t, \ldots, x^{(p)}_t\}$ of node t.

Messages and log disbeliefs are calculated iteratively for $n = 1, \ldots, N$ iterations. An estimate of the most likely configuration can be obtained with

$$\hat{x}_s = \arg\min_{x_s} B^N_s(x_s). \tag{8.4}$$

The main issue in PBP lies in how to sample new particles $x^n_s \sim B^n_s(x_s)$. Typically, the Metropolis-Hastings (MH) MCMC method is used. This method requires a proposal distribution q where new particles can be easily sampled from. Typically a Gaussian function $q = p_\sigma$ with a predefined standard deviation σ is used.

Figure 8.2 shows a schematic overview of the PBP framework. Figure 8.3 summarizes the Metropolis-Hastings based max-product particle belief propagation algorithm (MH-PBP).

Typically, q needs to be carefully adjusted to the true belief distribution. This introduces a dependency on prior knowledge about how the labels are distributed in the label space. In the following we propose to replace the MH sampling step by a slice sampling approach which does not depend on proposal distribution selection.

8.3.3 MCMC Slice Sampling

In this section we briefly summarize the concept of slice sampling (Andrieu *et al.*, 2003; Neal, 2003) which is defined in a general MCMC sampling framework. Suppose we are given a distribution $q(x)$ and want to sample from this distribution, i.e. MCMC sampling of M samples

Input: Initial set of particles: $\{x_s^{(i)}\}_{i=1,\ldots,p}$, proposal distribution p_σ

Initialize the messages $M_{t\to s}^0(x_s)$ and log disbelief $B_s^0(x_s^{(i)})$ with zero $\forall s, t$
for BP iteration $n = 1$ to N **do**
  for each node s and each particle $i = 1, \ldots, p$ **do**
    Initialize sampling chain $x_s^{(i)\langle 0\rangle} \leftarrow x_s^{(i)}$
    for MCMC iteration $m = 1, \ldots, M$ **do**
      Sample $\bar{x}_s^{(i)\langle m\rangle} \sim p_\sigma(x \mid x_s^{(i)\langle m-1\rangle})$
        from proposal distribution p_σ
      Calc. belief $B_s^n(\bar{x}_s^{(i)\langle m\rangle})$ from Eqs. (8.2), (8.3)
      Sample $u \sim \mathcal{U}_{[0,1]}(u)$
      if $B_s^n(\bar{x}_s^{(i)\langle m\rangle}) < B_s^n(x_s^{(i)\langle m\rangle}) - \log(u)$ **then**
        Accept: $x_s^{(i)\langle m\rangle} \leftarrow \bar{x}_s^{(i)\langle m\rangle}$
      end if
    end for
    $x_s^{(i)} \leftarrow x_s^{(i)\langle M\rangle}$
  end for
  Normalize messages and beliefs
end for

Figure 8.3: MH-PBP (Besse *et al.*, 2012; Kothapa *et al.*, 2011)

$x^{\langle 1\rangle}, x^{\langle 2\rangle}, \ldots, x^{\langle M\rangle}$:

$$x^{\langle m\rangle} \sim q(x \mid x^{\langle m-1\rangle}), \tag{8.5}$$

given an initial sample $x^{\langle 0\rangle}$.

Note that in the PBP framework, there is a MCMC sampling chain $\{x_s^{(i)\langle m\rangle}\}_{m=1,\ldots,M}$ for each particle $x_s^{(i)}$. MCMC sampling could be done using several sampling techniques such as Metropolis-Hastings (MH) or Gibbs sampling (provided the conditional distributions are easy to sample from). Metropolis-Hastings sampling has the drawback of requiring a proposal distribution. Choosing the proposal distribution is very often a difficult task and introduces a compromise between reducing the rejection rate and obtaining large random moves (Andrieu *et al.*, 2003).

In slice sampling, an auxiliary variable $u \in \mathbb{R}$ is introduced and the target distribution $q(x)$ is extended to

$$q^*(x, u) = \begin{cases} 1 & \text{if } u \in [0, q(x)] \\ 0 & \text{otherwise} \end{cases} \tag{8.6}$$

Sampling is then done by uniformly drawing the auxiliary variable u (defining the *slice*) and given this, uniformly drawing the new sample from an interval A defined over u as follows:

$$u^{\langle m\rangle} \sim q(u \mid x^{\langle m-1\rangle}) = \mathcal{U}_{[0,q(x^{\langle m-1\rangle})]}(u) \tag{8.7}$$

$$x^{\langle m\rangle} \sim q(x \mid u^{\langle m\rangle}) = \mathcal{U}_A(x), \tag{8.8}$$

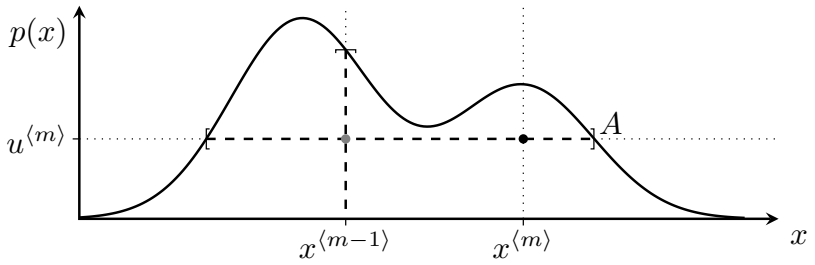


Figure 8.4: Slice Sampling (Andrieu *et al.*, 2003; Neal, 2003)

where $\mathcal{U}_I$ is the uniform distribution over an interval I and $A = \{x;\ q(x) \geq u^{\langle m \rangle}\}$. Figure 8.4 shows an exemplary slice sampling step.

Assume that $q(x)$ can be decomposed in L functions $f_l(x)$ such that $q(x) \propto \prod_{l=1}^{L} f_l(x)$. Then we can sample over $q(x)$ by introducing L auxiliary variables $u_1, \ldots, u_L$:

$$u_1^{\langle m \rangle} \sim q(u_1 \mid x^{\langle m-1 \rangle}) = \mathcal{U}_{[0, f_1(x^{\langle m-1 \rangle})]}(u_1) \tag{8.9}$$

$$\vdots$$

$$u_L^{\langle m \rangle} \sim q(u_L \mid x^{\langle m-1 \rangle}) = \mathcal{U}_{[0, f_L(x^{\langle m-1 \rangle})]}(u_L) \tag{8.10}$$

$$x^{\langle m \rangle} \sim q(x \mid u_1^{\langle m \rangle}, \ldots, u_L^{\langle m \rangle}) = \mathcal{U}_{A^{\langle m \rangle}}(x), \tag{8.11}$$

where $A^{\langle m \rangle} = \{x\ ;\ f_l(x) \geq u_l^{\langle m \rangle}, l = 1, \ldots, L\}$ (Andrieu *et al.*, 2003).

The main difficulty lies in determining the interval A. Fortunately it turns out, that in the max-product particle belief propagation framework the sampling interval A can be determined efficiently as shown in the following section.

8.4 Slice Sampling Particle Belief Propagation

Our main contribution is presented in this section. We propose to sample particles from the belief $b(x_s)$ using slice sampling rather than Metropolis-Hastings sampling. For applying the slice sampler, the sampling interval $A^{(i)\langle m \rangle}$ needs to be determined for the ith particle of node s and for the mth MCMC iteration which we can uniformly sample the particle $x_s^{(i)\langle m \rangle}$ from. The superscripts $(i)\langle m \rangle$ are omitted in the following for better readability.

The goal is to determine the sampling interval A. Given the potential functions $\psi_s(x_s)$ and $\psi_{s,t}(x_s, x_t)$, it is assumed that the intervals

$$A_{\psi_s}(\bar{u}) = \{x_s\ ;\ \psi_s(x_s) \leq \bar{u}\} \quad \text{and} \tag{8.12}$$

$$A_{\psi_{s,t}}^{x_t}(\bar{u}) = \{x_s\ ;\ \psi_{s,t}(x_s, x_t) \leq \bar{u}\} \tag{8.13}$$

can be computed analytically. Note that computations are done in negative log space, thus a slice interval $\{x\ ;\ f(x) \geq u\}$ is transformed to $\{x\ ;\ -\log(f(x)) \leq \bar{u}\}$, where $\bar{u}$ is the negative

logarithm of a uniformly sampled value.

The final sampling interval A can be computed from these intervals as shown below. If the intervals cannot be computed analytically then an approximated interval $\tilde{A}$ may be still computed and rejection sampling can be applied (Andrieu *et al.*, 2003).

The log disbelief can be decomposed as follows:

$$B(x_s) = \sum_{l=0}^{|\mathcal{N}_s|} F_l(x_s) \tag{8.14}$$

with $F_0(x_s) = \psi_s(x_s)$ and $F_j(x_s) = M_{t^{(j)} \to s}(x_s)$ where $t^{(j)}$ is the j-th neighbor of s. From this follows the decomposition of the sampling interval

$$A = \bigcap_{l=0}^{|\mathcal{N}_s|} A_l, \quad \text{with} \quad A_l = \{x \; ; \; F_l(x) \leq \bar{u}_l\}. \tag{8.15}$$

Using the definitions of Eqs. (8.2, 8.3), we obtain for A_l:

$$\begin{aligned} A_0 &= A_{\psi_s}(\bar{u}_1) \qquad (8.16) \\ A_j &= \{x_s \; ; \; M_{t^{(j)} \to s}(x_s) \leq \bar{u}_j\} \\ &= \{x_s \; ; \; \min_{x_t \in P_t} G_j^{x_t}(x_s) \leq \bar{u}_j\} \\ &= \bigcup_{x_t \in P_t} \{x_s \; ; \; G_j^{x_t}(x_s) \leq \bar{u}_j\} \\ &= \bigcup_{x_t \in P_t} A_{\psi_{s,t}}^{x_t}(\bar{u}_j - B_t(x_t) + M_{s \to t}(x_t)), \qquad (8.17) \end{aligned}$$

where $G_j^{x_t}(x_s) = \psi_{s,t}(x_s, x_t) + B_t^{n-1}(x_t) - M_{s \to t}^{n-1}(x_t)$. This result shows that A only depends on the given intervals $A_{\psi_s}(\bar{u})$ and $A_{\psi_{s,t}}^{x_t}(\bar{u})$ which are defined by the unary and binary potential functions ψ_s and $\psi_{s,t}$. Figure 8.5 summarizes the proposed method.

We further refer to the proposed technique as S-PBP (slice sampling particle belief propagation).

Example. Consider a quadratic unary potential function $\phi_s(x_s) = (x_s - d_s)^2$. Then $A_{\phi_s}(\bar{u})$ has the closed form solution $A_{\phi_s}(\bar{u}) = \{x_s : (x_s - d_s)^2 \leq \bar{u}\} = [d_s - \sqrt{\bar{u}}, d_s + \sqrt{\bar{u}}]$. Similarly, the closed form solution for $\phi_{s,t}(x_s, x_t) = (x_s - x_t)^2$ is $A_{\phi_{s,t}}^{x_t}(\bar{u}) = [x_t - \sqrt{\bar{u}}, x_t + \sqrt{\bar{u}}]$.

Multidimensional Bounds. In order to deal with multidimensional label spaces, i.e. $\mathcal{L}_s \in \mathbb{R}^d$ for $d > 1$, we propose to randomly sample one dimension in each MCMC step and slice sample on this dimension while the other dimensions are held fixed.

Analytic Bounds Calculation. Assume the unary and/ or binary potential functions ψ_s and ψ_{st} are given as an analytic function. Then one can use standard computer algebra solvers for defining $A_{\psi_s}(u)$ and/or $A_{\psi_{s,t}}^{x_t}(u)$. We have implemented our S-PBP framework in MATLAB®

Input: Initial set of particles: $\{x_s^{(i)}\}_{i=1,\dots,p}$

Initialize the messages $M_{t\to s}^0(x_s)$ and log disbelief $B_s^0(x_s^{(i)})$ with zero $\forall s, t$
for BP iteration $n = 1$ to N **do**
  for each node s and each particle $i = 1, \dots, p$ **do**
    Initialize sampling chain $x_s^{(i)\langle 0\rangle} \leftarrow x_s^{(i)}$
    for MCMC iteration $m = 1, \dots, M$ **do**
      Sample $\bar{u}_l = F_l(x_s^{(i)\langle m-1\rangle}) - \log(u_l)$ where
        $u_l \sim \mathcal{U}_{[0,1]}(u)$ for $l = 0, \dots, |\mathcal{N}_s|$
      Compute $A^{(i)\langle m\rangle}$ from Eqs. (8.15), (8.16), (8.17)
      Sample $\bar{x}_s^{(i)\langle m\rangle} \sim \mathcal{U}_{A^{(i)\langle m\rangle}}(x)$
      Calc. belief $B_s^n(\bar{x}_s^{(i)\langle m\rangle})$ from Eqs. (8.2), (8.3)
      if $F_l(\bar{x}_s^{(i)\langle m\rangle}) \leq \bar{u}_l$ for $l = 0, \dots, |\mathcal{N}_s|$ **then**
        Accept: $x_s^{(i)\langle m\rangle} \leftarrow \bar{x}_s^{(i)\langle m\rangle}$
      end if
    end for
    $x_s^{(i)} \leftarrow x_s^{(i)\langle M\rangle}$
  end for
  Normalize messages and beliefs
end for

Figure 8.5: S-PBP

with MEX and use the MATLAB®-MUPAD® interface to solve the inequalities automatically. This way no manual work has to be done.

8.5 Image Denoising

8.5.1 Denoising Model

For analyzing the random walk behaviour of our method we have chosen the application of image denoising due to its relatively simple model structure. The basic image denoising model is as follows:

$$\begin{aligned}\psi_s(x_s) &= \theta_1(x_s - d_s)^2,\\ \psi_{s,t}(x_s, x_t) &= \theta_2 \min\{\theta_3, (x_s - x_t)^2\}.\end{aligned} \tag{8.18}$$

For minimizing particle noise in the final estimation result an annealing scheme is used where the target belief distribution is modified to $b_s^n(x_s^{(i)})^{1/T_n}$, where $T_n = T_0 \cdot (T_N/T_0)^{n/N}$ is the temperature at PBP iteration n, T_0 is the start temperature, and T_N the end temperature. Given this annealing scheme the temperature is successively reduced for each new iteration n.

The evaluation was done on an example image as shown in Figure 8.6. The training and testing sets each include 10 noisy image instances with Gaussian noise standard deviation $\sigma =$

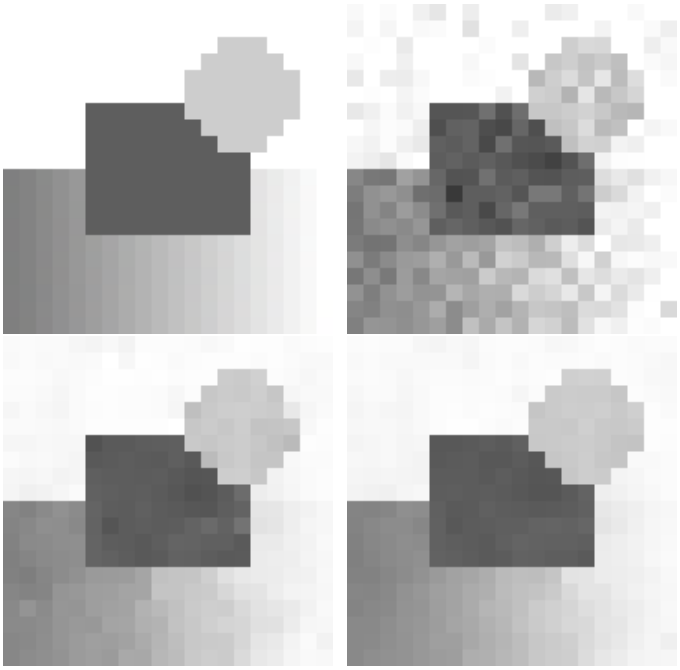


Figure 8.6: Denoising example: Groundtruth (top left), noisy input example (top right), reconstruction with MH-PBP (bottom left), reconstruction with our proposed S-PBP method (bottom right).

0.05 (where image intensity $\in [0, 1]$). Training of the parameter vector $\theta = \{\theta_1, \theta_2, \theta_3\}$ is done by minimizing the *empirical risk* $R(\theta) = \frac{1}{K}\sum_{i=1}^{K} L(\mathbf{x}_\theta^{(i)}, \mathbf{y}^{(i)})$ given the *loss function* $L(\mathbf{x}, \mathbf{y}) = \|\mathbf{x}-\mathbf{y}\|_2^2$ where $\{\mathbf{y}^{(i)}, \mathbf{d}^{(i)}\}$ is the training data pair with groundtruth $\mathbf{y}^{(i)}$ and noisy observation $\mathbf{d}^{(i)}$. $\mathbf{x}_\theta^{(i)}$ is the MAP estimate given $\mathbf{d}^{(i)}$ and the parameter θ. Learned parameters are $\theta_1 = 0.756$, $\theta_2 = 1.170$, $\theta_3 = 0.0059$.

8.5.2 Comparing S-PBP with MH-PBP

We further compared the efficiency of the slice sampling method to the Metropolis-Hastings sampling applied on the image denoising problem. For the experimental setup we use $N = 100$ PBP iterations, $p = 5$ particles, and a temperature schedule of $T_0 = 1$ to $T_N = 10^{-4}$. An MCMC chain of $M = 500$ samples is generated for each particle and in each PBP iteration. The iteration numbers are chosen to be more than sufficiently large in order to guarantee convergence and to collect statistical information in the MCMC chains in steady-state situations. For the MH-PBP proposal distribution the family of Gaussian distributions $p_\sigma(x \mid x^{\langle m-1\rangle}) = (2\pi\sigma^2)^{-0.5} \cdot \exp[-0.5(x - x^{\langle m-1\rangle})^2 \cdot \sigma^{-2}]$ is used. In order to provide a fair comparison the proposal distribution is adapted to the current temperature by using $p_\sigma(x \mid x^{\langle m-1\rangle})^{1/T_n}$ instead.

Figure 8.7 shows a comparison of the empirical risk for different MH-PBP proposal distributions. For $\sigma > 0.7$ the empirical risk stays nearly at the same level and thus we selected

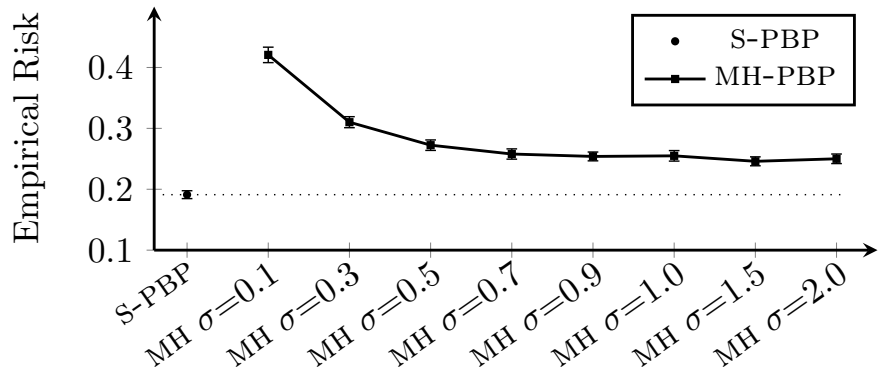


Figure 8.7: Comparison of the empirical risk for S-PBP and MH-PBP with different proposal distributions.

$\sigma = 0.7$ for further experiments. Another observation is that S-PBP outperforms MH-PBP in terms of minimal empirical risk. This is because the reconstructed images with MH-PBP have always much higher noise than images reconstructed with S-PBP. This effect can be significantly reduced by averaging over particles instead of only selecting the best one as stated in Eq. (8.4).

For comparing the random walk behavior of the MCMC sampling chains from S-PBP and MH-PBP, the normalized autocorrelation function

$$\rho_k = \frac{\sum_{m=1}^{M-k}(x^{\langle m\rangle} - \bar{x})(x^{\langle m-k\rangle} - \bar{x})}{\sum_{m=1}^{M-k}(x^{\langle m\rangle} - \bar{x})^2}, \tag{8.19}$$

where $\bar{x} = \frac{1}{M}\sum x^{\langle m\rangle}$, is used (Walsh, 2004). Only the last 50% of the MCMC chain is considered to skip any burn-in phase. Figure 8.8 shows a comparison of the first 20 kth order autocorrelation of S-PBP and MH-PBP at different PBP iterations n (and thus at different temperatures T_n). It can be observed that the MH-PBP method produces a much higher autocorrelation than the S-PBP method, thus the MCMC chain mixing behaviour of S-PBP outperforms MH-PBP.

8.6 Relational Feature Tracking

We propose to apply our S-PBP algorithm on a 2D relational feature tracking system inspired by Lin & Liu (2006); Salzmann & Urtasun (2012) as a more complex application.

8.6.1 Tracker Model

The proposed feature tracker uses a pairwise MRF model. The model is separated into two parts: (a) the unary potentials are derived from a feature patch matching model, and (b) the binary potentials encode the relative positioning of the features to each other. The label space of the MRF is the space of feature poses including the local central patch position, patch rotation, and

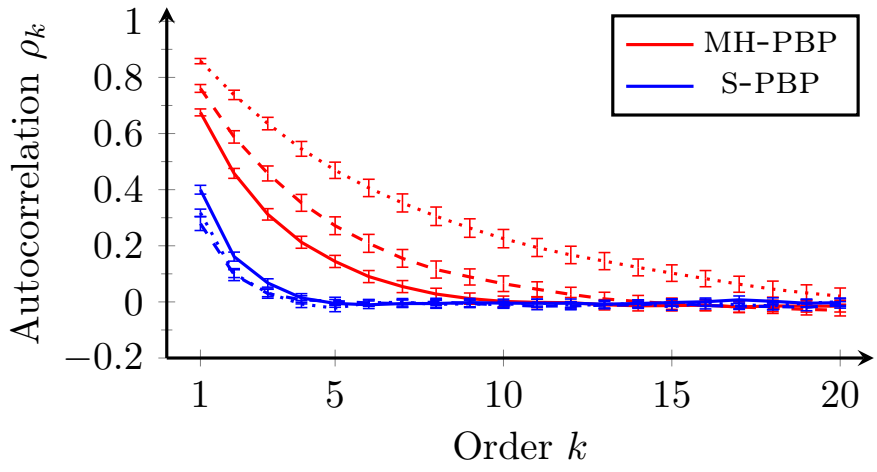


Figure 8.8: Comparison of S-PBP and MH-PBP at different PBP iterations (dotted $n = 30$, dashed $n = 50$, and solid $n = 70$) using an annealing schedule.

scale. The proposed MRF model is as follows:

$$E(\mathbf{x}) = \sum_{s \in \mathcal{V}} \psi_s(x_s) + \sum_{s \in \mathcal{V}} \sum_{t \in \mathcal{N}_s} \alpha \cdot \psi_{s,t}(x_s, x_t), \tag{8.20}$$

where the unary potential function

$$\psi_s(x_s) = \chi^2(\mathrm{HOG}_{I_n}(\mathbf{p}_s, \mathbf{o}_s) - \mathrm{HOG}_{I^{\mathrm{ref}}}(\mathbf{p}_s^{\mathrm{ref}}, \mathbf{o}_s^{\mathrm{ref}})) \tag{8.21}$$

is the Chi-square distance of HOG features (Ludwig *et al.*, 2009) of a patch at position $\mathbf{p}_s \in \mathbb{R}^2$ of the current image I_n and orientation $\mathbf{o}_s \in \mathbb{R}^2$, where $x_s = \{\mathbf{p}_s, \mathbf{o}_s\}$ and a reference image I^{ref} at reference position $\mathbf{p}_s^{\mathrm{ref}}$ and orientation $\mathbf{o}_s^{\mathrm{ref}}$. The orientation vector $\mathbf{o}_s$ encodes two aspects: the feature patch rotation (rotation of $\mathbf{o}_s$, i.e. $\mathrm{atan2}(\mathbf{o}_s)$) and feature patch scale (length of $\mathbf{o}_s$, i.e. $\|\mathbf{o}_s\|_2$).

The binary potential $\psi_{s,t}(x_s, x_t)$ is as follows:

$$\psi_{s,t}(\,\cdot\,) = \frac{\|\mathbf{p}_t - \mathbf{p}_s - \mathbf{R}_s \mathbf{d}_{st}\|_2^2 + \|\mathbf{p}_s - \mathbf{p}_t - \mathbf{R}_t \mathbf{d}_{ts}\|_2^2}{2 \cdot \|\mathbf{d}_{st}\|_2^2} \tag{8.22}$$

where $\mathbf{d}_{st(ts)} = \mathbf{p}_{t(s)}^{\mathrm{ref}} - \mathbf{p}_{s(t)}^{\mathrm{ref}}$ and $\mathbf{R}_{s(t)} = [o_{x,s(t)}, -o_{y,s(t)}; o_{y,s(t)}, o_{x,s(t)}]$ is a 2×2 rotation and scale matrix. The proposed binary potential function models the surrounding of each feature point as a weak-perspective model and transforms its neighbor points (with respect to the reference frame) according to a similarity transformation (consisting of translation, rotation, and scaling).

The scalar parameter $\alpha > 0$ is a weighting factor determining the "stiffness" of the feature mesh balancing between feature point independence ($\alpha \to 0$; i.e. multi-target tracker) and rigid

single object tracking.

8.6.2 Tracker Pipeline

A practical application requires some common modifications of the basic tracker pipeline in Section 8.6.1. The modifications include an additional *particle resampling* step, where for each frame the initial set of particles are sampled with replacement from the set of particles $\{x_s^{(i)}\}_{i=1,\dots,p}$ from the previous frame with probability $b_s^N(x_s^{(i)})$. For the tracker to be able to deal with fast moving objects, a *resolution pyramid* approach is applied. The resolution pyramid is only applied to the unary potential function, i.e. the feature descriptor is a concatenation of HOG descriptors of patches with the same center position but differing spatial resolution. For each resolution pyramid level (*scale*) the image is downsampled by a factor of 0.5 using bicubic interpolation.

Slice sampling. For the slice sampling approach we need to define the boundary functions $A_{\psi_s}(u)$ and $A^{x_t}_{\psi_{s,t}}(u)$. Since $\psi_{s,t}$ is given as an analytic function we can use our automatic inequality solver as described in Section 8.4. An analytic description of the unary potential is not available thus we have to define the boundary manually. We choose to set $A_{\psi_s}(u)$ to the whole image space for $\mathbf{p}_s$, i.e. $\mathbf{p}_s \in [1, W] \times [1, H]$, where W and H are the image width and height respectively, and to restrict $\mathbf{o}_s$ to $\mathbf{o}_s \in [-10, 10] \times [-10, 10]$. This way it is ensured that the sampling space is large enough. On the other hand, particles sampled outside the true (sub-)bounds are automatically rejected by the algorithm.

Metropolis-Hastings sampling. In order to provide a fair comparison of our slice sampling approach to the state-of-the-art MH-PBP approach, the design of the proposal distribution has to be done very carefully. We propose to use a 4D Gaussian distribution with a covariance matrix Σ combined with a suitable coordinate transformation to ensure a well-mixing random walk behavior. The label space can be divided into two parts, the feature position $\mathbf{p}_s \in \mathbb{R}^2$ and orthogonal feature transformation $\mathbf{o}_s \in \mathbb{R}^2$. The proposal distribution for $\mathbf{p}_s$ is $p(\mathbf{p}_s^{\langle m \rangle} \mid \mathbf{p}_s^{\langle m-1 \rangle}) = \mathcal{N}(\mathbf{p}_s^{\langle m-1 \rangle}, \mathbf{I}_{2\times 2} \cdot \sigma_{xy})$, where $\mathcal{N}(\mu, \Sigma)$ is a Gaussian pdf with mean μ and covariance Σ. $\mathbf{I}_{2\times 2}$ is the 2×2 identity matrix. The vector $\mathbf{o}_s$ is sampled analogously, but in the polar coordinate system with covariance matrix $\Sigma_{\text{polar}} = [\sigma_r^2, 0; 0, \sigma_\phi^2]$, where σ_r^2 is the variance for the radius and σ_ϕ^2 the variance for the angle. Finally we have to carefully tune the three parameters σ_{xy}, σ_r, and σ_ϕ.

8.6.3 Tracker Evaluation

Test sequences. We use four challenging test sequences (PAPER1, PAPER2, FACEOCC1, and FACEOCC2) to evaluate our proposed method. The self-made PAPER1 and PAPER2 sequences were chosen to challenge the methods on a fast moving deformable object under major scale changes. The sequences have a spatial resolution of 960×540 and consist of 563 and 726 frames respectively. The captured object (paper) is textured with patches of similar appearance and shape. The similar appearing features were chosen to stress the relational structure of our

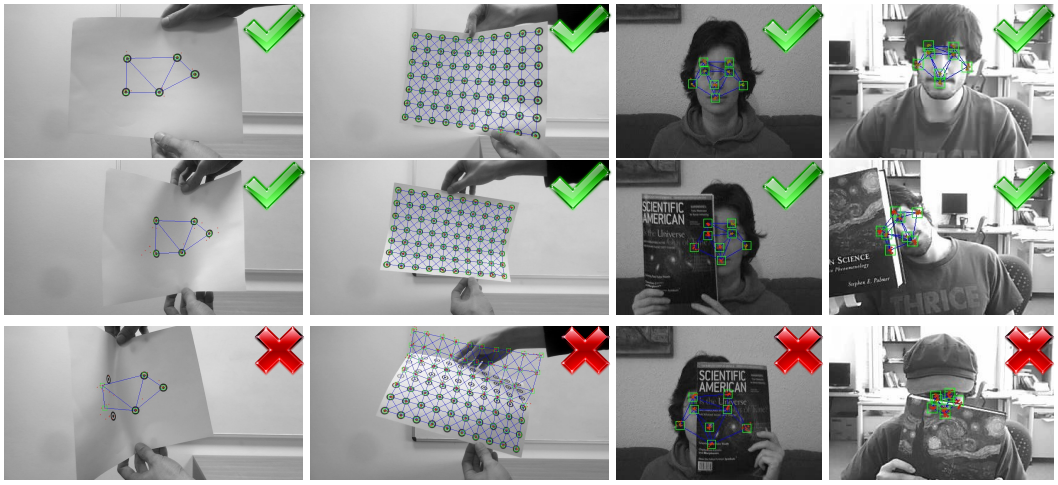


Figure 8.9: Datasets and tracking results for our proposed method: PAPER1, PAPER2, FACEOCC1, FACEOCC2 (from left to right). First two rows: successful tracking; third row: tracking failure cases.

tracker model. Thus the only way to distinguish the features is by considering the relative position of the feature patches to each other. The PAPER1 sequence consists of five feature patches with a carefully chosen position pattern which allows unique identification of the features by only having knowledge about the relative distances of the features to each other. The PAPER2 sequence is more challenging since the number of features is increased to 70 and the features are arranged in a grid structure allowing local relational ambiguities. The FACEOCC1 and FACEOCC2 sequences from Babenko *et al.* (2011); Duan *et al.* (2012) are designed for evaluating object trackers under major occlusions. The sequences have a spatial resolution of 352×288 (FACEOCC1) and 320×240 (FACEOCC2) and both consist of 888 frames each. While the FACEOCC1 sequence has only slow object movements, but showing substantial occlusions, the FACEOCC2 sequence challenges with fast movements, illumination changes, object rotation and substantial occlusions. The sequences and tracking results are shown in Figure 8.9.

Parameter selection. Parameter selection can be split into two parts. The first part consists in MRF model parameter selection. Since the proposed model is relatively robust to changes in α, we set α in an ad-hoc fashion for each sequence as follows: $\alpha = 20$ for PAPER1 and PAPER2 and $\alpha = 50$ for FACEOCC1 and FACEOCC2. For the HOG features we set the smallest scale pyramid resolution to 50×50. This leads to 3 scales for FACEOCC1 and FACEOCC2 and 4 scales for PAPER1 and PAPER2.

The second part is parameter selection for the PBP framework. We use $N = 20$ PBP iterations and $p = 10$ particles for each node. With this setting both algorithms (MH-PBP and S-PBP) converge well. Since we compare the overall *sampling* behaviour of the proposed method rather than the *belief propagation* convergence behaviour selecting these parameters should be uncritical.

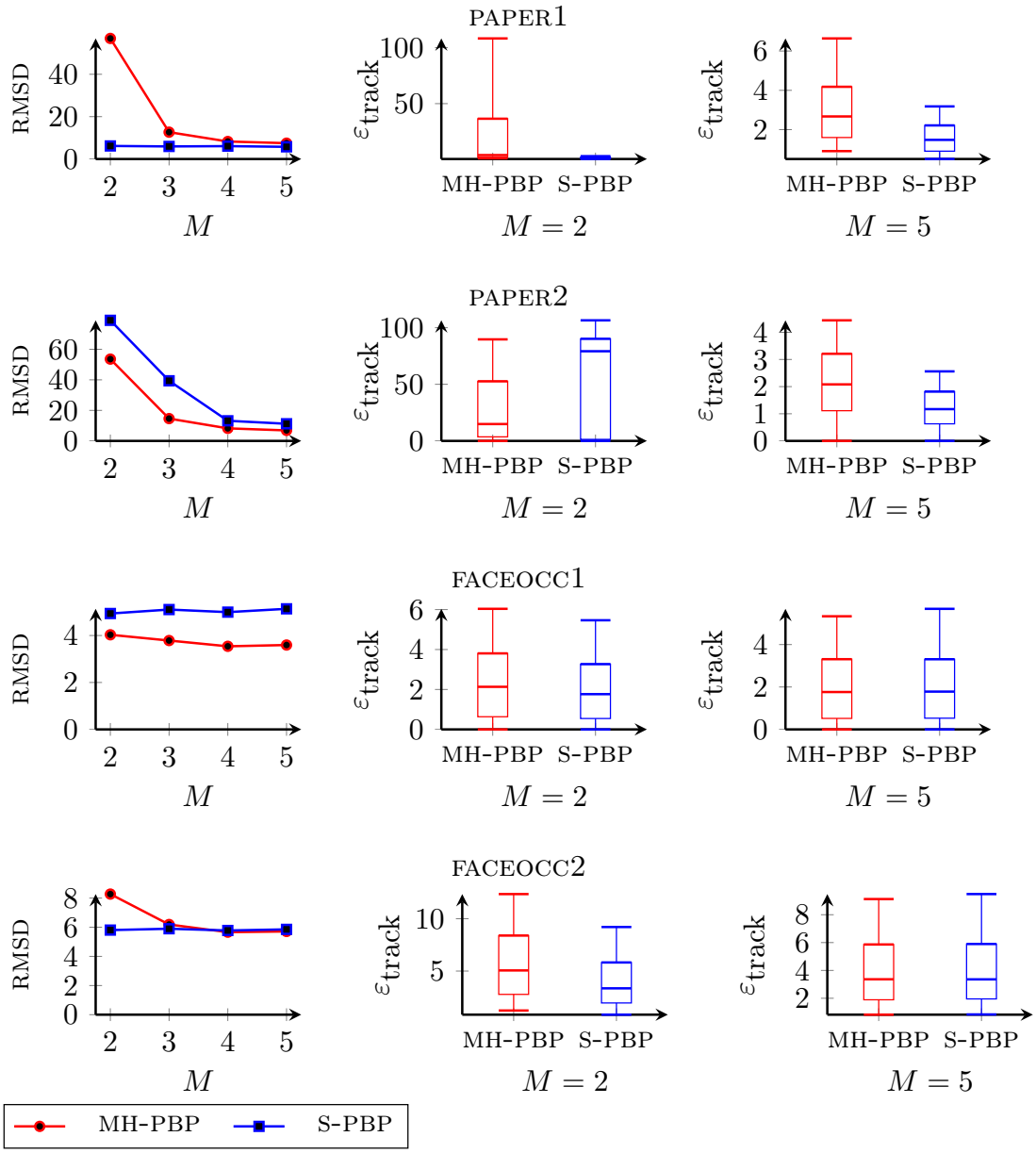


Figure 8.10: Relational feature tracker evaluation results showing the overal RMSD (for MCMC iterations from 2 to 5) and box plots over the error distance to groundtruth for selected MCMC iterations.

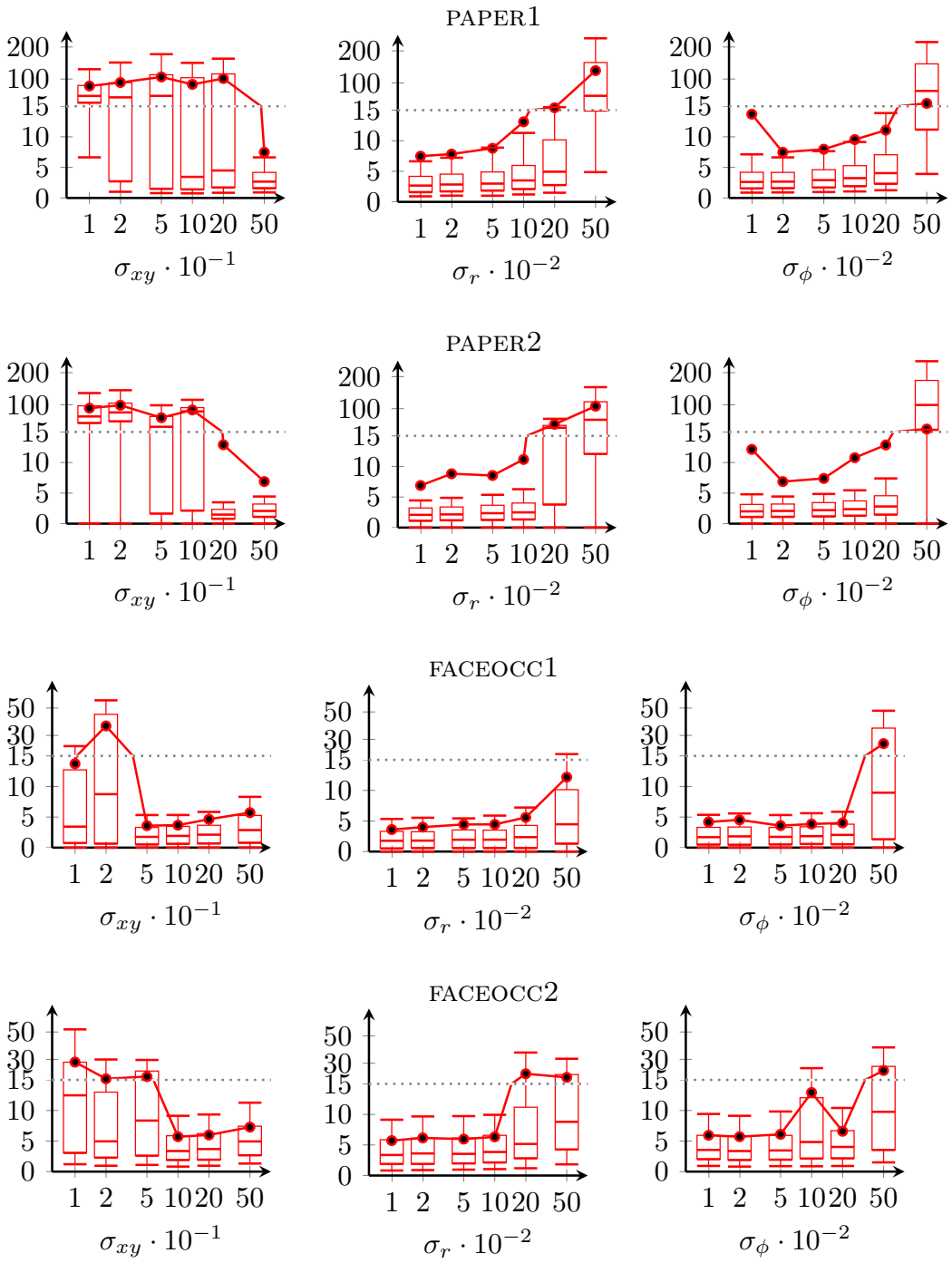


Figure 8.11: Optimal parameter evaluation for MH-PBP method (with $M = 5$). The vertical axis shows the error distance to groundtruth in px. Note that the vertical axis is stretched for error values lower than $15px$ in order to better visualize performance differences.

Evaluation metrics. We consider the distance $\varepsilon_{\text{track}}$ between the estimated feature position and the groundtruth (manually labeled) position as a quality measure. From this measure we derive two metrics: The rooted mean of squared distances (RMSD) and a quantile box-plot (10%, 25%, 50%, 75%, and 90% quantiles). While the first metric is very sensitive to outliers, the second metric provides more information about the overall error distribution.

Discussion. The evaluation results comparing S-PBP with MH-PBP using different MCMC iterations are shown in Figure 8.10. For MH-PBP, the MH sampling parameters $\{\sigma_{xy}, \sigma_r, \sigma_\phi\}$ are chosen (from the set $\{0.1, 0.2, 0.5, 1.0, 2.0, 5.0\} \times \{0.01, 0.02, 0.05, 0.10, 0.20, 0.50\} \times \{0.01, 0.02, 0.05, 0.10, 0.20, 0.50\}$) such that the RMSD is minimized. Note that for S-PBP such parameter tuning is not necessary. We have evaluated the tracking performance for different MCMC iterations $M = 2$ to 5. The box plots in Figure 8.10 show that S-PBP outperforms or performs equally well as MH-PBP for all tested sequences except for sequence PAPER2 with only 2 (and 3) MCMC iterations where both methods fail. This is mainly due to a much higher overall sampling noise of the MH-PBP method compared to S-PBP. We observed that the sampling noise of S-PBP is much less than with MH-PBP at feature positions with high confidence (i.e. high belief). On the other hand the sampling noise of S-PBP increases for uncertain feature positions. The RMSD in sequence PAPER2 and FACEOCC1 is higher for S-PBP than for MH-PBP due to temporal tracking failures. These tracking failures are caused by strong local deformations or by occlusions of many feature points. Typical tracking failures are depicted in the bottom row of Figure 8.9. It can be observed in such cases that S-PBP leads to much higher tracking error than MH-PBP due to broader particle sampling in uncertain feature positions.

Figure 8.11 shows an evaluation of MH-PBP under differing (non-optimal) sampling parameters. To this end, we vary each of the three sampling parameters individually and let the other two parameters stay fixed at their optimal values. Note that the estimation error varies highly, where very high values (usually > 15) indicate a tracking failure. In order to visualize both the performance differences for near-optimal parameters and tracking failures, the error values below and above the 15 mark are shown with a differing vertical axis scaling. In Figure 8.11, comparison for PAPER1, PAPER2, FACEOCC1, and FACEOCC2 is shown. It can be observed that the tracking performance of MH-PBP strongly depends on careful parameter selection. The parameter σ_{xy} has the highest impact on the tracking performance and the optimal parameter value varies strongly between sequences ($\sigma_{xy} = 5$ for PAPER1 and $\sigma_{xy} = 0.5$ for FACEOCC1). Selecting σ_{xy} is a compromise between allowing fast object motions and reducing overall localization noise. Selecting σ_r and σ_ϕ has analogous effects on changes in object scaling and rotation. This way one has to incorporate *prior knowledge* about the object motion in order to obtain good tracking results using MH-PBP. Tracked sequences and further comparisons are provided in the supplemental material.

The computational complexity for MH-PBP is $\mathcal{O}(NSpM\ (1 + Vp))$ and for S-PBP is $\mathcal{O}(NSpM(3 + 2Vp))$ given the number of PBP iterations N, nodes S, particles p, MCMC iterations M and the average number of neighbors per node V. This indicates a doubling of computation time of S-PBP compared to MH-PBP which is due to the overhead introduced for computing the interval bounds A. A look at the CPU times using fixed parameters for both al-

gorithms ($M = 5$, $p = 10$, $N = 20$) verifies this finding: FACEOCC: 0.69 s/frame (S-PBP) vs. 0.33 s/frame (MH-PBP) ; PAPER2: 7.43 s/frame vs. 3.66 s/frame. Nevertheless we have shown that S-PBP needs significant less MCMC iterations than MH-PBP such that the computational overhead can be typically well compensated.

8.7 Conclusion

We presented a novel particle belief propagation algorithm using slice sampling (S-PBP) instead of Metropolis-Hastings. We exploit the message passing equations to compute the slice sampling bounds, provided the unary and binary potentials are defined by analytic functions or can be bounded by one. We showed on a toy example that S-PBP outperforms MH-PBP in terms of MCMC chain mixing performance. Furthermore we showed that our approach performs equally well or better than MH-PBP on challenging relational feature tracking sequences.

Acknowledgments

The work is funded by the ERC-Starting Grant (DYNAMIC MINVIP). The authors gratefully acknowledge the support.

Part II

MEDICAL IMAGE ANALYSIS

Chapter 9

Multi-Region Labeling and Segmentation Using a Graph Topology Prior and Atlas Information in Brain Images

Medical image segmentation and anatomical structure labeling according to the types of the tissues is important for accurate diagnosis and therapy. In this chapter, we propose a novel approach for multi-region labeling and segmentation, which is based on a topological graph prior, registration of the labels, and the topological information of an atlas, using a multi-level set energy minimization method. We consider topological graph prior and atlas information to evolve the contour based on a topological relationship presented via a graph relation. This novel method is capable of segmenting adjacent objects with very close gray level that would be difficult to segment correctly using standard methods. The topological graph is registered from the low resolution and noisy source image to the topological information of an atlas to obtain region labeling. We explain our algorithm and show the graph prior and label registration techniques to explain how it gives precise multi-region segmentation and labeling. The proposed algorithm is capable of segmenting and labeling different regions in noisy or low resolution brain MRI images of different modalities. We compare our approach with other state-of-the-art approaches for multi-region labeling and segmentation. An earlier version of this chapter appeared in Computerized Medical Imaging and Graphics Journal (CMIG) (Al-Shaikhli *et al.*, 2014c).

9.1 Introduction

Multi-region image segmentation is a major task in medical imaging and it is important in diagnosis and therapy (Shattuck *et al.*, 2001). Due to poor resolution and weak contrast, image segmentation is difficult in the presence of noise and artifacts (Andrews *et al.*, 2011). Many existing methods for segmentation are based on image intensity information, shape proper-

ties or shape priors (Andrews *et al.*, 2011; Chan & Vese, 2001; Li *et al.*, 2011; Mishra *et al.*, 2009). Many researches addressed that the medical imaging systems like MRI, although it relatively provides high-resolution anatomical details but the identification of tissue information is limited by several factors like noise and image non-uniformity due to magnetic field inhomogeneities (Shattuck *et al.*, 2001). This gives difficulties of the brain tissue labeling (Kapur *et al.*, 1996; Shattuck *et al.*, 2001). Manual labeling of brain structures is achieved using a lot of information including image intensities, anatomical landmarks, position relative to neighboring brain structures and global position within the brain (Nocera & Gee, 1997), which need long processing time. Therefore, the automatic labeling is necessary and to maintain it for brain tissue we consider a prior topological information and tissue labeling for the segmentation that give a precise knowledge about position, size and type of the brain tissue. The topological graph prior gives an abstract information about the organs in the medical images and the atlas gives useful information about the label of the organs. The labels transfer from the atlas to the target image after warping the atlas with the target image.

Okada *et al.* (2012) proposed multi-organ segmentation of the upper abdomen by finding the interrelations between the organs based on canonical correlation analysis. Suzuki *et al.* (2012) proposed an atlas based multi-organ segmentation and detection of missing organ in abdominal CT images. Shimizu *et al.* (2007) proposed simultaneous extraction of multiple organs from abdominal CT using abdominal cavity standardization process with feature database and atlas guided segmentation incorporating parameter estimation for organ segmentation. Linguraru *et al.* (2012) proposed multi-region segmentation using graph cut method for four abdominal organ segmentation. Kohlberger *et al.* (2011) proposed multi-organ segmentation from CT medical images using learning-based segmentation and shape representation. Bazin & Pham (2008) proposed multi-region segmentation algorithm of brain image using topological and statistical atlases of brain as prior to the segmentation framework. Nocera & Gee (1997) proposed tissue classification of cerebral magnetic resonance images using Bayesian estimation method. Fischl *et al.* (2002) proposed an automatic labeling of Neuroanatomical structures in the human brain by estimating the probability information from manual labeled training data. Soni (2007) proposed brain tissue classification of only three types of tissue (gray matter, white matter and CSF) using conditional random field for magnetic resonance images. Cocosco *et al.* (2003) proposed a full automatic brain tissue classification method for three types of brain tissue in magnetic resonance images by measuring a tissue probability map. Liu *et al.* (2011) proposed a method for image segmentation using multi-context label tree structure. Sabuncu *et al.* (2010) proposed a nonparametric image segmentation and label fusion approach using image registration. Aljabar *et al.* (2009) proposed a framework for brain tissues multi-atlas segmentation. The accuracy of this approach degrades in the presence of strong noise and by using single atlas information. Mansouri *et al.* (2006) proposed multi-region competition algorithm for intensity-based image segmentation. Vazquez *et al.* (2004) proposed image segmentation algorithm from the viewpoint of image data regularized clustering. Previous work of multi-region segmentation and labeling focused on either image region intensity or shape priors or multi-atlas information for segmentation, however the shape of brain tissues may vary from person to person and the intensity differs according to the image modality. Furthermore, the performance of these works degrade in the presence of high level of noise.

In contrast to these works, our contribution of this work is multi-region segmentation and labeling using a multi-level set formulation which includes a topological graph prior and atlas information in an abstract fashion, in another words, topological information in the atlas like area, position and label that were mapped on a topological graph prior of the image. Therefore, we determine the location and the area of each region as well as the topological correlation and discrimination between different regions in the image. The graph prior is embedded in the multi-level set energy equation and acts as an additional prior term to identify both the overlapped regions and weak boundaries between adjacent regions in the image, as shown in Figure 9.1. The graph priors allow us to handle the huge variability of medical image data in a more abstract fashion. Consequently, our algorithm is less sensitive to noise and gives accurate segmentation of ambiguous regions depending on the atlas information and the topological correlation of different regions in the brain MRI image. For brain segmentation and labeling, we propose seven labels of brain tissue as shown in Figure 9.3 and Table 9.3. The outcome of our algorithm is conjoint of multi-class image segmentation and labeling. In all of our experiments, we concentrate on brain segmentation, however, it is worth noting that the method is general and can be applied to other scenarios, for example abdominal organ segmentation by computing the topological relationship of the abdominal organs using abdominal atlas. The organization of this work is as follows: Section 9.2 explains the proposed approach. The discussion of the experimental results is presented in Section 9.3. Finally, the conclusion is presented in Section 9.4.

9.2 Method

In this section we will explain our proposed method for multi-region segmentation and labeling based on a multi-level set formulation with an atlas information and a topological graph prior.

9.2.1 Graph Prior

Human body organs have specific topological correlations between them and according to these correlations, the exact location and boundary of these organs can be determined. If we consider the image B as sets of clusters (segments) $B = O_i, O_{i+1}, ..., O_N$ depending on the dissimilarity between them and χ_{O_i} is the membership function of each cluster. These clusters are connected with each other by a specific topological relationship then the topological graph of these clusters can give information like the area, the location and the topological relationship of each cluster in the image. The topological graph is constructed from test image to provide the prior knowledge to the segmentation process. The clusters in the topological graph of the image B are determined using *Otsu's* method (Otsu, 1979) and these clusters are labeled according to their topological relationship. We consider three types of topological relations (disjoint, contact, inside). Egenhofer & Herring (1990) have defined and computed O° as an interior of the cluster, O^c as a complement (exterior) of the cluster and ∂O as a boundary of the cluster. The topological region relationship (TRL) between the clusters is calculated in terms of probability of intersections of these clusters in a 9-intersection model in 3×3 matrix (Egenhofer &

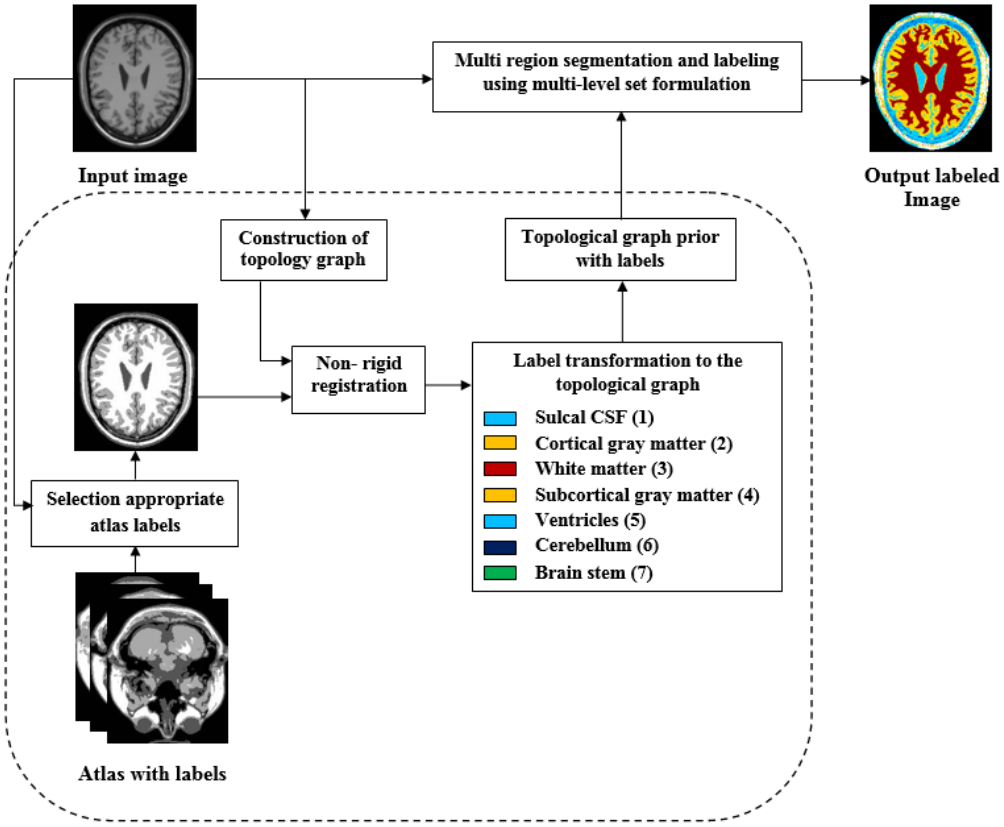


Figure 9.1: Block diagram explain proposed algorithm for multi-region labeling and segmentation.

Herring, 1990), as follows:

$$TRL(O_i, O_{i+1}) = \begin{pmatrix} a_{11}\left(O_i^{\circ} \bigcap O_{i+1}^{\circ}\right) & a_{12}\left(O_i^{\circ} \bigcap \partial O_{i+1}\right) & a_{13}\left(O_i^{\circ} \bigcap O_{i+1}^{c}\right) \\ a_{21}\left(\partial O_i \bigcap O_{i+1}^{\circ}\right) & a_{22}\left(\partial O_i \bigcap \partial O_{i+1}\right) & a_{23}\left(\partial O_i \bigcap O_{i+1}^{c}\right) \\ a_{31}\left(O_i^{c} \bigcap O_{i+1}^{\circ}\right) & a_{32}\left(O_i^{c} \bigcap \partial O_{i+1}\right) & a_{33}\left(O_i^{c} \bigcap O_{i+1}^{c}\right) \end{pmatrix} \tag{9.1}$$

Each element in Eq. (9.1) represents specific topological relationship. For example, to achieve the disjoint relation, we need to $a_{11} = 0$, $a_{12} = 0$, $a_{21} = 0$ and $a_{22} = 0$, which means that all pixels in cluster O_i are not in O_{i+1}. To achieve the inside relation between O_i and O_{i+1}, we need to $a_{21} = 1$, which means that the pixels in ∂O_i are in O_{i+1}°. To achieve the contact relation, we need to $a_{11} = 0$, $a_{22} = 1$, $a_{12} = 0$, and $a_{21} = 0$ which mean that the pixels in ∂O_i are in ∂O_{i+1}. The overlapped region is achieved by $a_{11} = 1$, $a_{12} = 1$, $a_{21} = 1$, and $a_{22} = 1$. This is summarized as follows:

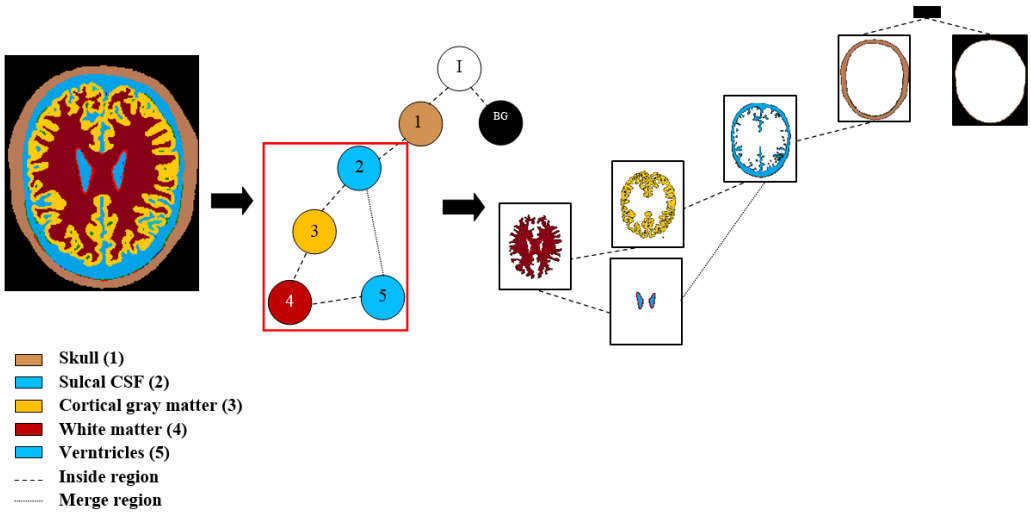


Figure 9.2: Example of image representation as topological graph.

$$\begin{cases} a_{11}=0, a_{12}=0, a_{21}=0, a_{22}=0 & \text{if } RL_{dis}(O_i, O_{i+1}) > 0 \\ a_{11}=0, a_{12}=0, a_{21}=0, a_{22}=1 & \text{if } RL_{con}(O_i, O_{i+1}) > 0 \\ a_{11}=0, a_{12}=0, a_{21}=1, a_{22}=0 & \text{if } RL_{in}(O_i, O_{i+1}) > 0 \\ a_{11}=1, a_{12}=1, a_{21}=1, a_{22}=1 & \text{if } RL_{ov}(O_i, O_{i+1}) > 0 \end{cases} \tag{9.2}$$

where RL_{dis}, RL_{con}, RL_{in}, and RL_{ov} are disjoint, contact, inside, and overlap region relationship respectively, as follows:

$$RL_{dis}(O_i, O_{i+1}) = 1 - \max_b\{|\chi_{O_i}(b) + \chi_{O_{i+1}}(b) - 1|\} \tag{9.3}$$

$$\begin{aligned} RL_{con}(O_i, O_{i+1}) = \quad & \min\{(1 - \max_b(|\chi_{O_i^\circ}(b) + \chi_{O_{i+1}^\circ}(b) - 1|)), \\ & \max_b(\min(\chi_{\partial O_i}(b), \chi_{\partial O_{i+1}}(b)))\} \end{aligned} \tag{9.4}$$

$$RL_{in}(O_i, O_{i+1}) = \min(1, \min_b(1 + \chi_{O_{i+1}^\circ}(b) - \chi_{O_i}(b))) \tag{9.5}$$

$$\begin{aligned} RL_{ov}(O_i, O_{i+1}) = \quad & \min\{\max_b(\min(\chi_{O_{i+1}^\circ}(b), \chi_{O_i^\circ}(b))), \\ & \max_b(\min(\chi_{O_i^\circ}(b), \chi_{\partial O_{i+1}}(b))), \\ & \max_b(\min(\chi_{O_{i+1}^\circ}(b), \chi_{\partial O_i}(b))), \\ & \max_b(\min(\chi_{\partial O_i}(b), \chi_{\partial O_{i+1}}(b)))\} \end{aligned} \tag{9.6}$$

Table 9.1 summarizes how each element of the matrix in Eq. (9.1) determines the relationship between the clusters by checking the primary conditions and the secondary conditions. The primary conditions are the main conditions to determine the topological relationship between the regions. The secondary conditions are proposed to be 'ones' in the matrix. We propose

the primary and secondary conditions because the topological properties between two region is achieved by how these regions are intersect with each other and this intersection achieved by the primary condition. The secondary conditions represent the intersection of each region with the complement of the other region which . For example, O_i and O_{i+1} are two regions in the image. To calculate the topological relationship of these regions, we need the primary conditions because they represents the intersection between them. Regarding to the secondary condition, if we choose a_{13} this condition represents the intersection of O_i with the complement of O^c_{i+1} and since O_i is part of O^c_{i+1}, therefore the intersection is always achieved in the secondary conditions and therefore we set them to 'ones' in the matrix. Figure 9.2 explains the representation of the anatomical structures in the image as topological graph. The red rectangle in Figure 9.2 shows that the labels that are considered in our calculation depend on the atlas labels illustrated in Figure 9.2 (as we will explain in the next subsection), in another words, the labels may consist of any combination depending on the slice level of the brain image. Table 9.2 shows TRL of each region in the image of Figure 9.2. The connected components represent the total relationship of each region. The number of cavities in each region indicates how many regions are inside it or held by it. For example, in case of region white matter (WM) in Figure 9.2, it has two relationships (two connected components), 1) TRL_1(WM,GM) and 2) TRL_2(WM,ventricles) as in Eq. (9.7). TRL_1(WM,GM) shows that WM is inside GM, while TRL_2(WM,ventricles) shows that WM covers ventricles according to Eq. (9.1):

$$TRL_1(WM, GM) = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix}, TRL_2(WM, ventricles) = \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix} \quad (9.7)$$

$$TRL_{total}(WM, R^{WM}) = TRL_1(WM, GM) + TRL_2(WM, ventricles) = \begin{pmatrix} 0 & 1 & 2 \\ 1 & 0 & 2 \\ 2 & 2 & 2 \end{pmatrix} \quad (9.8)$$

where TRL_{total} is the total topological relationship, R^{WM} are the regions that have topological relationship with (WM). In Eq. (9.8), $a_{12} = 1$ indicates that the region WM covers ventricles while $a_{21} = 1$ indicates that WM is inside GM. In our calculation, we consider the primary conditions, which are explained in Table 9.1, to compute the topological relation of the regions. The secondary conditions set as ones.

The topological similarity T_s between each cluster in the topological graph and the corresponding region in the image during evolution are for updating the labels of pixels of each region in the image at each t during evolution process, i.e. it is applied iteratively during curve evolution to update the label of each pixel in an image. T_s is determined by subtraction $TRL_{total}(O_i), R^{O_i}$ from $TRL_{total}(R_{\overrightarrow{\gamma_i}}), R^{R_{\overrightarrow{\gamma_i}}})$ during evolution process given by:

$$T_s(O_i, R_{\overrightarrow{\gamma_i}}) = \begin{cases} 0 & \text{for} \quad TRL_{total}(O_i, R^{O_i}) = TRL_{total}(R_{\overrightarrow{\gamma_i}}, R^{R_{\overrightarrow{\gamma_i}}}) \\ 1 & \quad otherwise \end{cases} \quad (9.9)$$

where $(R_{\overrightarrow{\gamma_i}})$ are the regions inside their contours during evolution process. R^{O_i} and $R^{R_{\overrightarrow{\gamma_i}}}$ are

Table 9.1: Topological properties of different regions of the image according to Eqs. (9.1) and (9.2).

$TRL(O_i, O_{i+1})$	**Primary conditions**	**Secondary conditions**
Contact regions	$a_{22} = 1$, $a_{11} = 0$, $a_{12} = 0$, $a_{21} = 0$	$a_{13} = 1$, $a_{23} = 1$, $a_{31} = 1$, $a_{32} = 1$, $a_{33} = 1$
Inside regions	$a_{21} = 1$, $a_{11} = 0$, $a_{12} = 0$, $a_{22} = 0$	$a_{13} = 1$, $a_{23} = 1$, $a_{31} = 1$, $a_{32} = 1$, $a_{33} = 1$
Disjoint regions	$a_{11} = 0$, $a_{12} = 0$, $a_{21} = 0$, $a_{22} = 0$	$a_{13} = 1$, $a_{23} = 1$, $a_{31} = 1$, $a_{32} = 1$, $a_{33} = 1$

Table 9.2: Topological properties of different regions of the image in Figure 9.2.

Region label	**Region name**	**#of connected components / (region name)**	**Internal cavity/Handles**
1	Skull	2 / (bg), (sulcal CSF)	1/ (Sulcal CSF)
2	Sulcal CSF	2 / (skull), (WM)	1 / (GM)
3	GM	2 / (sulcal CSF), (WM)	1 / (WM)
4	WM	2 / (GM), (Ventricles)	1/ (Ventricles)
5	Ventricles	2 / (sulcal CSF), (WM)	0

the regions that have topological relationship with O_i and $R_{\overrightarrow{\gamma_i}}$ respectively. The area and the centroid of each contour are calculated at each t during evolution process and compared with the area and the centroid of corresponding cluster in the topological graph:

$$A_i = \int_A dA, \qquad C_{xi} = \frac{1}{A}\int_A x_e \ dA, \qquad C_{yi} = \frac{1}{A}\int_A y_e \ dA \tag{9.10}$$

where A_i are the areas, C_{xi} and C_{yi} are the coordinates of centroid, x_e and y_e are coordinates of the centroid of the differential element of area dA. The prior information is added to the functional energy as topological graph prior term:

$$E_g[(\overrightarrow{\gamma_i})_{i=1}^{N-1}] = \underbrace{\alpha\left(\int_{R_{\overrightarrow{\gamma_i}}} (|A_{O_i} - A_{R_i}|)\, dx + (|C_{O_i} - C_{R_i}|) + T_s\right)}_{Topological \ \ graph \ \ prior \ \ term} \tag{9.11}$$

E_g is the energy of the topological graph. α is constant ($\alpha = 1 \ \ or \ \ 0$) to run the algorithm with or without topological graph prior. A_{O_i}, C_{O_i} are the area and centroid of the clusters in topological graph and A_{R_i},C_{R_i} are the area and the centroid of the regions in the image B during the evolution process respectively.

As mentioned above, the accuracy of the segmentation depends on the accuracy of the extraction of the topological graph information which may be affected in the presence of strong noise. Therefore we propose to use an atlas information registration as an additional prior information to solve this problem, as we will explain in the next sections.

9.2.2 Construction of Atlas Template and its Topological Properties

The segmentation accuracy depends on the accuracy of the topological graph prior extraction. In the presence of strong noise, the extraction of the topological graph will be affected. Therefore, we propose to use an atlas information to increase the accuracy of the topological graph in presence of strong noise.

The atlas template T is constructed using the average of 30 brain dataset from brainweb (Cocosco *et al.*, 1997). The average atlas template is calculated using average brain model (Guimond *et al.*, 2000). The topological properties of the atlas are summarized in Figure 9.3 and Table 9.3. Figure 9.3 shows the topological graph of the atlas template with the label of each region and Table 9.3 explains the topological properties of the atlas template, the first column represents how many regions are connected to the region of interest. The second column represents how many cavities available in the region of interest or held by it. For example, the white matter (label 3), there are three regions are connected to it (cortical gray matter (label 2), subcortical gray matter (label 4) and ventricles (label 5)) and it has two cavities so it holds two regions (subcortical gray matter and ventricles).

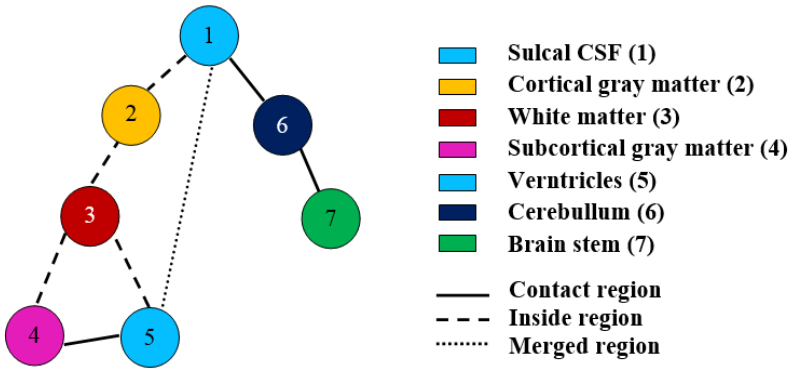


Figure 9.3: Topological graph of the atlas template with its labels.

9.2.3 Selection of Appropriate Atlas Template

We consider the graph similarity (object similarity and topological relationship similarity) between the test image and the atlas template. The topological graph in T are constructed by measuring the objects or regions in each image in atlas template using *Otsu's* method (Otsu, 1979) and then determine the topological relationship between these regions:

$$T = O_1, O_2, \cdots, O_N \tag{9.12}$$

where O are the objects in T, $N = 7$ is the total number of the objects in the atlas template. Assuming that the topological properties of the atlas template are summarized in Figure 9.3 and Table 9.3, while the topological properties of B are calculated as explained in Section 9.2.1. After constructing the topological graph for both the atlas template (atlas training data) and

Table 9.3: Topological properties of the atlas template.

Tissue label	tissue type	#of connected components	Internal Cavity/ Handles
1	Sulcal CSF	3	1/ (GM)
2	Cortical gray matter (GM)	2	1/ (WM)
3	White matter (WM)	3	2/ (Ventricles), (Subcortical GM)
4	Subcortical gray matter	2	0
5	Ventricles	3	0
6	Cerebellum	2	0
7	Brain stem	1	0

the input image, the similarity between T and B with associated graphs $G_1(O_B, E_B)$ and $G_2(O_T, E_T)$ is measured. O_B are the objects in B, O_T are the objects in T. E_B and E_T are the edges between the objects in both B an T respectively. As explained earlier, in the presence of strong noise the edges in B are not define quite enough. Therefore, the object similarity and the topological similarity are considered to determine the overall similarity between B and T as follows:

$$O_s(O_n, O_i) = \frac{\sum\limits_{O_n \in T,\, O_i \in B} (w_{O_n} w_{O_i} O_e(O_n, O_i))}{\sum\limits_{O_n \in T} w_{O_n} \sum\limits_{O_i \in B} w_{O_i}} \tag{9.13}$$

$$O_e(O_n, O_i) = 1 - \max_u \{|\chi_{O_n}(u(x,y)) - \chi_{O_i}(u(x,y))|\} \tag{9.14}$$

where O_s is the object similarity, $n = 1, 2, ..., N$, w_{O_n} and w_{O_i} indicate the importance given to O_n and O_i while computing the similarity. O_e is the object equality. $u(x, y) \in O_n$ and O_i.

The topological similarity between B and T is determined by measuring the similarity of $TRL_{total}(O_n, R^{O_n})$ and $TRL_{total}(O_i, R^{O_n})$. The similarity function SIM of the topological graph of B and the atlas template T is:

$$S(O_n, O_i) = O_s(O_n, O_i) + TRL_{total}(O_n, O_i) \tag{9.15}$$

9.2.4 Registration of the Atlas Information and Label Transformation

The human organs are non-rigid organs and normally there is a relative shape difference from person to person. Sometimes due to high level of noise or low image resolution the topological graph gives limited information about the regions in the image. Therefore a multi-modality non-rigid *demon* algorithm is proposed to use for image registration and label transformation from the atlas to the target image (Kroon & Slump, 2009). The registration model consists of

a similarity function, a transformation error function and a smoothing regularization:

$$E_r = \frac{1}{2}\|T - B \circ (S + U)\|^2 + \frac{\sigma_i^2}{\sigma_x^2}\|U\|^2 \tag{9.16}$$

$$\nabla E_r = (T \circ S - B)\left(\frac{\nabla T}{|\nabla T|^2 + \beta^2 (T \circ S - B)^2}\right) \tag{9.17}$$

where E_r is the registration error, T is the labeled atlas transformed image, B is the target image, S is the transformation field, U is the update of the transformation field. β is the normalization factor, σ_i and σ_x are constants of intensity uncertainty (image noise) and transformation uncertainty, for more details see (Kroon & Slump, 2009). A scale space approach is used to avoid local minimum and to speed up the registration. The image is resized to 8×8 pixels and these small size images are registered, then the original image and the found transformation fields are resized to 16×16, until reaching the original size of the image.

The mutual information (MI) is used as a similarity measurement of the intensity and texture in both T and the topological graph of B. MI represents the degree of dependency of B and T and it measures the degree of alignment between B and T. Then the labels are transformed from T to B.

$$MI(B,T) = \sum_{B,T} p(b,t) log\left(\frac{p(b,t)}{p(b)p(t)}\right) \tag{9.18}$$

where $p(b)$ and $p(t)$ are the probability of the gray values in B and T respectively. $p(b,t)$ is the joint probability of the images gray values which is derived from the joint histogram.

The labels transformation from the atlas template T to the topological graph of the image B is based on joint histogram peaks:

$$H(\lfloor B(x,y)NB \rfloor, \lfloor T(x,y)NB \rfloor) = H(\lfloor B(x,y)NB \rfloor, T(x,y)NB \rfloor) + 1 \tag{9.19}$$

where $H(B,T)$ is the joint histogram of B and T, NB is the number of bits (NB=255), B and $T \in$ [0,1]. (x,y) is the pixel location. In Eq. (9.19), $H(B,T)$ will pass through all pixel locations. The labels are transformed from the atlas template to the topological graph by finding the gray value for every pixel in the image B which overlaps with pixel value in the image T:

$$B_l(x,y) = argmax_t\left(\lfloor H(b(x,y)NB \rfloor, \lfloor tNB \rfloor)\right) \tag{9.20}$$

After label transformation from the atlas to the topological graph of the target image, Eq. 9.11 should be rewritten in a form of labeled topological graph prior:

$$E_{gl}[(\overrightarrow{\gamma_i})_{i=1}^N] = \underbrace{\alpha\left(\int_{R_{\overrightarrow{\gamma_i}}} \left(|A_{O_{i_l}} - A_{R_i}|\right) dx + \left(|C_{O_{i_l}} - C_{R_i}|\right) + T_s\right)}_{Labeled\ \ topological\ \ graph\ \ prior\ \ term} \tag{9.21}$$

where l is the label of the region as explained in Figure 9.3 and Table 9.3.

9.2.5 Multi-level Set Formulation and Curve Evolution

This section describes the multi-level set method with labeled topological graph prior. Let $B = (\bigcup_{i=1}^{N} R_i)$ is the input image with N regions, $i \in [1, 2, ..., N]$. We assume that for each R_i there is its complement R_i^c:

$$R_{\overrightarrow{\gamma_i}}(t) = \{u \in R | \overrightarrow{\gamma_i}(u,t) > 0\},\ i = 1, ..., N \tag{9.22}$$

$$\{R_{\overrightarrow{\gamma_1}}(t), R_{\overrightarrow{\gamma_1}}(t)^c \cap R_{\overrightarrow{\gamma_2}}(t), R_{\overrightarrow{\gamma_1}}(t)^c \cap R_{\overrightarrow{\gamma_2}}(t)^c \cap R_{\overrightarrow{\gamma_3}}(t), ..., (\bigcup_{i=1}^{N} R_{\overrightarrow{\gamma_i}}(t))^c\} \tag{9.23}$$

The total Euler-Lagrange energy functional can be written as follows:

$$E_{total}[(\overrightarrow{\gamma_i})_{i=1}^{N}] = \int_{R_{\overrightarrow{\gamma_i}}} \omega_i(b)db + \int_{R^c_{\overrightarrow{\gamma_i}}} \psi_i(b)db + \lambda \oint_{\overrightarrow{\gamma_i}} ds + E_{gl} \tag{9.24}$$

where the first two terms are the data terms of the region. The third term is the regularization term and the fourth term is our proposed prior term. λ is positive real constant to weigh the relative contribution of the energy equation. ω_i are the data in R_i and ψ_i are the data in R_i^c.

The data term is modified to be constrained by T_s: $\omega_i(b) = [-\ln P_{\mu_i}(B(b)) = (B(b) - \mu_i)^2]$ and $\psi_i(b) = [-\ln P_{\mu_j}(B(b)) = (B(b) - \mu_j)^2]T_s$ where μ_i are the mean over $R_{\overrightarrow{\gamma_i}}$ and μ_j are the mean over $R^c_{\overrightarrow{\gamma_i}}$. According to Eq. 9.9 and Eq. 9.24, T_s represents the topological function of the label state of the set R_i^c.

To minimize Eq. (9.11) by curve evolution we compute:

$$\frac{d\overrightarrow{\gamma_i}}{dt} = -\frac{\partial E}{\partial \overrightarrow{\gamma_i}} \tag{9.25}$$

$\frac{\partial E}{\partial \overrightarrow{\gamma_i}}$ are the derivative of functional energy with respect to $\overrightarrow{\gamma_i}$ and they are computed as for the standard region computation functional in Zhu & Yuille (1996). Following Zhu & Yuille (1996), we get the evolution equation of the curves $\overrightarrow{\gamma_i}$:

$$\frac{\partial \overrightarrow{\gamma_i}}{\partial t} = -\left(\omega_i(b) - \psi_i(b) + \underbrace{\alpha[\Big(|A_{O_{i_l}} - A_{R_i}|\Big) + \Big(|C_{O_{i_l}} - C_{R_i}|\Big) + T_s]}_{Labeled\ \ topological\ \ graph\ \ prior} + \lambda k_i\right)\overrightarrow{n_i} \tag{9.26}$$

where k_i are the curvature of zero level set of $\overrightarrow{\gamma_i}$, $\overrightarrow{n_i}$ are the external unit normal of the curve, $i \in [1, ..., N]$, $j \in [1, ..., N]$ and $i \neq j$.

During curve evolution, the curves are constrained by the labeled topological graph prior term and the curvature term. For N-region segmentation, let $b(x, y)$ be pixels in the image B ($b(x, y) \in B$) and let $\overrightarrow{\gamma_i}(0)$ be an initial curve and $\overrightarrow{\gamma_i}(t)$ is a curve in an iteration t:

1. $A_{\overrightarrow{\gamma}_{R_i}}$ and $C_{\overrightarrow{\gamma}_{R_i}}$ are updated for each time step during evolution process and compared with the area and the centroid of the topological graph prior after registration and label

transformation from the atlas. The errors between A_{O_i} and $A_{\overrightarrow{\gamma}_{R_i}}$ and between C_{O_i} and $C_{\overrightarrow{\gamma}_{R_i}}$ should be minimized.

2. The topological similarity T_s defines the label state of the pixel $b(x, y)$ in each region in the image B at time $t + 1$ with respect to the label of the same pixel at time t. Using Eq. (9.9), if $T_s = 0$ at $\overrightarrow{\gamma}_i(t)$ and $\overrightarrow{\gamma}_i(t + 1)$ then $b \in \overrightarrow{\gamma}_i$ and this depends on the label state of the corresponding pixel in the labeled topological graph prior. If $T_s = 0$ at $\overrightarrow{\gamma}_i(t)$ and $T_s = 1$ at $\overrightarrow{\gamma}_i(t + 1)$ then $b \in \overrightarrow{\gamma}_j$, $i \neq j$, $i \in [1, ..., N]$ and $j \in [1, ..., N]$.

3. If b is a point of contact between two curves $(\overrightarrow{\gamma_i}, \overrightarrow{\gamma_j})$, then the curve will be constrained by the curvature term as follows: If the curvatures are positive $(k_i(b) \geq 0, k_j(b) \geq 0)$ this indicates that these curves are retract and not intersect. If $(k_i(b) \leq 0, k_j(b) \geq 0)$ this indicates that these two curves will be in the same direction but because $|k_i(b) \leq k_j(b)|$, the curve $\overrightarrow{\gamma_j}$ retracts faster than $\overrightarrow{\gamma_i}$ and the curves will not intersect. The graph constraint makes the partitioning more precise during evolution process by adding addition constrain information (T_s, area, centroid).

9.3 Experiments

To highlight the advantages of the proposed algorithm, the algorithm is tested in the presence of strong noise and with low resolution images for more than 300 images using the MedPix (Khalatbari, 1999), Wesky E Snyder (Snyder, 2002), the brain web for simulated brain database (Cocosco *et al.*, 1997) and other medical images from the Internet. Our algorithm is compared to state-of-the-art methods. The medical images in all databases are 2D MRI images and CT images. The sizes of the images are 181×217 and 512×512 for brain sections of MRI images and CT images. The experiments run in MATLAB using a 2.0 GHz Intel core I3 CPU. First, we will show the qualitative results of topological graph prior without atlas registration. Then we will show the qualitative results of topological graph prior with atlas registration. Finally we will show the quantitative evaluation of our algorithm.

9.3.1 Qualitative Results of Topological Graph Prior without Atlas Registration

The results of the first part of our algorithm show the improvement in the segmentation by using the topological prior information. This information help the contour, during the evolution process, to detect the ambiguous regions in the image. Figure 9.4 shows multi-region segmentation for abdominal and brain MRI images with and without topological graph prior. The ground truth is obtained by manual segmentation. Figure 9.4 also shows the improvements of our algorithm to capture the overlapped and close gray level regions according to its topological location in the image. The abdominal image in Figure 9.4 shows the improvements of our algorithm mainly in the segmentation of aorta, liver and diaphragm. The brain images show the segmentation of the cerebellum, brainstem, white matter and gray matters. In Figure 9.4, the segmented regions in the proposed algorithm are labeled by colors according to their topological relationship. The result of the algorithms proposed by Mansouri *et al.* (2006); Vazquez *et al.* (2004) are labeled manually to visualize the differences. The accuracy of this part of our

algorithm depends on the accuracy of the precise extraction of each cluster in the topological graph, i.e. T_s, A and C should be computed precisely for each cluster in the topological graph.

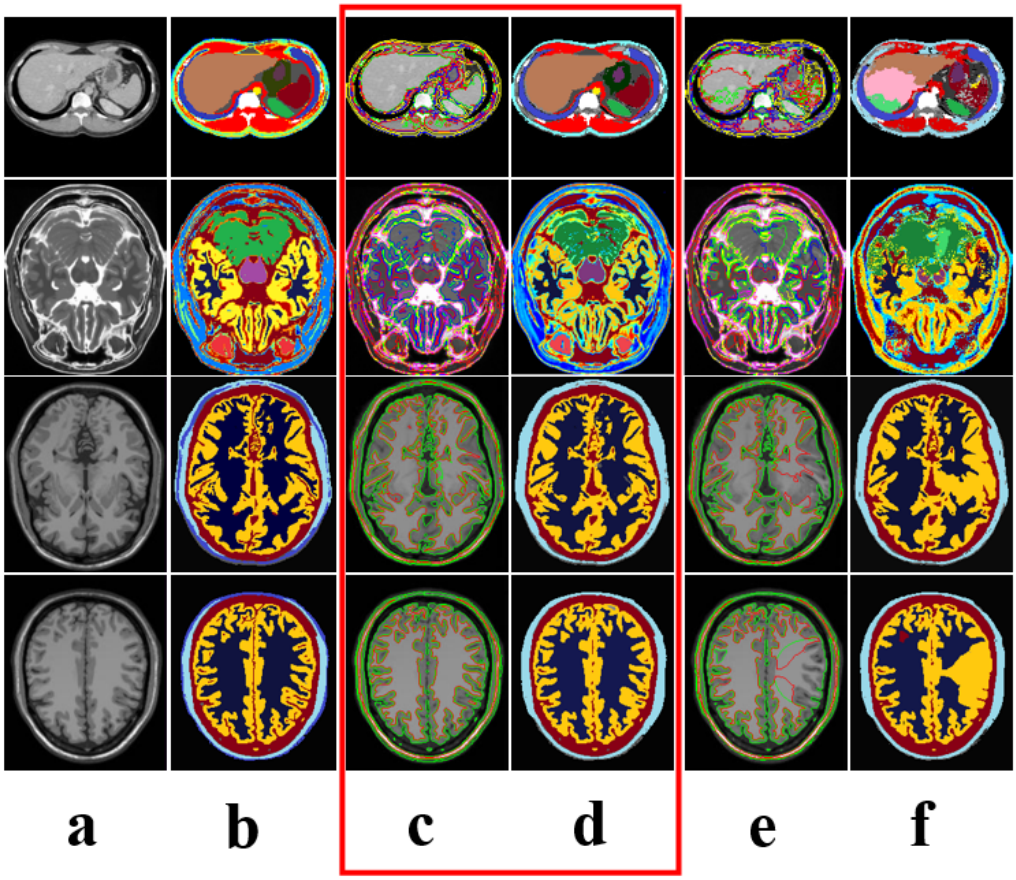


Figure 9.4: Multi-region segmentation results. (a) input images (b) ground truth, (c, d) proposed algorithm with graph prior and (e, f) without graph prior. $\lambda = 0.2$, iteration $= 70$. An example images from database (Cocosco *et al.*, 1997; Khalatbari, 1999; Snyder, 2002).

9.3.2 Qualitative Results of Topological Graph Prior with Atlas Registration

In this section, we will show the result of using the atlas information as an additional prior information with the topological graph prior. As mentioned previously, the topological graph prior may affected in the presence of high level of noise. This part of the proposed algorithm solves this limitation. We propose to use an atlas template to label the graph of the input image and eliminate the effect of noise. Figure 9.5 shows the results of using the atlas information with the topological graph compared with the results explained in Section 9.3.1 for multi-region segmentation as well as with the approaches proposed in Aljabar *et al.* (2009); Mansouri *et al.* (2006); Vazquez *et al.* (2004). In Figure 9.5 we observe the improvement of the segmentation in

the presence of noise. The atlas information provides an accurate extraction of the topological graph information which improves the multi-region segmentation and labeling in noisy and low resolution images with less computational time compared to the state-of-the-art methods as illustrated in Tables 9.4, 9.5 and 9.6. Figure 9.5 demonstrates the segmentation results in the presence of high level of noise; the black rectangle (a) shows the input noisy images and the ground truth segmentation, the red rectangle (b) shows the segmentation with topological graph prior and atlas information with accurate region segmentation and labeling. The green rectangle (c) shows the segmentation with topological graph prior. The blue rectangle (d) shows the result of atlas registration based segmentation, and the yellow rectangle (e) shows the segmentation without any prior information. The improvement of the segmentation of different brain tissues like white and gray matter, ventricles, and cerebellum can be seen. In Figure 9.5, the skull is not signed in the proposed labels and it is segmented and labeled randomly according topological relationship with the other regions in the image.

9.3.3 Quantitative Evaluation

To validate the accuracy of our algorithm, we compare our algorithm with other state-of-the-art methods (Aljabar *et al.*, 2009; Chan & Vese, 2001; Li *et al.*, 2011; Mansouri *et al.*, 2006; Vazquez *et al.*, 2004) using dice similarity coefficients (DSC) (Zou *et al.*, 2004). DSC is measured by computing the similarity between the ground truth segmentation and our algorithm as well as the methods proposed in Aljabar *et al.* (2009); Chan & Vese (2001); Li *et al.* (2011); Mansouri *et al.* (2006); Vazquez *et al.* (2004). A large DSC indicates higher accuracy:

$$DSC(I_{gt} - I_t) = \frac{2O(I_{gt} - I_{test})}{O(I_{gt}) + O(I_{test})} \tag{9.27}$$

where $O(I_{gt} - I_{test})$ is the number of overlapping pixels, $O(I_{gt}) + O(I_{test})$ is the summation of the number of pixel in each image.

We also employ the symmetric mean absolute distance (MAD) and Hausdorff distance (HD) (Wang *et al.*, 2009) between the resulting segmentation and the corresponding reference segmentation as additional metrics to evaluate the segmentation results. MAD is calculated by measuring the average distance from all points on the border of the automatically segmented brain tissue to the border of the reference segmentation. On the other hand, to assess the maximal local discrepancy between an automatic segmentation and reference segmentation, the symmetric Hausdorff distance between the border of the automatically segmented brain tissue and that of the reference segmentation is calculated. The smaller the MAD or Hausdorff distance, the better aligned the points on the two border and thus the better the agreement with the reference segmentation.

Figure 9.5 shows the improvement of our algorithm with and without noise using two databases (Khalatbari, 1999) and (Cocosco *et al.*, 1997) with respect to the state-of-the-art methods (Aljabar *et al.*, 2009; Chan & Vese, 2001; Li *et al.*, 2011; Mansouri *et al.*, 2006; Vazquez *et al.*, 2004). Our algorithm is robust with respect to the level of noise and the number of the segmented region. Figure 9.6 shows the stability of our algorithm as the number of the segmented region increases comparing with the other methods (Aljabar *et al.*, 2009; Chan &

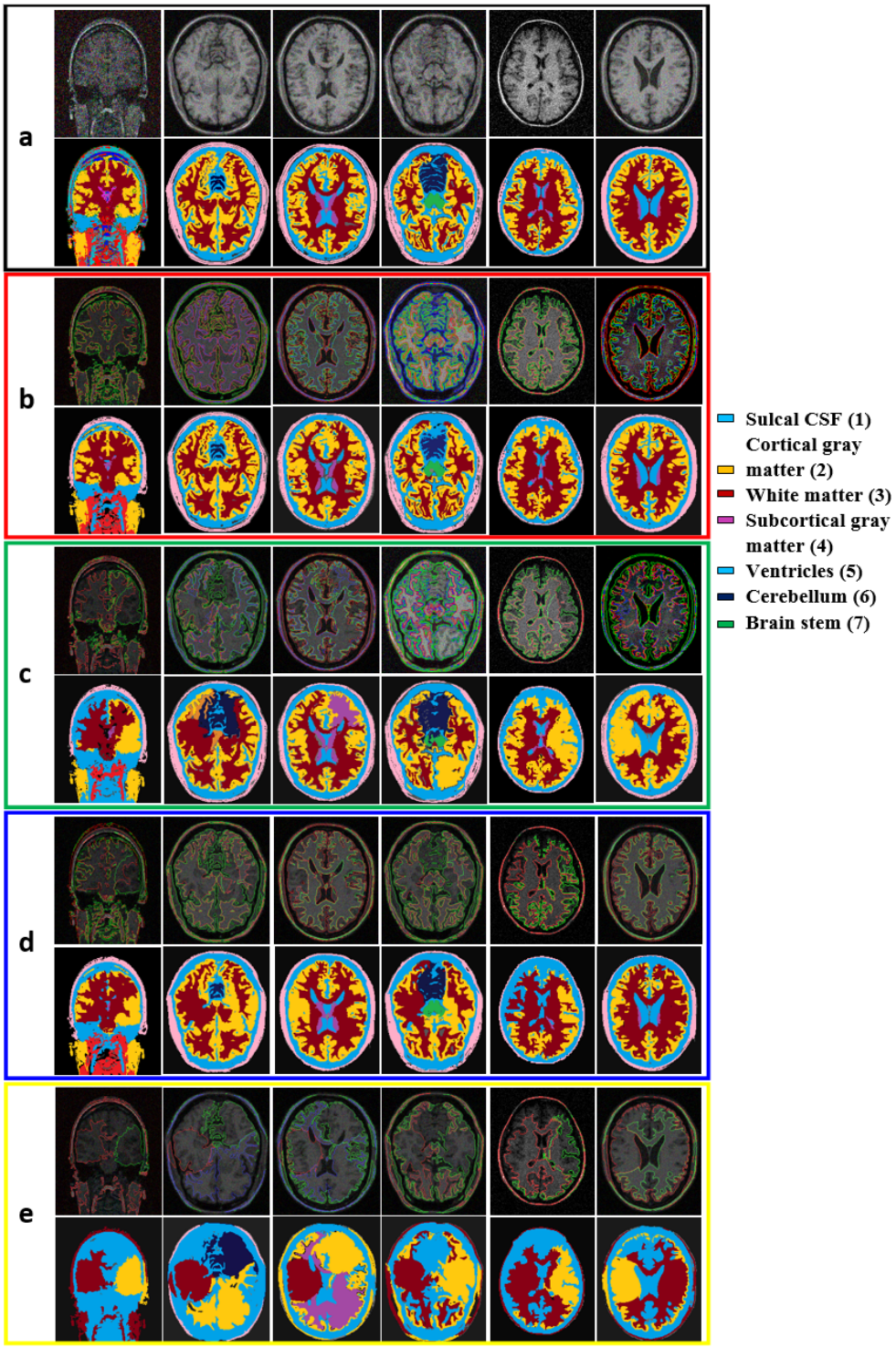


Figure 9.5: Examples of a multi-region labeling and segmentation using database (Cocosco *et al.*, 1997; Khalatbari, 1999), (a) are the input images and the ground truth, (b) are the segmentation with topological graph and atlas information, (c) are the segmentation with topological graph without atlas information, (d) are the segmentation with atlas information without topological graph, (e) are the segmentation without any prior (Mansouri *et al.*, 2006; Vazquez *et al.*, 2004), $\lambda = 0.2$, iteration $= 70$, noise $(SD = 0.16)$.

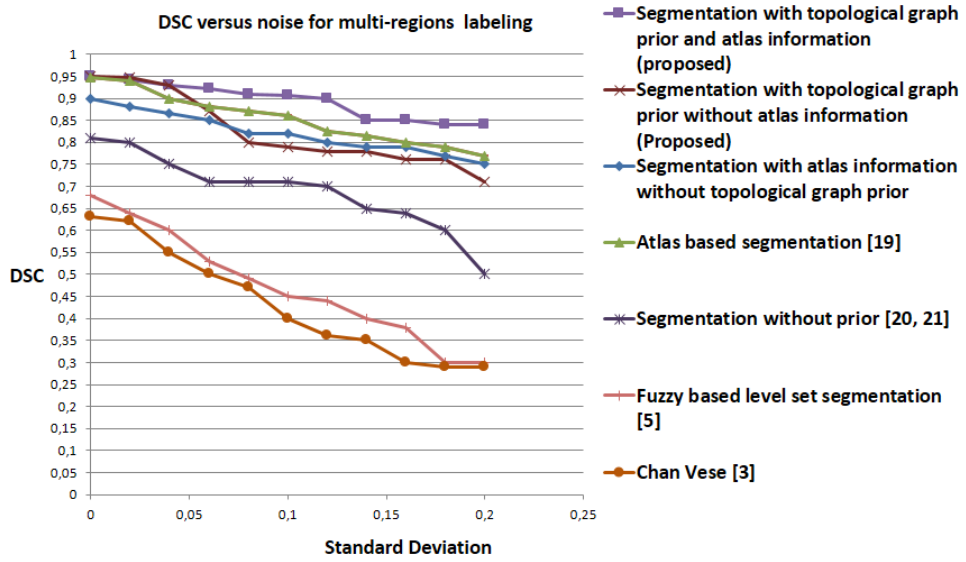
Vese, 2001; Li *et al.*, 2011; Mansouri *et al.*, 2006; Vazquez *et al.*, 2004). The proposed algorithm is based on a single topological atlas with the graph prior and it outperforms the other multi-atlas based segmentation (Aljabar *et al.*, 2009) which needs more training data and more computational time because if every atlas is registered with the target image, the computational time of segmentation increases linearly with the size of the training data. In Figure 9.6(a). we can see the performance of the proposed algorithm and the algorithm proposed in Aljabar *et al.* (2009) is quite similar for 3-region segmentation but with increase of the level of noise the performance of the proposed algorithm more robust than in Aljabar *et al.* (2009). Also as the number of the segmented region increases the performance of the algorithm (Aljabar *et al.*, 2009) decreases comparing with our algorithm as explained in Figure 9.6(b). Table 9.4 shows the accuracy (DSC) of our algorithm compared to the other methods for each database (Cocosco *et al.*, 1997; Khalatbari, 1999; Snyder, 2002). Table 9.5 and Table 9.6 show the overall accuracy (DSC, MAD, and HD) of our algorithm and other algorithms with and without presence of noise using the images in the databases (Cocosco *et al.*, 1997; Khalatbari, 1999; Snyder, 2002). For all these investigated scenarios, our algorithm outperforms other methods.

Table 9.4: Segmentation accuracy for each database without the effect of noise.

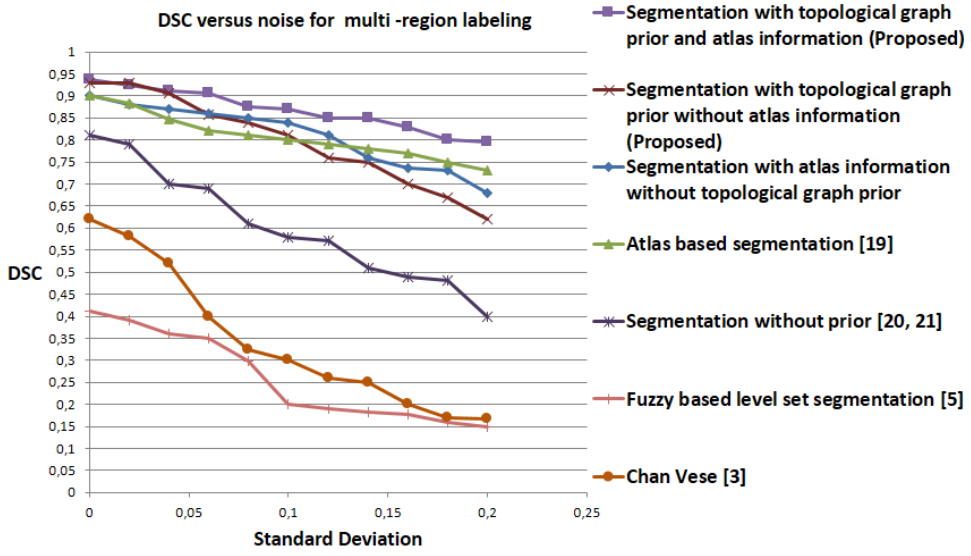
Algorithm	DSC Khalatbari	DSC Snyder	DSC Cocosco *et al.*
Topological graph prior with atlas inform.	94%	91.9%	95.5%
Topological graph prior without atlas inform.	93.56%	90.57%	94.88%
With atlas inform. without topological graph	88.5%	89.5%	92%
Atlas based segmentation (Aljabar *et al.*, 2009)	90%	87%	93%
Without graph prior (Mansouri *et al.*, 2006), (Vazquez *et al.*, 2004)	80.64%	79.89%	82.49%
Chan & Vese (2001)	61.82%	61.6%	62.78%
Level set fuzzy based (Li *et al.*, 2011)	40.87%	40.1%	42.63%

Table 9.5: Overall segmentation accuracy without the effect of noise of all images in database (Cocosco *et al.*, 1997; Khalatbari, 1999; Snyder, 2002) and the average computation time for each frame.

Algorithm	DSC	MAD	HD	# iteration	Time
Topological graph prior with atlas inform.	93.8%	0.66 mm	2.9 mm	70	2.45 min
Topological graph prior without atlas inform.	93%	0.68 mm	3.2 mm	70	2.24 min
Atlas inform. and without topological graph	90%	0.90 mm	3.9 mm	70	3.1 min
Atlas based segmentation (Aljabar *et al.*, 2009)	90%	0.93 mm	4.5 mm	70	2.9 min
Segmentation without graph prior (Mansouri *et al.*, 2006; Vazquez *et al.*, 2004)	81%	1.9 mm	5.5 mm	70	2.87 min
Chan & Vese (2001)	62%	3.2 mm	7.2 mm	400	5.15 min
Fuzzy based level set segmentation (Li *et al.*, 2011)	41.2%	5.3 mm	9.7 mm	300	3.2 min



(a) DSC versus noise for 3 regions segmentation



(b) DSC versus noise for 6 regions segmentation

Figure 9.6: Effect of Gaussian noise on segmentation performance of database (Cocosco *et al.*, 1997; Khalatbari, 1999; Snyder, 2002).

Table 9.6: Overall segmentation accuracy with the effect of noise (*standard deviation 0.16*) of all images in database (Cocosco *et al.*, 1997; Khalatbari, 1999; Snyder, 2002) and the average computation time for each frame.

Algorithm	DSC	MAD	HD	# iteration	Time
Topological graph prior with atlas inform.	83%	1.0 mm	4.3 mm	70	2.45 min
Topological graph prior without atlas inform.	70%	1.6 mm	5.0 mm	70	2.24 min
Atlas inform. and without topological graph	73%	2.5 mm	5.2 mm	70	3.1 min
Atlas based segmentation (Aljabar *et al.*, 2009)	77%	2.3 mm	5.1 mm	70	2.9 min
Segmentation without prior information (Mansouri *et al.*, 2006; Vazquez *et al.*, 2004)	49%	5.1 mm	6.8 mm	70	2.87 min
Chan & Vese (2001)	20%	7.3 mm	9.9 mm	400	5.15 min
Fuzzy based level set segmentation (Li *et al.*, 2011)	17.7%	7.5 mm	9.8	300	3.2 min

9.4 Conclusion

We propose to use a topological graph prior with atlas information in a multi-level set formulation for multi-region segmentation and partitioning. As a high-level prior, it gives accurate region partitioning with respect to their topological location and relationship as well as with the atlas information. The accuracy of the proposed approach depends on the accuracy of the extraction of the topological graph prior information which is achieved using the atlas information. The proposed algorithm has a less run time than other methods with high accuracy which beneficial to the field of brain image segmentation.

Chapter 10

Brain Tumor Classification Using Sparse Coding and Dictionary Learning

Brain tumor classification is considered as one of the most challenging tasks in medical imaging. In this chapter, a novel approach for multi-class brain tumor classification based on sparse coding and dictionary learning is proposed. We propose an individual (per-class) dictionary learning and sparse coding classification using K-SVD algorithm. This approach combines topological and texture features to build and learn a dictionary. Experimental results demonstrate that the sparse coding based classification outperforms other state-of-the-art methods. An earlier version of this chapter appeared at the IEEE International Conference on Image Processing (ICIP) (Al-Shaikhli *et al.*, 2014a).

10.1 Introduction

Early identification of brain tumors is important to treat the tumors effectively. Multi-class brain tumor classification is considered as one of the most important and challenging tasks in medical imaging due to the difficulty to extract the relevant information that can help to discriminate the tumor from the normal brain tissue (Sachdeva *et al.*, 2013). Brain tumor classification involves two steps, feature extraction and classification. Feature extraction is an essential step in the classification since the relevant information from the original image needs to be chosen in order to achieve high brain tumor classification accuracy (Gladis Pushpa Rathi & Palani, 2012). In general, brain tumors have different shapes and intensities from patient to patient (Sachdeva *et al.*, 2013), and sometimes, they also have different gray scales yet the same intensities as brain tissues (Sachdeva *et al.*, 2013). Therefore, features related to the shape or intensity create ambiguities during tumor classification (Sachdeva *et al.*, 2013).

Thiagarajan *et al.* (2013) proposed a sparse coding for brain tumor segmentation using intensity and location features. Bauer *et al.* (2011) developed a fully automatic algorithm for brain tumor segmentation and classification using a support vector machine (SVM) with a hi-

erarchical conditional random field. Han *et al.* (2011) proposed an algorithm for gliobastoma multiforme classification in the histological images based on dictionary learning and sparse coding. The sparse coding based classification was compared with the traditional kernel methods of classification. They concluded that the accuracy of kernel methods are better than sparse coding for histological images. Selvaraj *et al.* (2007) proposed an automatic classification technique based on Least Squares SVM to identify normal and abnormal slices of brain MRI images. Moon *et al.* (2002) proposed an automatic brain tumor segmentation based on statistical classification with a geometrical prior. Cocosco *et al.* (2003) proposed a fully automatic generation of correct training samples for MRI tissue classification. Weiss *et al.* (2013) proposed an approach for multiple sclerosis lesion segmentation using dictionary learning and sparse coding using intensity features. In the previous works, these approaches used either intensity-based or texture-based feature extraction for brain tumor classification, however a brain tumor may have the same intensity as normal brain tissue (Wu *et al.*, 2004). Furthermore, sparse representation has been shown to be an effective method for brain tumor classification by representing the images as dictionaries consist of linear combination of a few columns (atoms) of some redundant basis (Duarte-Carvajalino & Sapiro, 2009). While in the Linear-SVM, the data may not be linearly separable in the original feature space and needs higher dimensional space mapping to increase the classification accuracy which is computationally expensive (Han *et al.*, 2011).

In contrast to previous works, our **contribution** is a modified sparse coding and dictionary learning based multi-class classification. We proposed to use the K-SVD method to update both of the dictionary and sparse coding steps. Furthermore, due to the high degree of similarity in pixel intensities between normal brain tissue and tumor, and the variability of the tumor shape, location, and size, this variability justifies the use of topological and texture features to learn the dictionary. The topological feature gives information whether the case is normal or abnormal based on the assumption that the topology of normal brain is fixed. Therefore, the presence of tumor in the brain will change the normal brain topology. In addition, the texture features provide a good discrimination of the brain tumor types. The main novelty in our algorithm is the use of topology and texture features for learning, instead of applying learning directly on pixel values.

The rest of this work is organized as follows: Section 10.2 explains the proposed method. Section 10.3 discusses the results and Section 10.4 summarizes the work.

10.2 Method

10.2.1 Feature Extraction

In this subsection, the feature extraction step is explained by proposing a set of topological and texture features that give relevant information about the tumor.

Topological Matrix (TM) The proposed topological matrix is represented by a topological graph relationship and it considers the main feature to classify the normal and abnormal brain images by assuming that the topology of the normal brain is fixed. The topological graph is constructed from the input data to provide the feature knowledge to the classifier. To compute

TM, we consider an image I as sets of clusters depending on the dissimilarity between them $I = O_i, O_{i+1}, ..., O_N$. These clusters are connected with each other by a specific topological relationship. The clusters in the topological graph of the image I are computed using *Otsu's* method (Otsu, 1979) and the topological relationship of these clusters are computed using the method in Chapter 9 (Al-Shaikhli *et al.*, 2014c). Let O° be the interior of the cluster, ∂O be the boundary of the cluster, and χ_{O_i} is the membership function of each cluster. The topological relationship between the clusters is calculated in terms of probability of intersections of these clusters (Al-Shaikhli *et al.*, 2014c):

$$V_{TM}(O_i, O_{i+1}) = (m_{11}, m_{12}, m_{13}, \ldots, m_{33})^T \tag{10.1}$$

V_{TM} in Eq. (10.1) is a vector of zeros and ones and it is the sum of all individual V_{TM} that are computed for each region ($V_{TM} = \sum_{i=1}^{N} V_{TM_i}$). The elements (that have ones values) represent the topological relationship of each region in the image. In our calculation, we consider only four elements (m_{11}, m_{12}, m_{21}, and m_{22}) and the rest are set as ones:

$$\begin{cases} m_{11} = 0, m_{12} = 0, m_{21} = 0, m_{22} = 0 & \text{if } RL_{dis}(O_i, O_{i+1}) > 0 \\ m_{11} = 0, m_{12} = 0, m_{21} = 0, m_{22} = 1 & \text{if } RL_{con}(O_i, O_{i+1}) > 0 \\ m_{11} = 0, m_{12} = 0, m_{21} = 1, m_{22} = 0 & \text{if } RL_{in}(O_i, O_{i+1}) > 0 \\ m_{11} = 1, m_{12} = 1, m_{21} = 1, m_{22} = 1 & \text{if } RL_{ov}(O_i, O_{i+1}) > 0 \end{cases} \tag{10.2}$$

where RL_{dis}, RL_{con}, RL_{in}, and RL_{ov} are disjoint, contact, inside and overlap region relationship respectively as follows:

$$RL_{dis}(O_i, O_{i+1}) = 1 - \max_b\{|\chi_{O_i}(b) + \chi_{O_{i+1}}(b) - 1|\} \tag{10.3}$$

$$RL_{in}(O_i, O_{i+1}) = \min(1, \min_b(1 + \chi_{O_{i+1}^\circ}(b) - \chi_{O_i}(b))) \tag{10.4}$$

$$\begin{aligned} RL_{con}(O_i, O_{i+1}) = \\ &\min\{(1 - \max_b(|\chi_{O_i^\circ}(b) + \chi_{O_{i+1}^\circ}(b) - 1|)), \\ &\max_b(\min(\chi_{\partial O_i}(b), \chi_{\partial O_{i+1}}(b)))\} \end{aligned} \tag{10.5}$$

$$\begin{aligned} RL_{ov}(O_i, O_{i+1}) = \\ &\min\{\max_b(\min(\chi_{O_{i+1}^\circ}(b), \chi_{O_i^\circ}(b))), \\ &\max_b(\min(\chi_{O_i^\circ}(b), \chi_{\partial O_{i+1}}(b))), \\ &\max_b(\min(\chi_{O_{i+1}^\circ}(b), \chi_{\partial O_i}(b))), \\ &\max_b(\min(\chi_{\partial O_i}(b), \chi_{\partial O_{i+1}}(b)))\} \end{aligned} \tag{10.6}$$

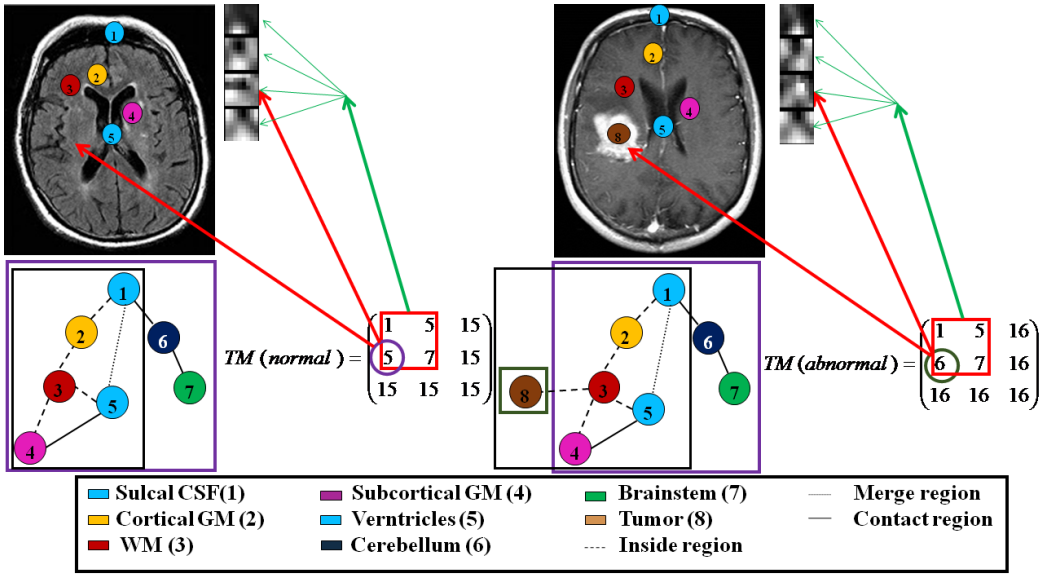


Figure 10.1: Example of normal and abnormal brain MRI images with their topological graph and basis vectors of the topological feature. The element $m_{21} = 5$ in the normal case while $m_{21} = 6$ in the abnormal case, this indicates that there is a tumor in the brain. The violet rectangle represents the overall topological graph of the normal case, the black rectangle represents the topological graph of the example images, and the green rectangle represents the abnormal connectivity of WM.

where b is a pixel in I. Table 10.1 and Figure 10.1 illustrate the proposed topological properties for both normal and abnormal cases of the brain. In Table 10.1, the connected components represent the total relationship of each region. The number of cavities in each region indicates the number of regions inside it. In Figure 10.1, the label (8) represents the abnormal connectivity of the white matter (presence of a tumor). Therefore the topological relationship of the white matter is changed. According to Eq. (10.1) and Eq. (10.2) this change is illustrated in element m_{21} in TM because the tumor is inside WM, for more details see Chapter 9 (Al-Shaikhli *et al.*, 2014c). This could be also seen in the basis vector of the topological feature of normal and abnormal brain MRI images.

Gray Level Co-occurance Matrix (GLM) GLM is an important method for textural feature extraction proposed by Haralick *et al.* (1973). Four texture features (contrast, correlation, energy, and inverse difference moment) are considered for brain tumor classification. These features have been calculated for four different offsets $(0°,\ 45°,\ 90°,$ and $135°)$.

10.2.2 Dictionary Learning

In this subsection, the dictionary learning step in our algorithm using a K-SVD method will be presented to learn and update the dictionary. Let $c = 1, ..., 4$ is the number of the class, N_c are the training images of each class. D_c are the dictionaries of the corresponding training

Table 10.1: Topological properties for the normal (abnormal) cases.

Tissue label	Tissue type	#connected	Internal cavity	Handles
1	Sulcal CSF	3 (3)	1 (1)	1 (1)
2	Cortical gray matter (GM)	2 (>2)	1 (>1)	1 (>1)
3	White matter (WM)	3 (>3)	2 (>2)	2 (>2)
4	Subcortical gray matter	2 (>2)	0 (>0)	0 (>0)
5	Ventricles	3 (>3)	0 (0)	0 (0)
6	Cerebellum	2 (>2)	0 (>0)	0 (>0)
7	Brain stem	1 (>1)	0 (>0)	0 (>0)

images of each class, and N is the sum of the training images of all four classes as explained in Figure 10.2 which illustrates the proposed algorithm.

Let D_c be a dictionary $n \times K_c$ matrix $D_c = (d_1, d_2, ..., d_{K_c})$, which consists of K_c atoms (columns), $\{d_i \in R^n : i = 1, 2, ..., K_c\}$ and each atom represents the key features extracted from Y_c, where $(K_c \ll N_c)\, Y_c = (y_1, y_2, ..., y_{N_c})$ is a $n \times N_c$ matrix which consists of feature vectors $\{y_i \in R^n : i = 1, 2, ..., N_c\}$ of N_c data samples (feature vectors) with dimension n. To compute the sparse representation $A_c = (a_1, a_2, ..., a_{N_c}) \in R^{K_c \times N_c}$, s.t. $y_i = D_c a_i$ and $\|a_i\|_0 << K_c, i = 1, ..., N_c$, the dictionary D_c by feature samples Y_c needs to be trained. In such a way that each feature vector in Y_c is represented by linear combination of a few atoms in the dictionary according to the non-zero elements in A_c as illustrated in the generative learning step in Figure 10.2.

Our goal is to update the dictionary and the sparse representation A_c by minimizing the following equation using the K-SVD method (Aharon *et al.*, 2006):

$$\begin{aligned} &\arg \min_{D_c, A_c} \|Y_c - D_c A_c\|_F^2 \ , \\ &s.t. \ \ \forall 1 \leq i \leq N_c, \ \|a_i\|_0 \ll K_c \end{aligned} \tag{10.7}$$

To get an update of the dictionary D_c and the sparse representation A_c, we assume that the condition in Eq. (10.7) is a $K_c \times N_c$ matrix multiplied by A_c as a dot product of multiplication:

$$P_c = \begin{cases} P_c(i,j) = 1 & \text{for} \quad A_c(i,j) = 0 \\ P_c(i,j) = 0 & \quad otherwise \end{cases} \tag{10.8}$$

Now, we can rewrite Eq. (10.7) as follows:

$$\{\hat{D}_c, \hat{A}_c\} = \arg \min_{D_c, A_c} \|Y_c - D_c A_c\|_F^2 \ \ s.t. \ \ P_c \circ A_c = 0 \tag{10.9}$$

The dot product $(P_c \circ A_c = 0)$ achieves all zeros in A_c without change. Equation (10.9) represents the update stage of the dictionary and we solve it by considering $D_c A_c$ as a sum of

rank-1 outer products:

$$\{\hat{D}_c, \hat{A}_c\} = \arg\min_{D_c, A_c} \|Y_c - \sum_{i=1}^{K} d_i a_i^T\|_F^2$$
$$s.t. \quad \forall 1 \leq i \leq K_c, \;\; p_i \circ a_i = 0 \tag{10.10}$$

To optimize the above equation, we use a block coordinate descent method. By multiplying the $(n \times N_c)$ rank-1 matrix $(1_n \cdot p_j^T)$ with Eq. (10.10), we compute the error matrices. Therefore, all columns of the samples that do not use j^th atom are removed.

$$E_i = (Y - \sum_{i \neq j} d_i a_i^T) \circ (1_n \cdot p_j^T) \tag{10.11}$$

where E_i are the overall representation error matrix. In Eq. (10.11), the rank-1 matrix represents the n times replication of the row p_j^T which forces the zeros in the right location in a_i. For each category c we have a learned dictionary D_c that contains atoms and each atom represents the key features of the samples in each category $\{Y_c \in R^{n \times N_c} : c = 1, \ldots, 4\}$ and the total number of feature samples is represented by $(Y = (Y_1, Y_2, \ldots, Y_4))$ in the dictionary:

$$D = (D_1, D_2, \ldots, D_4) \in R^{n \times N}, \qquad N = \sum_c N_c \tag{10.12}$$

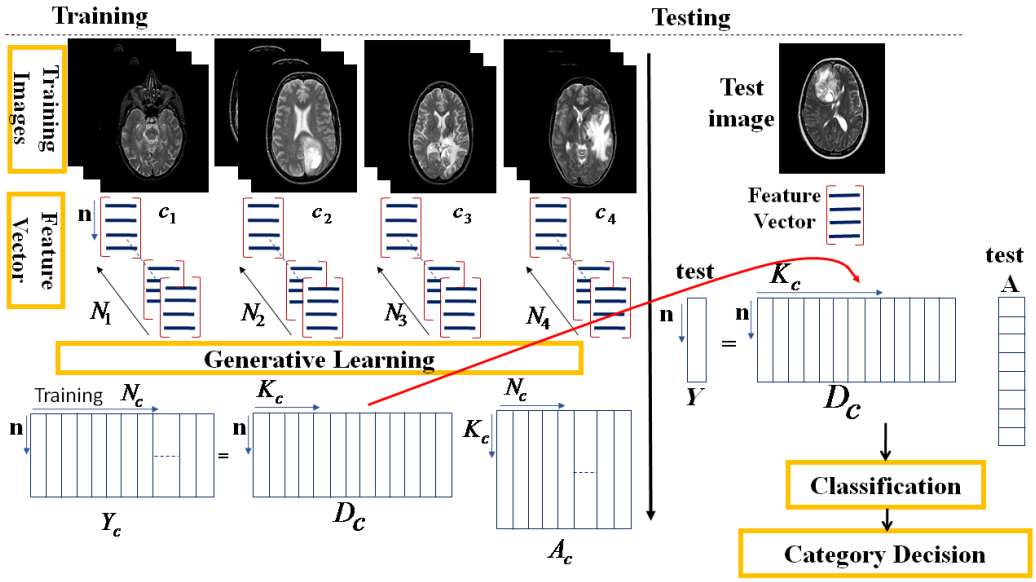


Figure 10.2: Schematic illustration of the proposed algorithm. In the training part, (c_1) normal, (c_2) glioma, (c_3) glioplastoma, (c_4) carcinoma and the feature vector represents the topological and texture features of each case. Generative learning illustrates the dictionary learning step. The testing part illustrates the computing of the sparse representation of the test images.

10.2.3 Classification

The classification step of the proposed algorithm is based on sparse representation. In Section 10.2.2, the dictionary learning step of each category was explained. To classify the testing data, the algorithm tries to find a match between the testing data Y and the dictionary of specific category D_c. This can be achieved by computing the similarity of the testing data with contents (key features) of the dictionary D_c. Therefore, the sparse representation of the testing data is computed using the individual dictionaries of the all categories (as illustrated in the testing part of Figure 10.2) and then Y is classified as a c^{th} category when appear that Y is more sparse with $D_{c^{th}}$:

$$\|Y - D_c A_c\|_F^2 \leq \epsilon \ , \ \ \|A_c\|_0 = \min\{\|A_b\|_0 : b = 1, \ldots, 4\} \tag{10.13}$$

10.3 Experimental Results and Discussion

To explore the advantages of the proposed algorithm compared to the other methods, several experiments have been conducted on diverse medical images. In this work three medical datasets are used, namely, brain web for simulated brain database (Cocosco *et al.*, 1997), brain tumor segmentation database (Kaus *et al.*, 2001; Warfield *et al.*, 2000), and whole brain atlas (Johnson & Becker, 1995), and other medical images with brain tumor from the internet. From all databases, 4 classes of images have been collected; 50 normal brain cases (class 1), 50 cases with brain glioma (class 2), 50 cases with brain glioplastoma (class 3), and 50 cases with brain metastatic carcinoma (class 4). Each case has a set of 10 images which make the total number of images for the training set of 4 classes 2000 images (50 cases $\times$ 4 classes $\times$ 10 images of each class = 2000 images). For testing, a 10-fold cross validation is used to evaluate the performance of the classification. Figure (2) shows examples of images from these databases.

Each patch in the dictionary is represented using a feature vector, including topological and textural information. The images are classified as normal case or abnormal case according to their topological properties as explained in Table 10.1. Then the images are classified furthermore according to the texture features of the abnormality if they exist. In the classification step, two types of classifiers are used (sparse coding classifier and Linear-SVM classifier). Sparse coding classifier performs higher classification accuracy than Linear-SVM classifier (93.7 % versus 88.75 %). Furthermore, the proposed algorithm (using sparse coding classifier) is compared with other classification methods (Han *et al.*, 2011; Weiss *et al.*, 2013) after adapting these methods for multi-class classification. In the proposed algorithm, the classification step is obtained by finding the match between the sparse representation of the testing data with the specific dictionary. The performance for multi-class classification (Recall, Precision, Average Accuracy (AA)) are computed by computing the True Positive (TP), True Negative (TN), False Positive (FP), False Negative (FN) using the algorithm (Sokolova & Lapalme, 2009):

$$Precision = \frac{\sum_{c=1}^{C} \frac{TP_c}{TP_c+FN_c}}{C}, Recall = \frac{\sum_{c=1}^{C} \frac{FP_c}{TP_c+FN_c}}{C}, AA = \frac{\sum_{c=1}^{C} \frac{TP_c+TN_c}{TP_c+TN_c+FP_c+FN_c}}{C}$$

Table 10.2 illustrates the classification performance of the proposed algorithm using sparse coding classifier better than other classification methods proposed in the literature (Han *et al.*,

	c1	c2	c3	c4
c1	0.99	0	0.01	0
c2	0	0.9565	0.0435	0
c3	0	0.0565	0.9435	0
c4	0	0.08	0.06	0.86

Figure 10.3: Confusion matrix for all datasets. The average accuracy is 93.75%. Most confusions occur in brain carcinoma. (c_1) Normal, (c_2) Glioma, (c_3) Glioplastoma, (c_4) carcinoma.

Table 10.2: Classification Evaluation.

Classifier Type	Recall	Precision	AA
Sparse Coding (Proposed)	**92.5**%	**94.87**%	**93.75**%
Han *et al.* (2011)	**92.5**%	90.24%	91.25%
Weiss *et al.* (2013)	90.0%	92.31%	90.0%

2011; Weiss *et al.*, 2013). From the confusion matrix in Figure 10.3, it can be observed that the class 1 (normal) is classified correctly with minimum because the topological feature gives accurate information of the normal and abnormal cases. The errors occurred mainly with the class 4 (carcinoma) with error 0.14 which is classified as class 2 (glioma) and class 3 (glioplastoma) due the textural similarity in T2 MRI images of these cases. Class 3 (glioplastoma) is classified as class 2 (glioma) with error 0.0565 and class 2 (glioma) is classified as class 3 (glioplastoma) with error 0.0435. Totally, 6.25 % of the four classes are classified incorrectly.

10.4 Conclusion

In this work dictionary learning and sparse coding are proposed for multi-class brain tumor classification. The dictionary is constructed and learned from the topological and texture features of the trained data. Then the learned dictionary is used to classify the testing data. Two types of classifiers are used for classification namely sparse coding and linear-SVM. The sparse coding classifier computes the matching between the sparse representation of the testing data and the corresponding dictionary. The results showed that the sparse representation based classification achieves higher classification accuracy than Linear-SVM based classification technique (93.75 % versus 88.75 %). The proposed algorithm has also been compared to other classification methods, demonstrating the advantages of the method proposed in this work.

Chapter 11

Coupled Dictionary Learning for Automatic Multi-Label Brain Tumor Segmentation in Flair MRI images

Brain tumor tissue segmentation and labeling is a challenging task in medical imaging. In this chapter, a novel patch based dictionary learning algorithm for automatic multi-label brain tumor segmentation is proposed. Based on image reconstruction, we present coupled dictionaries, one dictionary of grayscale brain tumor image patches and one dictionary of tumor labels, which can then be used for automatic multi-label brain tumor segmentation of a test image data. The dictionaries are learned from training images of BraTS-MICCAI and the SPL/NSG brain tumor databases. The label dictionary is proposed to select foreground and background labels for automatic graph-cut segmentation. For quantitative evaluation, five different metric scores are computed using the online evaluation tool provided by the BraTS organizers. Experimental results demonstrate that the proposed approach achieves accurate results and outperforms most of the state-of-the-art methods cited in BraTS-MICCAI challenge. An earlier version of this chapter appeared at the International Symposium on Visual Computing (ISVC) (Al-Shaikhli *et al.*, 2014b).

11.1 Introduction

Early identification and accurate boundary detection of brain tumor are important for effective diagnosis and treatment. Multi-label brain tumor segmentation is considered as one of the most challenging tasks in medical imaging due to the difficulty to extract the relevant information that can help to discriminate the tumor (Sachdeva *et al.*, 2013). Several obstacles like inter-patient heterogeneity and geometric variation in shape and size lead to difficulties of tumor detection using shape or intensity prior for brain tumor segmentation (Jiang *et al.*, 2013).

Many previous approaches use neural networks, interactive tools or morphology (Moon *et al.*, 2002) or atlas (Gooya *et al.*, 2012) for brain tumor segmentation. For multi-label brain tumor segmentation, several researchers used either random forest based on feature extrac-

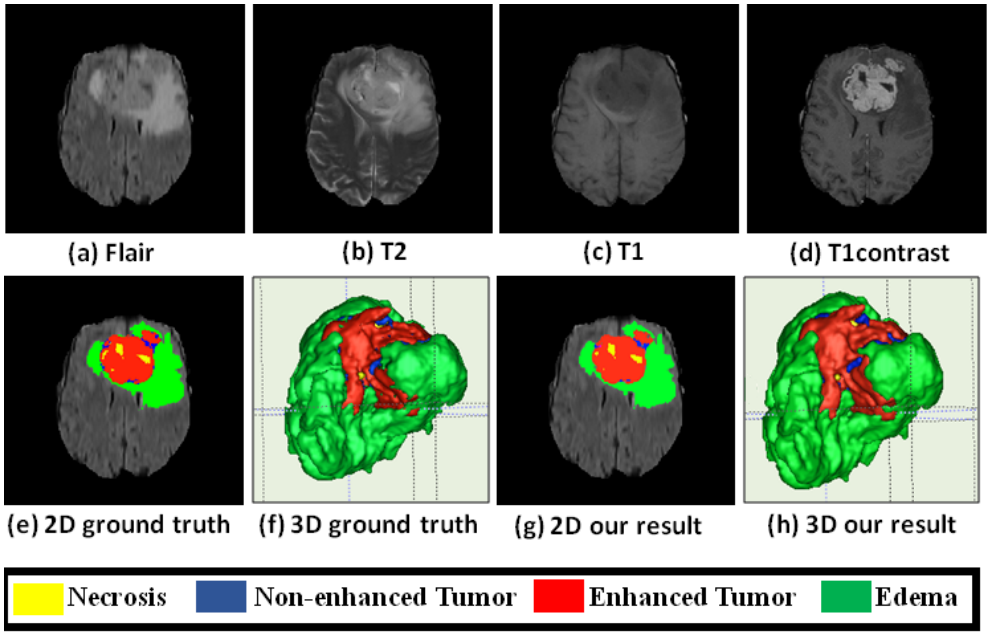


Figure 11.1: Example of the BraTS training data (Menze *et al.*, 2014). (a-d) are four brain MRI modalities with glioma, (e, f) 2D and 3D ground truth respectively, (g, h) 2D and 3D segmentation of our algorithm respectively using only Flair MRI modality. Each image modality gives specific information of the glioma sub-regions. Edema clearly appear in T2 and Flair modalities (a, b), necrosis appears in T2 (with high intensity) and in T1 (with low intensity) (b, c) while enhanced tumor clearly appears in T1-contrast modality (d).

tion (Festa *et al.*, 2013; Reza & Iftekharuddin, 2013; Tustison *et al.*, 2013) or Markov Random Field (MRF) (Zhao *et al.*, 2013a,b). Some other approaches used atlas template for single label (Warfield *et al.*, 2000) or multi-label brain tumor segmentation (Cao *et al.*, 2013; Gooya *et al.*, 2011, 2012). All the above methods require multi-channel input (multi-modality MRI data). In recent years, some approaches used image patch dictionary learning for single-label tumor segmentation (Thiagarajan *et al.*, 2013; Weiss *et al.*, 2013) or multi-modal coupled dictionary learning for microscopical image registration (Cao *et al.*, 2013). These methods used one dictionary for each class and the residual error to discriminate the tumor/non-tumor classes. Cordier *et al.* (2013) proposed a patch based brain tumor segmentation. The method is based on the patch similarity to detect the tumor region in different image modalities (Flair, T2, T1, and T1-contrast). The method requires an initial bounding box localization to detect the tumor region and four dictionaries for the four image modalities. Brain glioma of BraTS database (Menze *et al.*, 2014) is represented by four labels (necrosis, enhanced tumor, non-enhanced tumor, and surrounding edema). The appearance of these labels differs in each MRI image modality (T1, T2, T1-contrast, and Flair). In T2, edema pixels tend to have higher intensity rather than in T1. Necrosis pixels tend to have high intensity in T2 while low intensity in T1. Enhanced tumor pixels tend to have high intensity in T1 while low intensity in T2 as illustrated in Figure 11.1. This makes it difficult to identify the multi-label glioma from just one

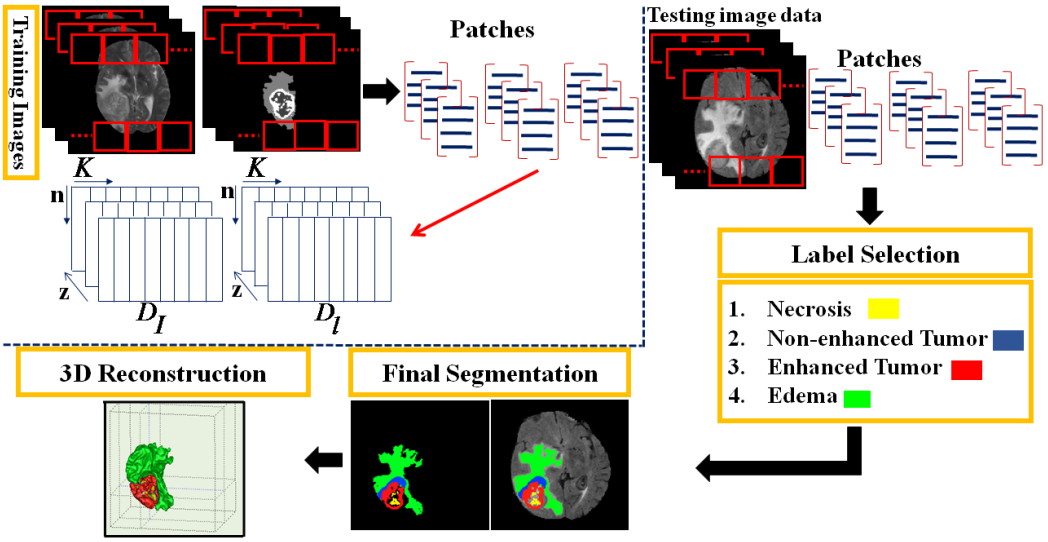


Figure 11.2: Schematic illustration of our algorithm. The training part represents the training images with associated label images and their dictionaries. The testing part represents selection of foreground and background labels, 2D tumor segmentation, and 3D tumor reconstruction of the test image (patient # 308).

MRI modality and therefore, several multi-label glioma segmentation methods require more than one modality as explained in aforementioned approaches. Figure 11.1 shows an example of brain glioma of BraTS training database (Menze *et al.*, 2014). In Figure 11.1, (a-d) show the appearance of glioma and the surrounding edema in four MRI modalities, (e, f) show the 2D and 3D ground truth segmentation respectively, and (g, h) show the 2D and 3D segmentation of the proposed algorithm using only one modality (Flair MRI modality).

In contrast to aforementioned methods and in order to solve the limitation of using multi-channel input, which may result in more computational complexity (time and memory), in this work, we propose a novel coupled dictionary learning approach (one dictionary of the original image data and one of the associated label image data) of automatic multi-label brain tumor segmentation, as illustrated in Figure 11.2. Our **contribution** is a novel fully automatic algorithm for multi-label segmentation using coupled dictionaries learned from single modality (Flair MRI modality) image training data with associated label image data (ground truth segmentation). Patches are extracted from the training image data and concatenated to a matrix in a dictionary. Each patch has its corresponding patch in a label dictionary. The label dictionary represents four foreground labels (necrosis, enhanced tumor, non-enhanced tumor, and edema) and one background label. For testing, the proposed method requires single MRI modality input of the testing data. After extracting the patches from the test image data, the patch similarity is retrieved between the patches of the testing data and these in the dictionary of the training image data, then the corresponding atoms in the label dictionary are selected. The label dictionary is used to provide the foreground and background labels for graph-cut segmentation.

The following sections are organized as follows. Our novel approach is described in detail

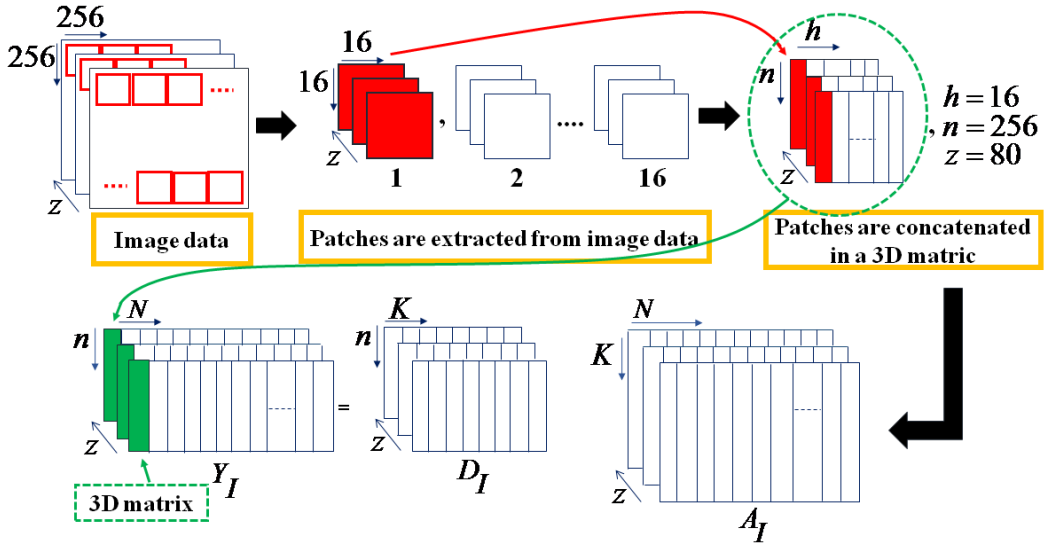


Figure 11.3: Schematic illustration of the matrix dimension. Patches are extracted from each image data and concatenated in a 3D matrix. The 3D matrices of training set are concatenated in a big 3D matrix Y_I to learn D_I and A_I.

in Section 11.2. In Section 11.3, experimental results and discussion are presented. Finally, this work is concluded in Section 11.4.

11.2 Method

The proposed method consists of two steps: image patch dictionary learning step and segmentation step.

11.2.1 Dictionary Learning

In this subsection, we explain the dictionary learning step of both the original image data and tumor label image data. The size of all images is set to $256 \times 256 \times 80$. Sixteen patches are extracted from each image data with size $16 \times 16 \times 80$ and these patches are concatenated in a 3D matrix as illustrated in Figure 11.3.

Let D_I be a dictionary $n \times K \times z$ matrix $D_I = (d_1, d_2, ..., d_K)$, which consists of K atoms (columns), $\{d_i \in R^{n \times 16 \times z} : i = 1, ..., K\}$. Therefore, each atom represents grayscale image patches concatenated to a matrix with size $n \times 16 \times z$ and z is the depth of the training image volume Y_I. Y_I is a $n \times N \times z$ matrix $Y_I = (y_1, y_2, ..., y_N)$, which consists of sample matrices $\{y_i \in R^{n \times 16 \times z} : i = 1, ..., N\}$ of N data samples and $(K \ll N)$, as illustrated in Figure 11.3. To compute the sparse representation $A_I = (a_1, a_2, ..., a_N) \in R^{K \times N \times z}$, s.t. $y_i = D_I a_i$ and $\|a_i\|_0 << K, i = 1, ..., N$, where D_I needs to be trained by Y_I. In such a way that each sample in the training image data is represented by a linear combination of a few atoms in the corresponding dictionary according to the non-zero elements in A_I.

The dictionary D_I is coupled with a corresponding dictionary matrix of tumor label $D_l \in R^{n \times K \times z}$. D_l is built and learned from the ground truth segmentation of the training data. D_l atoms have the same spatial extension as D_I as explained in Figure 11.3, so labels in D_l can be inferred on each pixel of the original image data. Each atom d_I in D_I has an associated atom d_l in D_l. To achieve this coupling, the dictionary learning procedure is done simultaneously for both D_I and D_l to approximate the solution of error matrix in Eq. (11.5) below. Below, we will explain the dictionary learning procedure of D_I and this procedure is also done for D_l. The dictionary D_I and the sparse representation A_I are learned by minimizing the following equation using the K-SVD method (Aharon *et al.*, 2006):

$$\arg \min_{D_I, A_I} \|Y_I - D_I A_I\|_F^2 \quad s.t. \quad \forall 1 \leq i \leq N, \ \|a_i\|_0 \ll K \tag{11.1}$$

To minimize Eq. (11.1), we assume that it is constrained by a dot product, $(P \cdot A_I = 0)$, of a $K \times N \times z$ matrix (A_I) with a $K \times N \times z$ matrix (P) defined below:

$$P = \begin{cases} P(i,j) = 1 & \text{for} \quad A_I(i,j) = 0 \\ P(i,j) = 0 & otherwise \end{cases} \tag{11.2}$$

Now, we can rewrite Eq. (11.1) as follows:

$$\{\hat{D_I}, \hat{A_I}\} = \arg \min_{D_I, A_I} \|Y_I - D_I A_I\|_F^2 \quad s.t. \quad P \cdot A_I = 0 \tag{11.3}$$

The dot product $(P \cdot A_I = 0)$ achieves all zeros in A_I remain unchanged. Equation (11.3) represents the update stage of the dictionary and we solve it by considering $D_I A_I$ as a sum of rank-1 outer products:

$$\{\hat{D_I}, \hat{A_I}\} = \arg \min_{D_I, A_I} \|Y_I - \sum_{i=1}^{K} d_i a_i^T\|_F^2 \quad s.t. \quad \forall 1 \leq i \leq K, \ p_i \cdot a_i = 0 \tag{11.4}$$

where d_i is the i^{th} atoms in D_I and a_i^T is the i^{th} row in A_I. To optimize the above equation, we use an SVD operation. The error matrices are computed by multiplying the $(n \times N \times z)$ rank-1 matrix $(1_n \cdot p_i^T)$ with $(Y_I - \sum_{i \neq j} d_j a_j^T)$:

$$E_i = (Y_I - \sum_{i \neq j} d_j a_j^T) \cdot (1_n \cdot p_i^T) \tag{11.5}$$

where E_i are the overall representation error matrices, p_i^T is the i^{th} row in the matrix P. In Eq. (11.5), the rank-1 matrix represents the n times replication of the row p_i^T. Therefore, all columns of the samples that do not use i^{th} atom are removed.

The tumor in the label image data is represented by four labels l_i (clusters) with a gray level range $\{0, 0.25, 0.5, 0.75, 1\}$ (necrosis=1, edema=0.75, non-enhanced tumor=0.5, enhanced tumor=0.25, and background=0). Let c be a set of voxels in the label atom. In the perfect representation of tumor labels in label dictionary D_l, each atom in D_l represents by voxels which

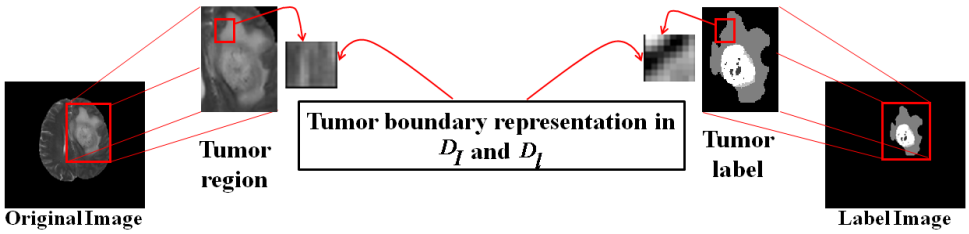


Figure 11.4: Example of brain tumor image with associated tumor label image with their dictionary representation.

have maximum probability for one label and minimum probability for other three labels:

$$\hat{D}_l(c) = \begin{cases} 1 & \text{if } l_1 = max_{l_1}\ d_{l_1}(c) \\ 0.75 & \text{if } l_2 = max_{l_2}\ d_{l_2}(c) \\ 0.5 & \text{if } l_3 = max_{l_3}\ d_{l_3}(c) \\ 0.25 & \text{if } l_4 = max_{l_4}\ d_{l_4}(c) \\ 0 & \text{if } otherwise \end{cases} \tag{11.6}$$

To optimize this requirement of containing one label information in each voxel, the label dictionary D_l is computed by minimizing the following equation:

$$\arg\min_{D_l, A_l} \|\hat{D}_l - D_l A_l\|_F^2 \quad s.t. \quad \forall 1 \leq i \leq N,\ \|a_i\|_0 \ll K \tag{11.7}$$

For all atoms in D_I and D_l, if $\min_{i=1}^{K}(1 - \sum_{l=1}^{4} \|y_{il} - \hat{D}_l\|) \geq 0$, then the atom d_{I_i} is associated with the atom d_{l_i}. Figure 11.4 illustrates the representation of the original image in the dictionary and its corresponding representation in the label dictionary. The dictionary D_I (Eq. 11.1) and D_l (Eq. 11.7) and their sparse representation A_I and A_l are learned using the same learning procedure explained in Section 11.2.1 (Eqs. 11.1 - 11.5).

11.2.2 Label Selection for Graph-Cut Segmentation

To segment the tumor region with surrounding edema, four labels for foreground ad one for background are used as explained in Section 11.2.1. Firstly the patches are extracted from the testing image data (Flair MRI modality) and the maximum match of these patched with dictionary D_I is found. Then the corresponding label patches from the label dictionary D_l are selected. This can be achieved by computing the similarity of the testing image data with the contents of the dictionary D_I. Then the labels are selected from the corresponding atom in the label dictionary D_l:

$$\|Y_I - D_I A_I\|_F^2 \leq \epsilon\ ,\ \min_A \|A\|_0 \tag{11.8}$$

where ϵ is very small value[1]. In this step, only the patches of the tumor (i.e. the patches of the tumor region of original image data and label patches of the ground truth segmentation) are

[1] In this work, ϵ is set to 10^{-4}.

selected for graph-cut segmentation.

For segmentation, we use multi-label graph-cut toolbox (Delong *et al.*, 2012). The segmented slices of the test image data are rendered for the 3D reconstruction of the brain tumor. The energy function consists of three terms: data cost, smooth cost, and label cost:

$$E(f) = \underbrace{\sum_{c \in C} \varphi_c(f_c)}_{data\ cost} + \underbrace{\sum_{cq \in G} V_{cq}(f_c, f_q)}_{smooth\ cost} + \underbrace{\sum_{l \subseteq L} h_l \cdot \delta_l(f)}_{label\ cost} \tag{11.9}$$

where $E(f)$ is the energy to be minimized, C is a set of all voxels in the image data and for each $c \in C$ there is a label $f_c \in L$, ($f_c = l \in \{1, 2, 3, 4\}$ as explained in Eq. 11.6). G is a set of all edges between voxels. φ_p is the data of the label c. Each V_{cq} represents the edge between two voxels c and q and penalizes $f_c \neq f_q$. The label cost term consists of the (1) non-negative label subset cost of each label h_l and it is represented by the patches in the dictionary of the original image data D_I, (2) the indicator function δ_l which is represented by label patches in D_l (Eq. 11.6).

11.3 Experimental Results and Discussion

To explore the advantages of the proposed algorithm compared to the other methods, several experiments have been conducted on diverse medical images. In this work two medical datasets are used, namely, brain tumor (glioma) database (SPL/NSG) (Johnson & Becker, 1995; Warfield *et al.*, 2000) (10-patients image data) and brain tumor segmentation database (BraTS) (Menze *et al.*, 2014) (Images of 20 patients as training and images of 10 patients as testing image data with different MRI modalities (T1, T2, T1-contrast, and Flair)). The BraTS database is publicly available through the MICCAI 2013 Brain Tumor Segmentation challenge (Menze *et al.*, 2014), where these databases provide the ground truth of the brain tumor segmentation (manual expert annotations). In this work, we use only the Flair MRI modality to build and train the dictionary of the original image and the ground truth segmentation as tumor label image data to build the label dictionary. The size of all images is set to $256 \times 256 \times 80$. All experiments are conducted in MATLAB using a 2.0 GHz RAM.

Sixteen patches are extracted from the original image data and image label with size $16 \times 16 \times 80$ and each patch in the dictionary is concatenated in a 3D matrix $256 \times 16 \times 80$ as illustrated in Figure 11.3. Each patch in the dictionary of the original data has its corresponding patch in the label dictionary with same size. The selection of labels is based on finding a match between the target image data and the dictionary D_I as explained in Section 11.2.2. Then the labels are selected from the corresponding atom in the label dictionary D_l. Figure 11.5 shows one-label glioma segmentation of two examples of the SPL/NSG database (Johnson & Becker, 1995; Warfield *et al.*, 2000). In each example, the first column is the original image data (coronal, sagittal, and axial sections). The second column is the 2D tumor segmentation. The 3D tumor reconstruction is represented in the last row. Figures 11.6 and 11.7 show the results of the multi-label brain tumor segmentation of the testing data and the training data of BraTS database (Menze *et al.*, 2014) respectively. In Figure 11.6, the first, third, and fifth

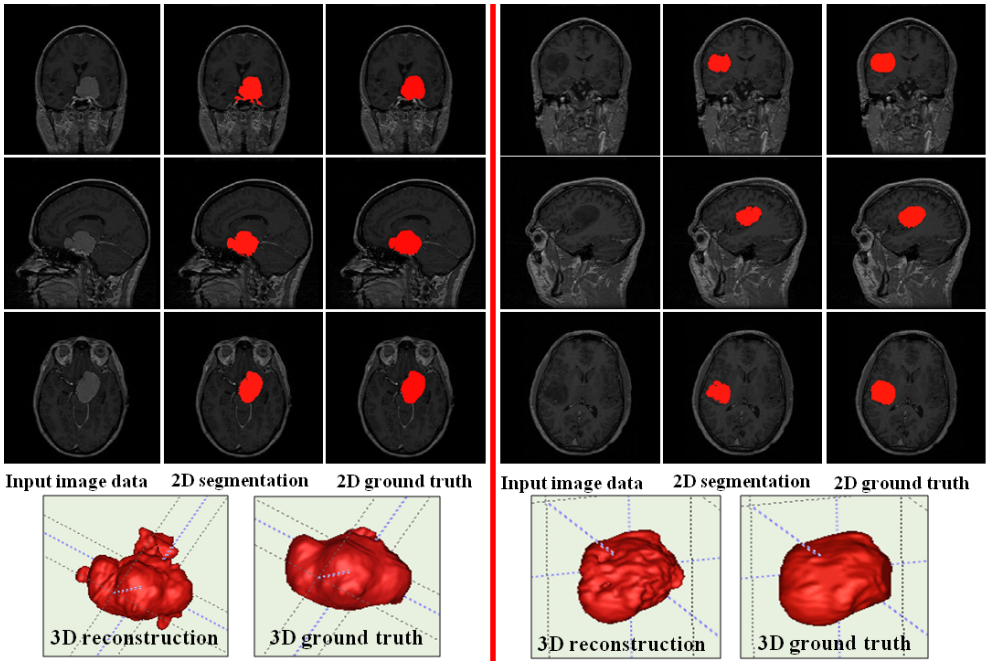


Figure 11.5: Two examples of one label glioma segmentation of SPL/NSG database (Johnson & Becker, 1995; Warfield *et al.*, 2000), in each example, the first column is the original image data. The second column is our tumor segmentation. The third column is ground truth. The last row is the 3D tumor reconstruction with 3D ground truth.

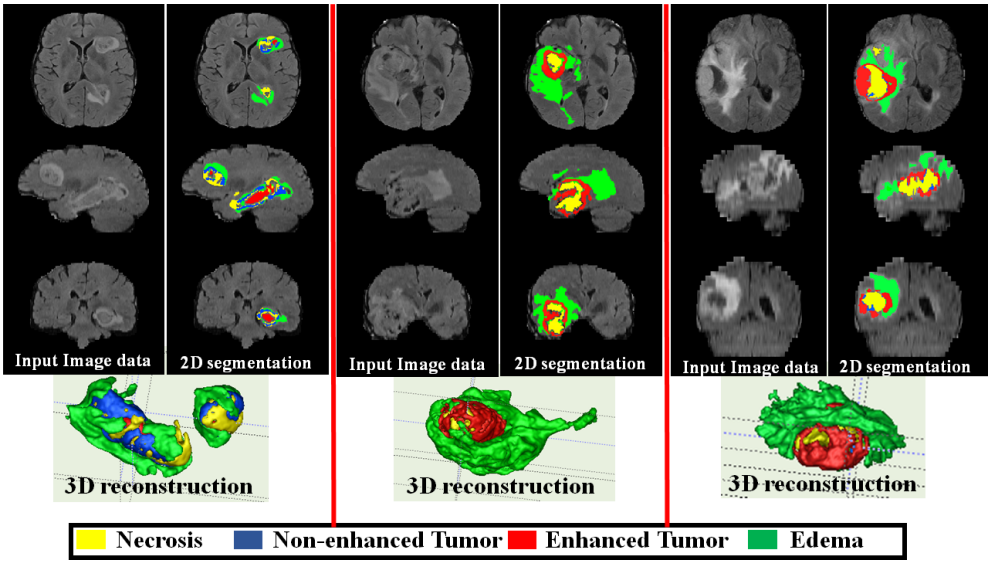


Figure 11.6: Three examples (left → to right: patients # 310, 309, 302) of 3D Multi-label glioma segmentation of BraTS testing data, in each example, the first column is the input image data (axial, coronal, and sagittal planes), the second column is the 2D tumor segmentation, the fourth row is 3D tumor reconstruction.

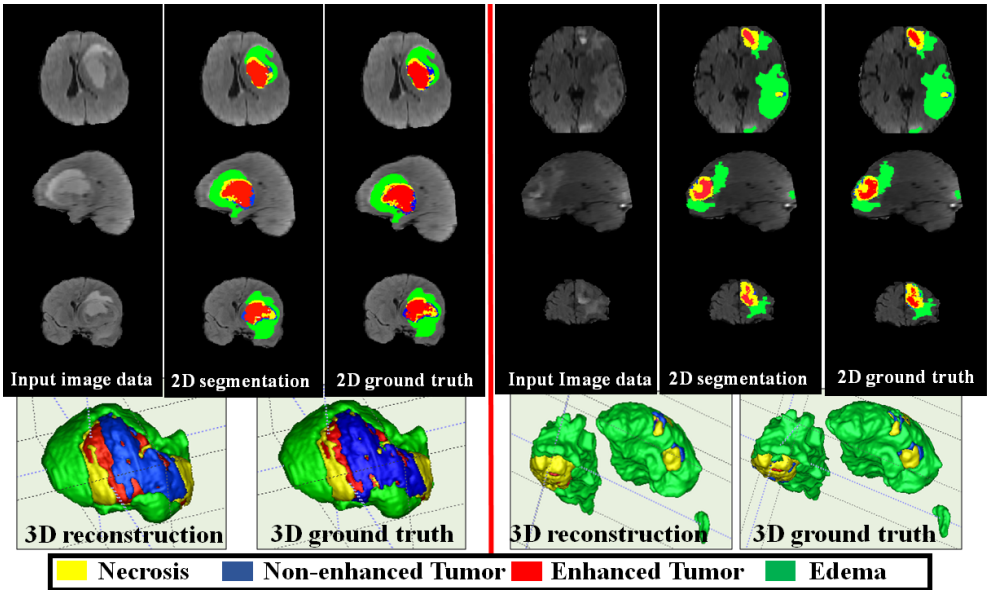


Figure 11.7: Two examples of 3D Multi-label glioma segmentation of BraTS training data, in each example, the first column is the input image data (axial, coronal, and sagittal planes), the second column is our results (2D), the third column is the 2D ground truth, the fourth row is 3D tumor reconstruction (our results) and 3D ground truth respectively.

columns are the input image data in three different planes (coronal, sagittal, and axial). The second, fourth, and sixth columns are the segmentation results of the proposed algorithm and the fourth row represents the 3D tumor reconstruction. Figure 11.7 shows our segmentation results comparing with the ground truth. The best segmentation are achieved for the edema and necrosis more than the enhanced tumor and the thin layer of non-enhanced tumor.

The performance of the our algorithm was evaluated with the four BraTS 2013 labels: label 1: necrosis, label 2: edema, label 3: non-Enhancing tumor, and label 4: enhancing Tumor. The results were uploaded to the BraTS 2013 Virtual Skeleton web site and the evaluation of the testing image data is obtained for 3 different tumor sub-regions according to BraTS challenge conditions:

- Region 1 (R1): Complete tumor (labels 1+2+3+4 for patient image data)
- Region 2 (R2): Tumor core (labels 1+3+4 for patient image data)
- Region 3 (R3): Enhancing tumor (label 4 for patient image data)

Table 11.1 shows the results per region reported by the Virtual Skeleton web site for the 10 real-case high grade BraTS 2013 testing image data (Menze *et al.*, 2014). Tabel 11.1 illustrates five different metrics (Dice, Positive Predictive Value (PPV), Sensitivity, Jaccard, and Kappa) with the average value (Avg) and standard deviation (Std). Patient #305 is the best segmentation result. We compare our results with the best reported 3D brain tumor segmentation methods (Cordier *et al.*, 2013; Festa *et al.*, 2013; Meier *et al.*, 2013; Reza & Iftekharuddin, 2013;

Table 11.1: Evaluation Results of different tumor labels for 10 high grade real-patient of BRATS Testing Data.

P. #	Dice			PPV			Sensitivity			Jaccard			Kappa		
	R1	R2	R3	R1	R2	R3	R1	R2	R3	R1	R2	R3	R1	R2	R3
301	0.86	0.90	0.80	0.90	0.90	0.82	0.82	0.90	0.78	0.75	0.81	0.66	0.99	**1.0**	**1.0**
302	0.84	0.76	**0.87**	0.74	0.94	0.85	**0.97**	0.63	0.90	0.73	0.61	**0.78**	0.99	0.99	**1.0**
303	0.90	0.82	0.57	0.87	**0.997**	0.67	0.92	0.70	0.50	0.81	0.70	0.40	0.99	0.99	0.99
304	0.85	0.74	0.38	0.87	0.68	0.76	0.84	0.82	0.25	0.74	0.60	0.23	0.99	0.99	0.99
305	**0.91**	**0.91**	0.81	0.91	0.91	0.79	0.93	0.90	0.84	**0.85**	**0.83**	0.68	**1.0**	**1.0**	**1.0**
306	0.86	0.81	0.68	**0.96**	0.95	0.82	0.78	0.70	0.58	0.76	0.68	0.52	0.99	**1.0**	**1.0**
307	**0.91**	0.27	0.60	0.93	0.16	0.60	0.90	0.71	0.61	0.84	0.15	0.43	**1.0**	0.99	**1.0**
308	**0.91**	0.89	0.67	0.86	0.87	0.53	0.96	**0.91**	**0.91**	0.83	0.80	0.50	0.99	**1.0**	0.99
309	0.85	0.85	0.84	0.95	0.93	**0.87**	0.76	0.78	0.82	0.73	0.73	0.73	0.99	**1.0**	**1.0**
310	0.82	0.85	0.82	0.95	0.93	0.85	0.72	0.80	0.80	0.70	0.74	0.70	**1.0**	**1.0**	**1.0**
Avg.	0.87	0.78	0.70	0.89	0.83	0.75	0.86	0.78	0.70	0.77	0.66	0.56	0.99	1.0	1.0
Std	0.03	0.18	0.15	0.06	0.24	0.11	0.08	0.09	0.20	0.05	0.19	0.17	0.0	0.0	0.0

Tustison *et al.*, 2013; Zhao *et al.*, 2013a) of the BraTS 2013 challenge. Among these methods, the proposed method gives high score results as illustrated in Table 11.2. In Table 11.2 we report the same validation measures of the testing data (Dice, Positive Predictive Value, Sensitivity, Kappa, and Jaccard) as used in the challenge. Those results have also been published online [1]. On this challenge database, we get comparable results to best scored methods as illustrated in Table 11.2. From Table 11.2 we note that our algorithm gives the highest PPV score among the other methods. For dice and sensitivity coefficients, the proposed algorithm gives the highest score of R1 but not for R2 (tumor core) and R3 (enhanced tumor) because pixels in R2 and R3 tend to have low intensity in the Flair MRI modality but they still achieve the clinical requirements. Kappa coefficient indicates the agreement between the labels of the segmented tumor in the testing data with those in the ground truth. Our method also presents the advantage of being robust and requiring only one image modality input (single-channel input) comparing to other methods (Cordier *et al.*, 2013; Festa *et al.*, 2013; Meier *et al.*, 2013; Reza & Iftekharuddin, 2013; Tustison *et al.*, 2013; Zhao *et al.*, 2013a) which all require multi-channel input (four MRI modalities). Moreover, we evaluate the robustness of our algorithm using a k-fold cross validation of the training data (Johnson & Becker, 1995; Menze *et al.*, 2014; Warfield *et al.*, 2000), where k=30 (20 patient image data with different image modality T1, T2, T1-contrast, and Flair) for training image BraTS database (Menze *et al.*, 2014) and (10 patient image data MRI-T1) for SPL/NSG database (Johnson & Becker, 1995; Warfield *et al.*, 2000) as illustrated in Table 11.3.

[1] `http://martinos.org/qtim/miccai2013/results.html`

Table 11.2: Evaluation results for BraTS testing data.

Method	Dice			PPV			Sensitivity			Kappa
	R1	R2	R3	R1	R2	R3	R1	R2	R3	
Proposed	**0.87**	**0.78**	0.70	**0.89**	**0.83**	**0.75**	0.86	0.78	0.70	0.99
Tustison *et al.*	**0.87**	**0.78**	**0.74**	0.85	0.74	0.69	0.89	**0.88**	**0.83**	0.99
Meier *et al.*	0.82	0.73	0.69	0.76	0.78	0.71	**0.92**	0.72	0.73	0.99
Reza & Iftekharuddin	0.83	0.72	0.72	0.82	0.81	0.70	0.86	0.69	0.76	0.99
Zhao *et al.*	0.84	0.70	0.65	0.80	0.67	0.65	0.89	0.79	0.70	0.99
Cordier *et al.*	0.84	0.68	0.65	0.88	0.63	0.68	0.81	0.82	0.66	0.99
Festa *et al.*	0.72	0.66	0.67	0.77	0.77	0.70	0.72	0.60	0.70	0.98

Table 11.3: Evaluation for SPL/NSG (Johnson & Becker, 1995) and BraTS Training (Menze *et al.*, 2014) databases. R1 represents the average score of both of databases (Johnson & Becker, 1995; Menze *et al.*, 2014), R2 and R3 represent the average score of BraTS database (Menze *et al.*, 2014)

Method	Dice			PPV			Sensitivity			Kappa
	R1	R2	R3	R1	R2	R3	R1	R2	R3	
Proposed	0.96	0.94	0.80	0.96	0.95	0.78	0.90	0.90	0.74	0.99

11.4 Conclusion

In this work, a fully automatic multi-label brain tumor segmentation based on coupled dictionary learning and sparse coding algorithm is proposed. We design two types of associated dictionaries, one is the dictionary of the original image data and the other is the dictionary of tumor label image data (ground truth segmentation of the tumor). The evaluation of the results of the BraTS test image data has been obtained by the online evaluation tool provided by the BraTS-MICCAI challenge organizers. The experimental results show that the proposed algorithm achieves higher segmentation accuracy of R1 than R2 and R3. Because R2 and R3 tend to have low intensity in Flair MRI modality, therefore, the segmentation accuracy of R2 and R3 is slightly less than R1 but still achieve the clinical requirements.

Our approach requires only one modality comparing to the other methods which require multi-modalities which may result in less computational complexity (time and memory). The proposed algorithm has been compared to the best reported 3D brain tumor segmentation methods of the BraTS 2013 challenge, demonstrating the advantages of the method proposed in this work.

11. COUPLED DICTIONARY LEARNING FOR AUTOMATIC MULTI-LABEL BRAIN TUMOR SEGMENTATION IN FLAIR MRI IMAGES

Part III

REMOTE SENSING IMAGE CLASSIFICATION

Chapter 12

Combine Markov Random Fields and Marked Point Processes to Extract Building from Remotely Sensed Images

Automatic building extraction from remotely sensed images is a research topic much more significant than ever. One of the key issues is object and image representation. Markov random fields usually referring to the pixel level can not represent high-level knowledge well. On the contrary, marked point processes can not represent low-level information well even though they are a powerful model at object level. In this chapter, we propose to combine Markov random fields and marked point processes to represent both low-level information and high-level knowledge, and present a combined framework of modeling and estimation for building extraction from single remotely sensed image. At high level, rectangles are used to represent buildings, and a marked point process is constructed to represent the buildings on ground scene. Interactions between buildings are introduced into the the model to represent their relationships. At the low level, a MRF is used to represent the statistics of the image appearance. Histograms of colors are adopted to represent the building's appearance. The high-level model and the low-level model are combined by establishing correspondences between marked points and nodes of the MRF. We adopt reversible jump Markov Chain Monte Carlo (rjMCMC) techniques to explore the configuration space at the high level, and adopt a Graph Cut algorithm to optimize configuration at the low level. We propose a top-down schema to use results from high level to guide the optimization at low level, and propose a bottom-up schema to use results from low level to drive the sampling at high level. Experimental results demonstrate that better results can be achieved by adopting such hybrid representation. An earlier version of this chapter appeared at the ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, ISPRS Congress (Chai *et al.*, 2012).

12.1 Introduction

With the progresses in image capture techniques, more and more remotely sensed images with high spatial, spectral, temporal and radiometric resolutions are available. With the popularity of tools such as Google Map, more and more up-to-date geoinformation are demanded by people. As a traditional research topic in photogrammetry and remote sensing, building extraction from remotely sensed images is a topic much more significant than ever.

In spite of the research efforts of the past decades fully automatic extraction is still a challenging task. The key issue is representation of objects and images (Baltsavias, 2004; Mayer, 1999; Sowmya & Trinder, 2000). Statistical approaches provide a strong framework of modeling and estimation. Markov random fields and marked point processes represent context-dependent entities well (Baddeley & van Lieshout, 1993; Li, 2009; Winkler, 2003). Based on Markov random fields (MRF), low-level information referring to the single image pixels and interaction between neighboring pixels are represented concisely. However, high-level knowledge, such as free semantic structures and variable topology, can not be represented by MRFs conveniently. Based on spatial point process, high-level knowledge can be introduced via marks attached to the points and the relationships between neighboring points. While specific shapes can be represented by geometric marks, general shape can not be determined based on image content. This problem results from the weakness of representing low-level information.

Motivated by the complementary characteristics of Markov random fields and marked point processes, we combine them to represent both low-level information and high-level knowledge. Based on this representation, we propose an automatic approach for extracting buildings from single remotely sensed image.

12.2 Previous Works

12.2.1 Markov Random Fields based Representation

Markov random fields provide a natural representation of context-dependent entities (Besag, 1974; Geman & Geman, 1984). A set of sites are used to represent pixels or primitives, and a set of labels attached to each site are used to denote events that may happen at the site. Furthermore, a neighborhood system is used to describe the interrelationships between sites. Benefiting from the equivalence between MRF's and Gibbs distribution (Hammersley & Clifford, 1971), MRF can be described by local characteristics. Moreover, MRF-MAP estimation can be rigorously achieved in case of binary classification by Graph Cut algorithm (Boykov *et al.*, 2001) and approximately e.g. by belief propagation (Felzenszwalb & Huttenlocher, 2004a).

Although there are some high-level MRF models (Li, 1994), their structures are fixed in modeling and estimation. They are not flexible enough to represent random structures. Successes are demonstrated in low level, especially for regular lattices corresponding to image grids. For example, image restoration (Felzenszwalb & Huttenlocher, 2004a), stereo matching (Tappen & Freeman, 2003), image segmentation (Rother *et al.*, 2004) or clustering (Zabih & Kolmogorov, 2004) are formulated as pixel labeling problems with labels denoting pixel intensity, disparity or object category respectively.

Researchers introduced strong priori information into MRFs to improve its performance of representation. Kumar *et al.* (2010) combined pictorial structures and MRFs and proposed an object category specific MRFs model for detecting and segmenting instances of a particular object category. The object-specific shape prior is represented using pictorial structures, and it relies on a large library of exemplars. Winn & Shotton (2006) adopted a part labeling which densely covers the object and proposed a Layout Consistent Random Field (LayoutCRF) model to impose asymmetric local spatial constraints on these labels to ensure the consistent layout of parts. It can detect and segment partially occluded objects of a known category. They introduced shape priori for objects and layout of object parts. Therefore, single object or a small number of objects can be segmented cleanly.

Since the number of objects presented in remotely sensed image simultaneously is large, object shapes and scene topology are too complex to be modeled using above mentioned approaches. On the other hand, it is important to utilize such information to segment object precisely.

12.2.2 Marked Point Processes based Representation

Marked point processes provide a useful representation of spatially distributed objects. A set of points randomly distributed are used to represent objects. The number of points, their positions, and their interactions are random. Furthermore, marks are attached to each point to represent high-level knowledge such as category or geometric shape. Marked point processes are flexible enough to model the scene at the object level. Given a proposed model, reversible jump Markov Chain Monte-Carlo (Green, 1995) can be adopted to explore the configuration space and annealing schema can be adopted to simulate the objective distribution to find the optimal solution.

The Ariana Research Group CNRS/INRIA/UNSA introduced marked point process into remotely sensed image analysis. They demonstrated that marked point processes have a great potential in object extraction from remotely sensed image. Ortner *et al.* (2007, 2008) adopted rectangle to represent building footprint, and proposed an approach for building footprint extraction from DSM. Lafarge *et al.* (2008) adopted a 3D model to represented buildings, and proposed an approach for building reconstruction from DSM based on a library of 3D models. Tournaire *et al.* (2010) adopted above framework and formulated the energy in an efficient way, easy to parameterization and fast to compute.

The devised marks, however, can only represent specific shapes. Due to computational limitation, it is impossible to adopt a huge number of marks to represent general shapes. And, general shapes can not be determined based on image content adaptively. Moreover, images are linked with the model via a data term computed using hypothesis testing schema, which can not make full use of low-level information.

12.3 Bayesian Framework for Building Extraction

As a whole, we represent buildings as foreground X and the rest as background $\overline{X}$, and formulate building extraction as foreground/background segmentation in a Bayesian manner. Given

the observed image I, the buildings X can be estimated by maximizing the posterior:

$$X = \arg\max_X P(X, \overline{X}|I; \Theta) \tag{12.1}$$

$$= \arg\max_X P(I|X, \overline{X}; \Theta) P(X, \overline{X}|\Theta) \tag{12.2}$$

where, Θ is a set of models and parameters, $P(X, \overline{X}|\Theta)$ is the priori probability of a specific configuration $X, \overline{X}$ conditioned on Θ, and $P(I|X, \overline{X}; \Theta)$ is the likelihood of observing an image I given the configuration $X, \overline{X}$ conditioned on Θ.

We will address issues of modeling and estimation in Section 12.4 and Section 12.5 respectively.

12.4 Modeling

12.4.1 Hybrid Representation

We combine marked point processes and Markov random fields and propose a hybrid representation for building extraction. As illustrated in Figure 12.1, marked point process is adopted to represent the high-level knowledge, i.e. the buildings and their distribution; Markov random field is adopted to represent the low-level information, i.e. the properties of all pixels.

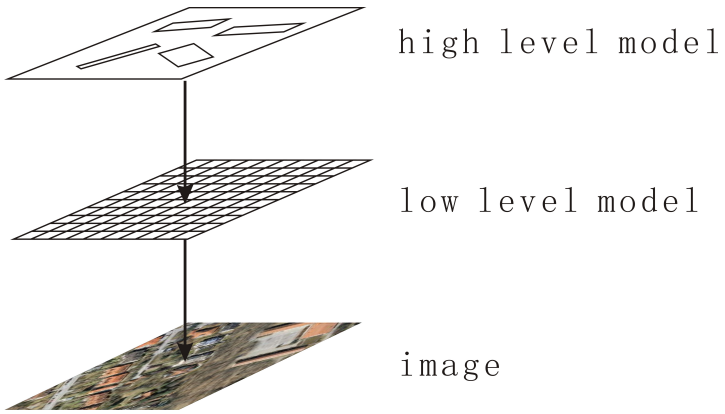


Figure 12.1: Hybrid representation.

12.4.2 High-level Model

At high level, only buildings are represented explicitly, the rests are represented implicitly. The buildings are modeled as a marked point process.

Marked Point Process A rigorous definition of spatial point process involves measure theory which is difficult for the readers who have not studied the subject. Instead, we present only a simple description in this section. Let $S \subset R^2$ be a compact set, and Ω_n be the set of

configurations $X = \{x_1, ..., x_n\}$ that consist of n unordered points of S, the probability density of a specific configuration X is defined as follows:

$$h(X) = \alpha\beta^{n(X)} \prod_{x_i \in X} \phi^{(1)}(x_i) \cdots \prod_{(x_{i_1},...,x_{i_k}) \in X} \phi^{(k)}(x_{i_1}, ..., x_{i_k}) \tag{12.3}$$

where, α is the normalizing constant, $n(X)$ is the number of points, β is a positive constant, and $\phi^{(k)}(x_{i_1}, ..., x_{i_k})$ reflects the interactions among k-tuplets neighboring points.

By attaching a geometric mark $m_i = (l_i, w_i, dir_i)$ to each point $x_i \in X$, we can augment a spatial point process to be a marked point process, where, the triplet denotes the length, width and main direction of a rectangle. The marked point is denoted as $\widehat{x_i}$. All buildings are represented by a rectangle-marked point process.

Return to Bayesian formulation in Section 12.3, $h(X)$ is computed from a prior term and a data term:

$$h(X) \propto h_p(X) h_d(X) \tag{12.4}$$

where, the prior term $h_p(X)$ measures the priori probability of different scenes, the data term $h_d(X)$ measures the coherence between the scene and the image, which is identified with the likelihood term. $h_p(X)$ and $h_d(X)$ correspond to $P(X, \overline{X}|\Theta)$ and $P(X, \overline{X}|\Theta)$ in Eq. 12.2 respectively.

Priori Most existing approaches rely on very specific priori (Lafarge & Descombes, 2010; Ortner *et al.*, 2007). Benedek *et al.* (2010); Lafarge & Descombes (2010) adopted the relation `overlap` as a general interaction and developed a concise priori model.

In this work, we argue that buildings can not overlap with each other and use this condition as a hard constraint. It is realized by defining the following density:

$$h_p(X) \propto \exp(\theta_0 o(X)) \tag{12.5}$$

where $\theta_0 = -\infty$ prevents overlapped rectangles to appear in configuration, $o(X)$ is the number of pairs of overlapped rectangles:

$$o(X) = \sum_{x_i, x_j \in X} \text{overlap}(\hat{x}_i, \hat{x}_j) \tag{12.6}$$

where,

$$\text{overlap}(\hat{x}_i, \hat{x}_j) = \begin{cases} 1, & \hat{x}_i \text{ overlaps with } \hat{x}_j \\ 0, & \hat{x}_i \text{ does not overlap with } \hat{x}_j \end{cases} \tag{12.7}$$

To reflect the sparse distribution of buildings, small distances between neighboring buildings are penalized. We augment above model by defining the density of valid configuration as unconditional Strauss process (Strauss, 1975):

$$h_p(X) \propto \exp(\theta_1 n(X)) + \theta_2 p(X, r)) \tag{12.8}$$

where θ_1 and θ_2 are two parameters of this model, $n(X)$ is the number of points in X, $p(X, r)$

is the number of pairs of points that are nearer than r:

$$p(X, r) = \sum_{x_i, x_j \in X} d(x_i, x_j) \leq r \tag{12.9}$$

where, $d(x_i, x_j)$ is the distance between x_i and x_j.

To reflect parallel of buildings, large difference of directions between neighbouring buildings are penalized. We augment above model by adding a direction term:

$$h_p(X) \propto \exp(\theta_3 q(X, r)) \tag{12.10}$$

where, $q(X, r)$ is sum of square of direction differences between neighboring buildings no far than r:

$$q(X, r) = \sum_{x_i, x_j \in X, d(x_i, x_j) \leq r} \alpha^2(\hat{x}_i, \hat{x}_j) \tag{12.11}$$

where, $\alpha(\hat{x}_i, \hat{x}_j)$ is the difference of the directions of x_i and x_j:

$$\alpha(\hat{x}_i, \hat{x}_j) = dir(\hat{x}_i) - dir(\hat{x}_j) \tag{12.12}$$

where, $dir(\hat{x}_i)$ is the main direction of the rectangle attaching to x_i. Furthermore, $\alpha(\hat{x}_i, \hat{x}_j)$ is compared with $0, \pi/2, ..., 2\pi$ and the minimal difference is adopted as the result. This measure can reflect both parallel and orthogonal of buildings.

Finally, we combine above models and get a full distribution as follows:

$$h_p(X) \propto \exp(\theta_0 o(X)) + \theta_2 p(X, r) + \theta_3 q(X, r)) \tag{12.13}$$

where the first term assures that neighboring buildings do not overlap with each other, the second and third terms assure that all building distribute sparsely, the last term assures that neighboring buildings align with each other. $\theta_0 \in \Theta, \theta_1 \in \Theta, \theta_2 \in \Theta, \theta_3 \in \Theta, r \in \Theta$ are all parameters of the model.

Data Term The data term measures the coherence between the scene and the image, i.e. the likelihood of presenting an image given the scene:

$$h_d(X) = P(I|X, \widehat{X}; \Theta) \tag{12.14}$$

Since the low-level model links both high-level model and image, it serves as an intermediate model between marked point process and image, the data term is calculated based on low-level model. Details will be presented in section 12.4.4.

12.4.3 Low-level Model

At the low level, both foreground and background are modeled explicitly and together as a Markov Random Field.

Markov Random Field Since foreground and background can be denoted by two labels, we model them as an Ising model (Li, 2009). The probability of presenting a specific configuration f is computed as follows:

$$P(f) = \frac{1}{Z}\exp(-U(f)) \propto \exp(-(U_p(f) + U_d(f))) \tag{12.15}$$

where, Z is a normalizing constant called the partition function, which is common to all configurations and can be ignored in computation. $U_p(f)$ and $U_d(f)$ correspond to $P(X, \overline{X}|\Theta)$ and $P(I|X, \overline{X}|\Theta)$ in Eq. (12.2) respectively.

Priori The priori energy $U_p(f)$ for Ising model is calculated as follows:

$$U_p(f) = \sum_{i,j} |f_i - f_j| \tag{12.16}$$

where, i and j are horizontally or vertically neighboring pixels, all neighboring pixels with different labels contribute to the total energy.

Above priori term is based on the assumption that the random field varies smoothly everywhere. Every pair of pixels with different labels will increase the priori energy and decrease the probability. In fact, neighboring pixels in foreground or background regions should have the same labels. Neighboring pixels across the region boundaries should have different labels. In other words, label field should be allowed to change at region boundaries without increasing the priori energy.

To reflect such priori knowledge, neighboring pixels across regions should contribute nothing to the priori energy, while as neighboring pixels within regions should contribute to the priori energy as traditional way. We augment above priori term to be:

$$U_p(f) = \sum_{i,j} \beta_{i,j}|f_i - f_j| \tag{12.17}$$

where, if i and j across foreground and background regions, $\beta_{i,j}$ should be 0; otherwise, it should be 1.

Such discontinuity preserving constraint is more reasonable than the simplest constraint making configuration varies smoothly everywhere. However, It is difficult to express the discontinuity preserving constraints because nothing is known in advance about the regions and their boundaries. In the proposed hybrid representation, the high-level model provides a guess of such knowledge, it can be utilized to calculate $\beta_{i,j}$. Details will be presented in Section 12.4.4.

Data Term The data term $U_d(f)$ corresponds to the likelihood is defined as follows:

$$U_d(f) = U_d(I|f) = \sum_i U(I_i|f) \tag{12.18}$$

where, the energy $U(I_i|f)$ is calculated as follows:

$$U_d(I_i|f) = \{ \begin{array}{ll} -\log(P(I_i|H_b)), & f_i = 0 \\ -\log(P(I_i|H_f)), & f_i = 1 \end{array} \tag{12.19}$$

where, $H_n \in \Theta$ and $H_f \in \Theta$, are two normalized histograms for background and foreground respectively. $P(I_i|H_b)$ and $P(I_i|H_f)$ measures the likelihood of the pixel with color I_i belonging to background and foreground respectively.

12.4.4 Linking High-level and Low-level Models

Each marked point at the high level denotes one building, and it corresponds to one rectangular region in the Markov random field at the low level. The high-level model and the low-level model are combined together by establishing correspondences of marked points at the high level and regions (each one consists of a set of pixels) at the low level. High-level knowledge is introduced as a priori term in the MRF and low-level information is introduced into data term in the marked point process. In this way, a flexible and robust representation is achieved.

Priori As pointed out in Section 12.4.3, high-level knowledge can be utilized to construct a discontinuity preserving a priori term for the low-level model.

Suppose that there are a set of marked points at the high level, we can get a set of rectangular regions at low level by projecting the marked points on to the grid of Markov random field. Each pixel i has one label f_i which denotes its class, i. e. foreground or background. Without loss of generality, suppose that i and j are neighboring pixels. The discontinuity preserving priori term is constructed by defining $\beta_{i,j}$ as follows:

$$\beta_{i,j} = \{ \begin{array}{ll} 0, & f_i \neq f_j \\ 1, & f_i \neq f_j, \end{array} \tag{12.20}$$

i.e. the label configurations of the Markov random field are not evaluated at the borders induced by the marked point process.

Data Term Suppose that there is a set of marked points at the high level. We can obtain a set of rectangular regions at the low level by projecting the marked points on to the grid of Markov random field. Each pixel i has one label f_i which denotes its class, i.e. foreground or background. Using Eq. (12.18), we can calculate the data term for the high level model.

More specifically, summing the data terms over one rectangular region, we get the data term corresponding to one marked point at the high level.

12.5 Optimization

We adopt simulated annealing to simulate the posterior distribution so that an optimal configuration can be achieved as the temperature gradually approaches zero. It iteratively simulates the distribution $h^{\frac{1}{T}}(X)$ with T gradually decreasing to 0.

Furthermore, we use reversible jump Markov random Monte Carlo (rjMCMC) techniques to explore the configuration space at the high level, an uniform birth and death kernel and a translation kernel are developed to generate new states according the current state. The former randomly generates a point together with a rectangle (length, width and direction) in the image region, while the latter randomly selects an existing rectangle and adjusts its parameters randomly. The new state is adopted with an accept rate to keep the detailed balance.

What distinguishes our approach from existing ones is that a top-down schema and a bottom-up schema are proposed for random sampling. In each sample, it first generates a new state at high level, then uses it to guide the optimization at low level, then uses the optimization results to adjust state at high level.

12.5.1 Top-down Schema

Since there are a large number of buildings presented in remotely sensed image simultaneously, previous MRF based approaches need seed pixels provided manually. Otherwise, neighboring buildings can not be distinguished well without knowledge about their spatial distribution. We, however, use the results at the high level to provide the information about the spatial distribution and approximate shapes of buildings.

Low-level optimization is conducted only when a new marked point is birthed. Given a new states, i.e. a new rectangle-marked point. we use it to guide the optimization at low level:

1. Project the rectangle into image to get projected regions;
2. Construct discontinuity preserving priori term for low-level model;
3. Adopt Graph Cut algorithm (Boykov *et al.*, 2001; Greig *et al.*, 1989) to optimize the proposed object function with discontinuity preserving priori term, the optimization is conducted in the projected regions and it results in some building regions.

12.5.2 Bottom-up Schema

Motivated by the data driven MCMC (Tu & Zhu, 2002), we use results at low level to drive the sampling, i.e. compute new state according results of optimization at low level. In their data driven MCMC, the low-level results are computed by edge detection or segmentation and are fixed in the process of MCMC sampling. In our approach, the low-level results are computed from low level optimization and vary in the process of MCMC sampling.

Given the optimization results achieved at low level, i.e. some building regions at low level, we use them to adjust the new state at high level:

1. Select the largest one and find a minimal rectangle enclosing the selected region;
2. Adjust the new marked point to be the rectangular region found above;
3. Calculate the data term based on the optimization result;
4. Calculate the accept rate using the data term calculated above;

5. Move the current state to the (adjusted) new state with the accept rate.

Benefited from optimization at low level, more precise building region can be found in each random sampling, this may improve the definition of both shape and data term. As a result, this schema may drive rjMCMC sampling to achieve better results. Theoretical foundation is guaranteed by simulated annealing and rjMCMC technique.

12.6 Experiments

12.6.1 Results and Comparison

We apply our approach to extract buildings from three satellite images of developed urban or suburban areas. For comparison, we also apply marked point processes based approach and Markov random fields based approach to extract buildings.

In this experiment, the histograms H_b and H_f (i.e. three dimensional arrays) are learned from real images by manually annotating images. The rest parameters are set as follows: $\theta_0 = -\infty, \theta_1 = 1, \theta_2 = -0.01, \theta_3 = -0.01, r = 10$ pixels.

The original images, manually annotated reference images, and extraction results are presented in Figure 12.2. As illustrated, there are some clear errors in the third row achieved by marked point processes based approach. Since the data term for a rectangle is calculated as a whole but not pixelwise, some regions that contain buildings may be recognized as a building region by mistake. Theoretically, infinite sampling can remove such cases, however, it can not be achieved in practice.

As illustrated, there are many buildings missed in the fourth row achieved by Markov random fields based approach. Since the priori term (smooth term) drive the results as smooth as possible, some regions with low density of being buildings are segmented as background by mistake. On the contrary, some regions between neighboring buildings are segmented as foreground. As illustrated, the last row achieved by our hybrid representation is better than above rows. Benefited from the hybrid representation, both high-level knowledge and low-level information are well-represented and utilized in the process of building extraction. The point distribution at high level provides a topology structure of the scene. Based on the topology, the optimization at the low level is expected to achieve robust results. The detailed data terms computed at low level improve foreground and background distinguishing at high level.

We also recorded the number of building pixels extracted as building pixels, building pixels extracted as non-building pixels, and non-building pixels extracted as building pixels. They are divided by the total number of building pixels and presented in Table 12.1, Table 12.2 and

Table 12.1: Quantitative evaluation on first image.

	True	False Positive	False Negative
MPP-based	0.53	0.47	0.14
MRF-based	0.41	0.59	0.04
Hybrid-based	0.52	0.48	0.09

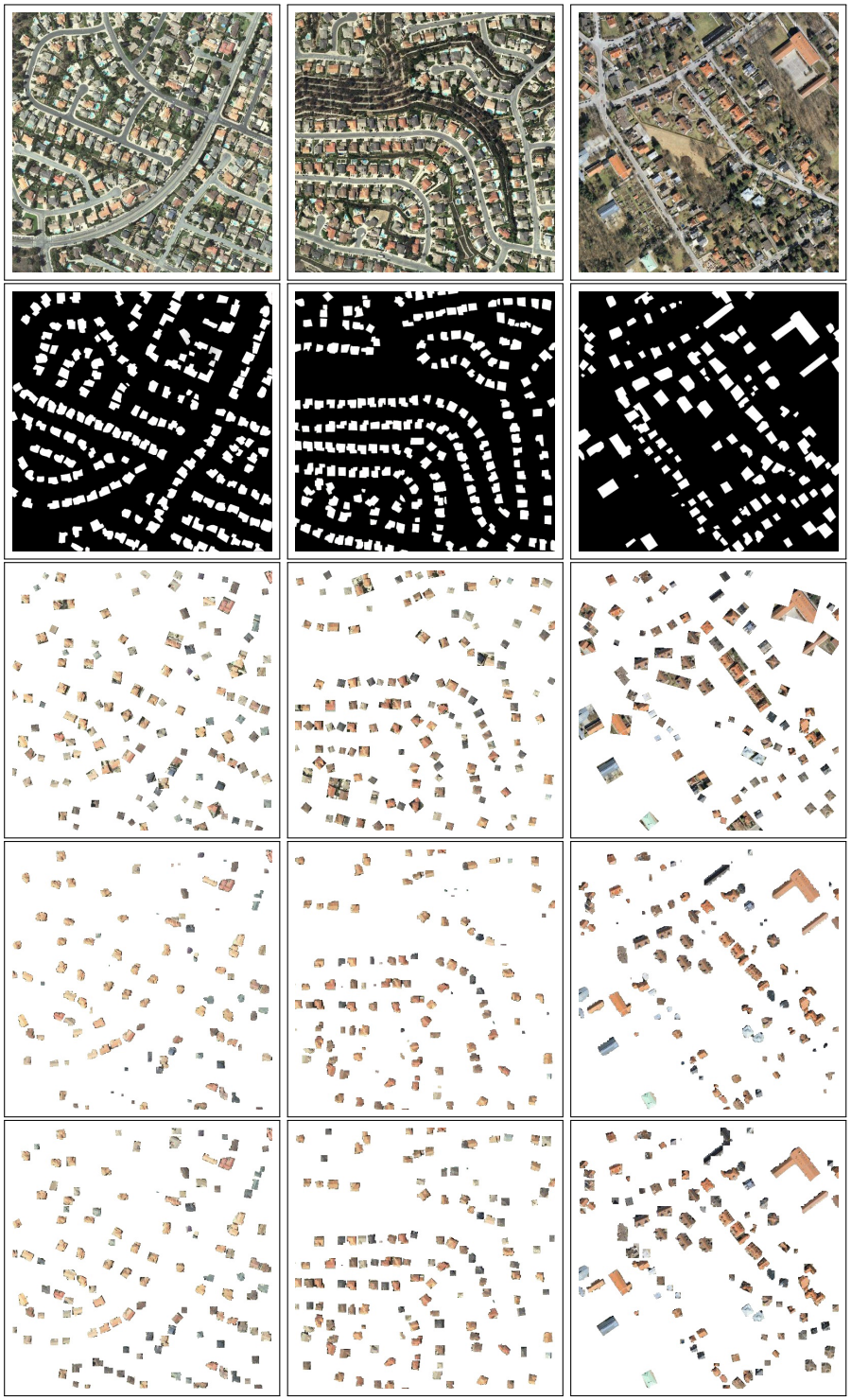


Figure 12.2: Extraction results: the first row illustrates the original images, the second row illustrates the reference extraction results, which are annotated manually, the rest rows illustrate extraction results based on Marked point processes, Markov random fields, and hybrid representation.

Table 12.3 respectively. As illustrated, the results achieved by hybrid representation are much better than those of Markov random fields based approach, they are also better than those of marked point processes based approach, especially indicated by False Negative, which means the number of non-building pixels extracted as building pixels.

Table 12.2: Quantitative evaluation on second image.

	True	False Positive	False Negative
MPP-based	0.56	0.44	0.15
MRF-based	0.44	0.56	0.04
Hybrid-based	0.57	0.43	0.08

Table 12.3: Quantitative evaluation on third image.

	True	False Positive	False Negative
MPP-based	0.74	0.26	0.34
MRF-based	0.69	0.31	0.07
Hybrid-based	0.78	0.22	0.13

12.7 Conclusion

This work presents a hybrid representation for buildings in remotely sensed image and an approach for building extraction from single remotely sensed image. First, it formulates building extraction in a Bayesian framework. Then, it addresses modeling issue and optimization issue respectively. Buildings are modeled at two levels. At the high level, marked point processes are used to represent such high-level knowledge as topology structure of a scene. At the low level, a Markov random field is used to represent pixel color and interaction. After establishing a link between high-level model and low-level model, it proposes a top-down schema and a bottom-up schema optimizing an objective function. Benefited from the hybrid representation and optimization schema, good extraction results are achieved as demonstrated by experiments presented in this work.

To our knowledge, it is the first work on the combination of marked point process and Markov random fields. Therefore, there are many issues to be investigated in the near future. First, the optimization schema can be improved greatly since the interactions between high-level model and low-level model are not fully utilized. Second, much more information from the image data need to be explored to improve the extraction quality since histograms of colors do not fully represent information contained in the images, this can be seen in the density images calculated using histograms.

Chapter 13

Multi-Source Multi-Scale Hierarchical Conditional Random Field Model for Remote Sensing Image Classification

Fusion of remote sensing images and LiDAR data provides complimentary information for the remote sensing applications, such as object classification and recognition. In this chapter, we propose a novel multi-source multi-scale hierarchical conditional random field (MSMSH-CRF) model to integrate features extracted from remote sensing images and LiDAR point cloud data for image classification. MSMSH-CRF model is then constructed to exploit the features, category compatibility of multi-scale images and the category consistency of multi-source data based on the regions. The output of the model represents the optimal results of the image classification. We have evaluated the precision and robustness of the proposed method on airborne data, which shows that the proposed method outperforms standard CRF method. This research appears at the ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Photogrammetric Image Analysis (PIA) (Zhang *et al.*, 2015).

13.1 Introduction

In the fields of photogrammetry and remote sensing, there exist many sources of earth observation data with the different characteristics of targets on the ground. For a long period, integration of the multi-source data reasonably and effectively has been an active topic. Fusion of remote sensing images and LiDAR data provides complimentary information for the remote sensing applications, such as object classification and recognition.

Many methods have been developed for the fusion of remote sensing images and LiDAR data. In general those methods are classified into three categories, namely image fusion (Parmehr *et al.*, 2012), feature fusion (Dalponte *et al.*, 2012; Deng *et al.*, 2012), and decision fusion (Huang *et al.*, 2011; Shimoni *et al.*, 2011). The methods for image fusion include different resolution data sampling and registration, so the processing is time-consuming, and the accuracy is affected by the accuracy of registration, which reduces the performance of the sub-

sequent image classification. In the feature fusion methods, the features are usually extracted independently from different data source, and the fusion lacks consideration of correspondence of location and contextual information, by which the classification could be improved.

In order to overcome the limitations of the aforementioned methods, we present a novel multi-source multi-scale hierarchical conditional random field (MSMSH-CRF) model to fuse features extracted from remote sensing images and LiDAR point cloud data for image classification. In this work, the major **contribution** is that both the category compatibility of the multi-scale image in a hierarchical structure and the category consistency of multi-source data are considered in the MSMSH-CRF model. The following sections are organized as follows. The related work is discussed in Section 13.2. In Section 13.3, the MSMSH-CRF model is presented in detail. In Section 13.4, experimental results are presented. Finally, this contribution of this work is concluded and the future work is discussed in Section 13.5.

13.2 Related Work

In order to make full use of multi-source data for image classification and object recognition, many feature-based fusion methods have been proposed. One of the classic tools are graphical models (Bishop, 2006), i.e. probabilistic models defined on a graph describing the conditional dependence structure between random variables. As the one branch of the graphical model, Markov Random Fields (MRFs) have been used for image interpretation since 1986 (Besag, 1986), and their limiting factor only allowing for local image features has been overcome by Conditional Random Fields (CRFs) (Lafferty *et al.*, 2001), where arbitrary features can be used for classification. CRFs have the ability to discriminatively model contextual dependencies, conditioned on observations, for capturing global as well as local image context, which makes them suitable for accurate labeling (Perez *et al.*, 2012). Therefore, they have been receiving more and more attention in recent years (Niemeyer *et al.*, 2014; Yang & Förstner, 2011b; Zhang *et al.*, 2012).

Schindler (2012) gives a systematic overview of image classification methods, which impose a smoothness prior on the labels. Both local filtering-type approaches and global random field models developed in other fields of image processing are reviewed. He shows a detailed experimental comparison and analysis of the methods, using two different aerial data sets from urban areas with known ground-truth. Based on the standard CRF model (Shotton *et al.*, 2009), Yang & Förstner (2011a) introduce a hierarchical conditional random field to deal with the problem of image classification by modeling spatial and hierarchical structures. Perez *et al.* (2012) formulate a multi-scale CRF model to deal with the problem of region labeling in multi-spectral remote sensing images. Zhang *et al.* (2013) propose the multi-source hierarchical conditional random field (MSHCRF) model to fuse features extracted from remote sensing images and LiDAR point cloud data for image classification. Hierarchical pairwise potentials are introduced to consider category consistency of multi-source data based on regions. Niemeyer *et al.* (2014) integrate a random forest classifier into a CRF framework, which is a flexible method for obtaining a reliable 3D classification in complex urban scenes. These methods exploit both spatial and hierarchical structures of objects in images. Considering the limitation of visual feature information from the images, the classification results could

be potentially improved by incorporating information from different source data, such as the elevation information in LiDAR data and the spectral information in the hyperspectral images.

13.3 MSMSH-CRF Model for Automatic Classification

In this section, we start by presenting the graphical model to integrate an image and LiDAR data, so-called MSMSH-CRF model, with corresponding energy function. Then, we describe the model construction process. Afterward, we will derive the features from each region obtained from the unsupervised segmentation algorithm. Then, we will give particular formulations for each of the unary, pairwise, hierarchical potentials respectively. Finally, we will discuss the learning and inference of this graphical model.

13.3.1 MSMSH-CRF Model

In the field of image analysis, the regions of interest are usually detected independently, but considering the relative position between regions in single source data and the correspondence between regions from multi-source data, the labeling of every region should not be independent. The CRF model is an effective way to solve the problem of prediction of the non-independent labeling for multiple outputs, and in this model, all the features can be normalized globally to obtain the global optimal solution.

Based on the standard CRF model, we propose the MSMSH-CRF model to learn the conditional distributions over the class labeling given an image and corresponding LiDAR data, and the model allows us to incorporate different features and correspondence information in a single unified model, as illustrated in Figure 13.3. The conditional probability of the class labels c given an image X and LiDAR data L, which has a distribution of the Gibbs form, is defined as follows

$$P(\mathbf{c}|\mathbf{X},\mathbf{L},\theta) = \frac{1}{Z(\theta,X,L)}\exp(-E(\mathbf{c}|X,L,\theta)) \tag{13.1}$$

And the energy function

$$\begin{aligned} E(\mathbf{c}|\mathbf{X},\mathbf{L},\theta) = &\sum_{i\in S} E_1(c_i,x_i,\theta_{\mathbf{1}}) + \sum_{(i,j)\in N} E_2(c_i,c_j,x_i,x_j,\theta_2) \\ &+ \sum_{(i,k)\in M} E_3(c_i,c_k,x_i,x_k,\theta_3) + \sum_{(i,t)\in H} E_4(c_i,c_t,x_i,l_t,\theta_{\mathbf{4}}) \end{aligned} \tag{13.2}$$

where $\theta = \{\theta_{\mathbf{1}},\theta_{\mathbf{2}},\theta_3,\theta_{\mathbf{4}}\}$ is the vector of model parameters, $Z(\theta,\mathbf{X},\mathbf{L})$ is the partition function, i,j and k respectively index regions x_i,x_j and x_k in the image, which correspond to nodes in the graph, and t index regions l_t in the LiDAR data, which also correspond to nodes in the graph. S is the set of all the nodes in image level of the graph, N is the set of corresponding pairs collecting neighborhood in both images and LiDAR data, M is the set of pairs collecting parent-child relations between regions with neighboring scales, and H is the set of corresponding pairs collecting neighborhood in both images and LiDAR data. E_1 is the unary potentials, which represent relationships between class labels and the observed data, E_2 is the pairwise potentials, representing relationships between class labels of neighboring regions within each scale. E_3 is the multi-scale hierarchical pairwise potential, which represents

corresponding relationships between regions in neighboring scales of images. E_4 is the multi-source hierarchical pairwise potential, representing corresponding relationships between images and LiDAR data.

13.3.2 Model Construction

In order to integrate features extracted from multi-source data for image classification, the MSMSH-CRF graphical model is consist of two levels: Image level and LiDAR level. In Image level, Texton is utilized to distinguish between different regions effectively and obtain the different segmented regions, which form all the nodes in Image level of the graph. Meanwhile, we can change the amount of channels of the Texton filter (Shotton *et al.*, 2009) to get different results which are similar to the multi-scale segmentation, and Figure 13.1 shows the example results of our algorithm. The neighborhood in Image level is defined as the relationship of two regions which have the common edge. In LiDAR level, the mean shift algorithm is used to get the flat regions corresponding to continuous planes of different targets in LiDAR data, which form all the nodes in LiDAR level of the graph.

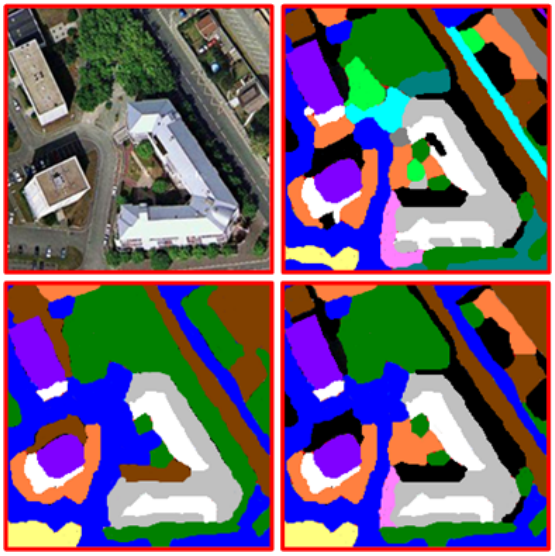


Figure 13.1: The example region images of Texton segmentation results at scale 1, 2, 3 respectively. The color of each region is assigned randomly that neighboring regions are likely to have different colors. *Top row left*: Original image, *Top row right*: segmentation result at scale 3. *Bottom row left*: segmentation result at scale 1, *Bottom row right*: segmentation result at scale 2.

For describing the consistency of multi-source data, we firstly choose the optimal scale of images to match with the LiDAR data. Assuming that there is a registration of multi-source data acquired on the same airborne platform, such as the algorithm introduced in literature (Mastin *et al.*, 2009), and we calculate the center of each region (or line)RL_i in the depth image converted from LiDAR data, and the center should be inside the region (or line) and at the symmetric axis. Then based on the relative position of the centers, the corresponding regions (or lines) RL_{ia} in multi-scale images can be selected. The procedure of choosing optical scale images is illustrated in Figure 13.2. Therefore, for each pixel s

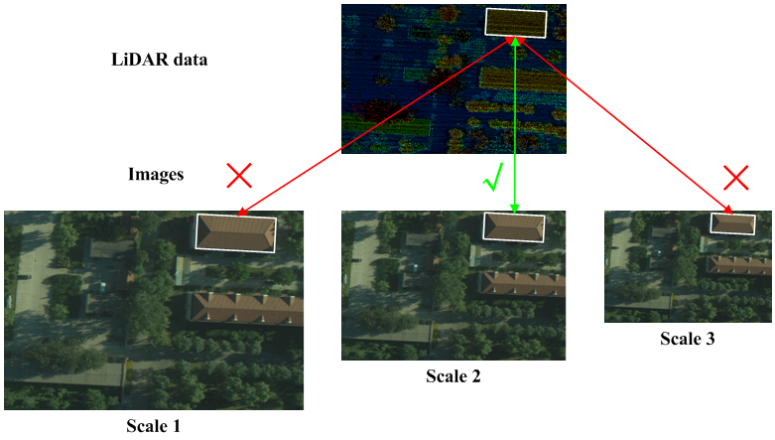


Figure 13.2: The example image of illustrating the procedure of choosing the optimal scale image to match the LiDAR data.

in the region (or line) RL_i, we obtain the optimal scale of images by

$$a^* = \arg\min_{a} \sum_{s \in \{RI_i \cup RI_{ia}\}} \mid RI_i(s) - RI_{ia}(s) \mid \tag{13.3}$$

and

$$RL_i(s) = \begin{cases} 1, & s \in RL_i, \\ 0, & s \notin RL_i \end{cases}, RL_{ia}(S) = \begin{cases} 1, & s \in RL_{ia}, \\ 0, & s \notin RL_{ia} \end{cases} \tag{13.4}$$

where i index the sequence number of all regions (or lines) in the depth image converted from the Mean Shift Feature (MSF) or Alpha Shape Feature (ASF) of LiDAR data.

Therefore, the MSMSH-CRF graphical model is constructed as follows, illustrated in Figure 13.3. Firstly, typical features are derived from the interest regions in multi-source data, where the regions are generated by an unsupervised segmentation algorithm. In the graphical model, the nodes correspond to regions. The blue edges represent the dependencies between neighboring regions, and the orange edges indicate the hierarchical relations between regions at different scales in a multi-scale segmentation. Purple edges indicate the hierarchical relations between regions from multi-source data, where the optimal scale of images is selected to match the LiDAR data. The MSMSH-CRF model is constructed to exploit the features and category compatibility of multi-scale images as well as the category consistency of multi-source data based on regions. The output of the model represents the optimal results of the image classification.

13.3.3 Features

Four types of features are extracted, namely the line features (LF), the texture features (TF), the mean shift features (MSF), and alpha shape features (ASF). The line features (LF) and the texture features (TF) are extracted from remote sensing images, whereas the mean shift features (MSF) and alpha shape features (ASF) are from LiDAR data.

Line Features (LF) Shape features, in particular line features, not only describe the structures of targets directly, but also are stable to light change, color change, etc. As a new and effective one of line

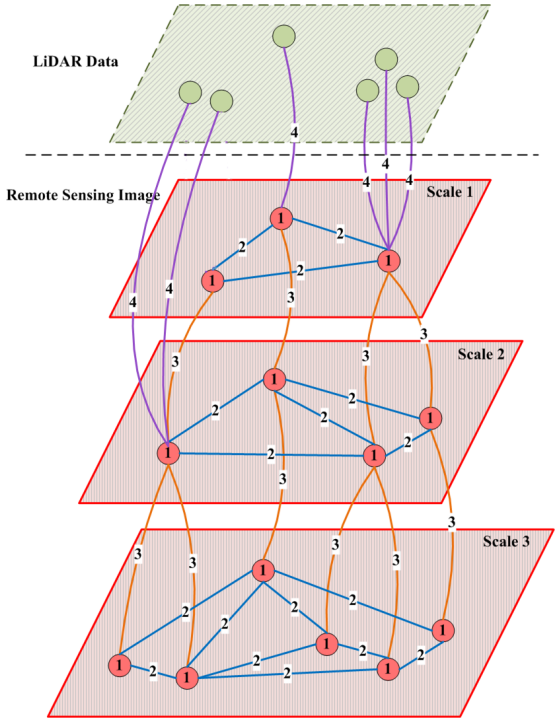


Figure 13.3: Illustration of the MSMSH-CRF model architecture. In Image level, red nodes (# 1) correspond to image regions, blue edges (# 2) represent the dependency between neighboring regions, and orange edges (# 3) indicate the hierarchical relation between regions at different scales. In LiDAR level, green nodes represent the extracted regions. Purple edges (# 4) linking red and green nodes indicate the hierarchical relation between regions from multi-source data, where the optimal scale of images is selected to match the LiDAR data.

features, the LSD (Line Segment Detector) (Grompone *et al.*, 2010) can be used to give accurate results extracted, a controlled number of false detections, and requires no parameter tuning. In the method, the level-line orientation is defined and calculated by gradient magnitude, and then the pixels with the same level-line orientation are merged to cover the so-called line support regions, in which all the pixels are regarded as a long continuous segment. In accordance with the method introduced in Grompone *et al.* (2010), we can calculate the response value of LSD at each pixel, denoted by $LF(s)$.

Texture Features (TF) Texture is one of the basic properties of objects, as well as the most direct and reliable way of characterization. The basic unit of texture is often referred to as Texton, and we can represent the texture most directly by describing the distribution of the components, namely Texton. In the process of textonization, images are convolved with a 17-dimensional filter-bank. The 17D responses for all training pixels are then whitened (to give zero mean and unit covariance), and an unsupervised clustering is performed by the Euclidean-distance K-means clustering algorithm. Finally, each pixel in each image is assigned to the nearest cluster center, producing the Texton map. Similar to the method in Shotton *et al.* (2009), we can obtain the value of Texton classifier of each pixel in the image, denoted by $TF(s)$.

Mean Shift Features (MSF) The mean shift method (Comaniciu & Meer, 2002) is a robust clustering technique which does not require prior knowledge of the number of clusters, and does not constrain the shape of the clusters. The number of clusters is obtained automatically by finding the centers of the densest regions in the space, so this method is widely used for clustering of discrete points. In our model, the specific process of achieving the MSF is introduced in Georgescu *et al.* (2003), all the LiDAR points are clustered in different regions, and the elevation of all points in one region are assigned as the same value which is the mean of all the ones.

Alpha Shape Features (ASF) There are many methods for extracting the boundary of LiDAR data. Compared with other algorithms (Berger, 2012; Kong *et al.*, 2012), Alpha Shapes algorithm works effectively in inner and outer boundaries extraction from LiDAR data with convex and concave polygon shape. Moreover, it can keep fine features of buildings adaptively and filter the footprints of non-building. Based on the MSF regions obtained, the alpha shape algorithm is used to extract the boundary contour of each region, and then the Delaunay triangulation is used to get the line feature. The extraction of the ASF refers to Shen *et al.* (2011), similar to the MSF, all the points in one lines have the same elevation which is the mean of all the ones.

13.3.4 Unary Potentials

The unary potentials consist of two element, LF and TF potentials, predict the label c_i of the region x_i based on the image $\mathbf{X}$

$$E_1(c_i, x_i, \theta_\mathbf{1}) = LF(c_i, x_i, \theta_{LF}) + TF(c_i, x_i, \theta_{TF}) \tag{13.5}$$

where $LF(c_i, x_i, \theta_{LF})$ is the LF potential and $TF(c_i, x_i, \theta_{TF})$ is the TF potential, and $\theta_\mathbf{1} = \{\theta_{LF}, \theta_{TF}\}$ is the vector of model parameters.

LF Potentials The LF potentials capture the (relatively weak) dependence of the class label and the boundaries of targets on the response value of LSD and absolute location of the pixel in the image. We can get the line segment image $LFI(s)$ by calculating the response value of LSD LF_s of each pixel s in the region x_i. The LF potentials take the form of a look-up table with an entry for each class ci and value of LSD LF_s and pixel location s

$$LF(c_i, x_i; \theta_{LF}) = -\log \sum_{s \in x_i} \theta_{LF}(c_i, LF_s, s) \tag{13.6}$$

where the parameter θ_{LF} represents the relationship among the value of each pixel LF_s, namely $LFI(s)$, the pixel location s and the label c_i.

TF Potentials Based on the Joint Boost algorithm, an adapted version of boosting learning algorithm, we can obtain the classifier of Texton, to which the responses are used directly as a potential in the MSMSH-CRF model, so that

$$TF(c_i, x_i; \theta_{TF}) = -\log \sum_{s \in x_i} P(c_i | TF_s) \tag{13.7}$$

where TF_s corresponds to the response of classifier at each pixel s, and $P(c_i|TF_s)$ is the normalized distribution given by the classifier using the learned parameters θ_{TF}.

13.3.5 Pairwise Potentials

The pairwise potentials describe category compatibility between neighboring regions x_i and x_j obtained from the line segment image $LFI(s)$, and the responses of Texton classifier on the image $\mathbf{X}$.

$$E_2(c_i, c_j, x_i, x_j, \theta_{\mathbf{2}}) = PLF(c_i, c_j, x_i, x_j, \theta_{PLF}) + PTF(c_i, c_j, x_i, x_j, \theta_{PTF}) \tag{13.8}$$

where $PLF(c_i, c_j, x_i, x_j, \theta_{PLF})$ is the pairwise potentials of LF and $PTF(c_i, c_j, x_i, x_j, \theta_{PTF})$ is the pairwise potentials of TF, $\theta_{\mathbf{2}} = \{\theta_{PLF}, \theta_{PTF}\}$ is the vector of model parameters.

Pairwise Potentials of LF Based on the line segment image $LFI(s)$, we can calculate the pairwise potentials of LF as the form of the contrast-sensitive Potts model (Boykov & Jolly, 2001)

$$PLF(c_i, c_j, x_i, x_j, \theta_{PLF}) = \theta_{PLF} \frac{1 + 6\exp(-2l(x_i, x_j))}{N_i + N_j} \sigma(c_i \neq c_j) \tag{13.9}$$

where θ_{PLF} is the weight factor, $l(x_i, x_j)$ is the Euclidean metric of the pixel value between regions x_i and x_j in the LF images, N_i is the number of regions neighbored to region i, N_j is the number of regions neighbored to j, and $\sigma(\cdot)$ is a 0-1 indicator function, and the number 6 in Eq. (13.9) is set empirically. The pairwise potentials $PLF(c_i, c_j, x_i, x_j, \theta_{PLF})$ are scaled by N_i and N_j to compensate for the irregularity of the graph.

Pairwise Potentials of TF Similar to the pairwise potentials of LF, the pairwise potentials of TF take the form of the contrast-sensitive Potts model:

$$PTF(c_i, c_j, x_i, x_j, \theta_{PTF}) = \theta_{PTF} \frac{1 + 4\exp(-2l(x_i, x_j))}{N_i + N_j} \sigma(c_i \neq c_j) \tag{13.10}$$

where θ_{PTF} is the weight factor, $t(x_i, x_j)$ is the Euclidean metric of the value of Texton classifier at each pixel between regions x_i and x_j in the results of marked images, and the number 4 in Eq. (13.10) is set empirically. The pairwise potentials PTF are scaled by N_i and N_j to compensate for the irregularity of the graph.

13.3.6 Multi-scale Hierarchical Pairwise Potentials

From the pairwise potentials in Section 13.3.5, there is a lack of longer range contextual relationship in the graphical modeling. To overcome those local restrictions, we analyze the image at multiple scales to enhance the model by evidence aggregation on a local to global level. Furthermore, we integrate multi-scale pairwise potentials to regard the hierarchical structure of the regions.

Based on results of multi-scale segmentation, the multi-scale hierarchical pairwise potentials describe category compatibility between hierarchically neighboring labels c_i and c_k given the image $\mathbf{X}$, which take the form of the contrast-sensitive Potts model:

$$E_3(c_i, c_k, x_i, x_k, \theta_3) = \theta_3 \cdot [1 + 4\exp(-2m(x_i, x_j))]\, \sigma(c_i \neq c_k) \tag{13.11}$$

where θ_3 is the weight factor, $m(x_i, x_j)$ is the Euclidean metric of the value of Texton classifier between regions x_i and x_j in the results of marked images, and the number 4 in Eq. (13.11) is set empirically. Multi-scale hierarchical pairwise potentials act as a link across scale, facilitating propagation of information in the model.

13.3.7 Multi-source Hierarchical Pairwise Potentials

Compared to the remote sensing images, LiDAR data is sparse. The features extracted from multi-source data are different. In order to enhance the fusion performance, we introduce the hierarchical pairwise potentials, which represent correspondences between the data from different source in our MSMSH-CRF model. The hierarchical pairwise potentials describe category consistency between the corresponding regions in multi-source data, from which we can obtain the TF and MSF, which are named as planar features, and the LF and ASF, which are named as linear features. In order to enhance the fusion performance, we refer to the category consistency with the planar and linear features separately, denoted as $HPP(c_i, c_t, x_i, l_t, \theta_p)$ and $HPL(c_i, c_t, x_i, l_t, \theta_l)$ respectively. So there is

$$E_4(c_i, c_t, x_i, l_t, \theta_{\boldsymbol{4}}) = HPP(c_i, c_t, x_i, l_t, \theta_p) + HPL(c_i, c_t, x_i, l_t, \theta_l) \tag{13.12}$$

where $\theta_{\boldsymbol{4}} = \{\theta_p, \theta_l\}$ is the vector of model parameters.

Hierarchical Pairwise Potentials of Planar Features Based on the TF results of the optimal scale image, we firstly normalize the value $TF_s(x_i)$ of Texton classifier of each pixel s in the region x_i to get $NTF_s(x_i)$:

$$NTF_s(x_i) = TF_s(x_i)/TF_{max} \tag{13.13}$$

where TF_{max} is the maximum value of Texton classifier of each pixel in the image.

In the MSF results of LiDAR data, elevations of different regions are obtained, and the normalized elevation $NMSF(l_t)$ of all points in the regions l_t extracted is calculated:

$$NMSF(l_t) = MSF(l_t)/MSF_{max} \tag{13.14}$$

where $MSF(l_t)$ is the elevation of all points in the region l_t, and MSF_{max} is the maximum elevation of all flat regions in the LiDAR data.

So based on the normalized value $NTF_s(x_i)$ and $NMSF(l_t)$, the hierarchical pairwise potentials of planar features is defined by

$$HPP(c_i, c_t, x_i, l_t, \theta_p) = \theta_p \sum_{s \in x_i} \exp(-\epsilon_p |NTF_s(x_i) - NMSF(l_t)|^2) \sigma(c_i \neq c_t) \tag{13.15}$$

where $\epsilon_p = (2 < |NTF_s(x_i) - NMSF(l_t)|^2 >)^{-1}$ is the comparative item, $< \cdot >$ is the averaging operator, and θ_p is the weight.

Hierarchical Pairwise Potentials of Linear Features The hierarchical pairwise potentials of linear features take the form as

$$HPL(c_i, c_t, x_i, l_t, \theta_t) = \theta_t \sum_{s \in x_i} \exp(-\epsilon_t |NLF_s(x_i) - NASF(l_t)|^2) \sigma(c_i \neq c_t) \tag{13.16}$$

where $\epsilon_t = (< 2|NLF_s(x_i) - NASF(l_t)|^2 >)^{-1}$ is the comparative item, and θ_l is the weight. $NLF_s(x_i)$ is the normalized value from the LF results of the optimal scale image, and $NASF(l_t)$ is the normalized value from the ASF of LiDAR data.

13.3.8 Parameter Learning

In this work, piecewise training method (Sutton & McCallum, 2005) is adopted for the learning of the parameters of MSMSH-CRF model. This method divides the MSMSH-CRF model into pieces

corresponding to the different terms in Eq. (13.2). Each of these pieces is then trained independently, as if it were the only term in the model.

Parameters of LF Potentials The formula for calculating the parameters of LF Potentials respectively for each image is defined as

$$\theta_{LF}(c_i, LF_s, s) = 1 - \mid (\mid \sigma(c_i) - \sum_{s \in x_i} \sigma(LF_s) / \sum_{s \in x_i} 1 \mid) - w_{LF} \mid \tag{13.17}$$

where the small positive integer w_{LF} is set to 0.1 in practice.

Parameters of TF Potentials The learning of parameters of TF Potentials is based on Joint Boost algorithm, and an excellent detailed treatment of the learning process is given in literature (Shotton *et al.*, 2009), but we briefly describe it here for completeness. Each training example s (a pixel in a training image) is paired with a target value $Z_s^c \in \{-1, +1\}$ ($+1$ if the example s has ground truth class c, -1 otherwise) and assigned a weight ω_s^c specifying its classification accuracy for class c after iteration of boosting. Each round of iteration chooses a new weak learner by minimizing an error function incorporating the weights. The training examples are then re-weighted ω_s^c to reflect the new classification accuracy. This procedure emphasizes poorly classified examples in subsequent rounds of iteration, and ensures that over many rounds, the classification for each training example approaches the target value and the parameters are optimal.

Parameters of Other Potentials The parameters of other potentials of MSMSH-CRF model,θ_{PLF}, $\theta_{PTF}, \theta_3, \theta_p$ and θ_l, are selected manually such that the classification error is minimized on the training set.

13.3.9 Model Inference

Given a set of parameters learned for the MSMSH-CRF model, the optimal labeling $\mathbf{c}^*$, which minimizes the energy function in Eq. (13.2), is found by applying the alpha-expansion graph-cut algorithm (Boykov & Jolly, 2001; Boykov *et al.*, 2001).

13.4 Experiments

In this section, experiments are performed on the Beijing Airborne Data (Zhang *et al.*, 2013), to evaluate the performance of the proposed method.

13.4.1 Dataset

We conduct experiments to evaluate the performance of the MSMSH-CRF model on the Beijing Airborne Data (Zhang *et al.*, 2013), which include remote sensing images with a resolution of 0.12m and LiDAR data with a point density of 4 points/m^2, as illustrated in Figure 13.4. The objects in all images correspond to one of three classes: Building, Road and Vegetation. These classes are typical objects appearing in airborne images. In the experiments, we take the ground-truth label of a region to be the majority vote of the ground-truth pixel labels, and randomly divide the images into a training set with 50 images and a testing set with 50 images.

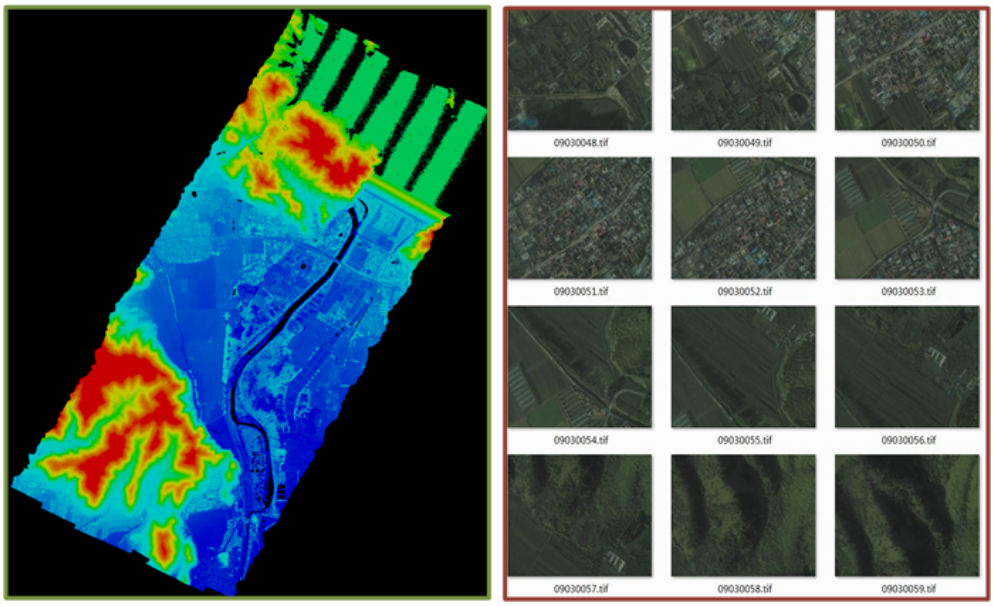


Figure 13.4: The example images of the Beijing Airborne Data (Zhang *et al.*, 2013). *Left*: LiDAR data, *Right*: remote sensing images of the surveying area.

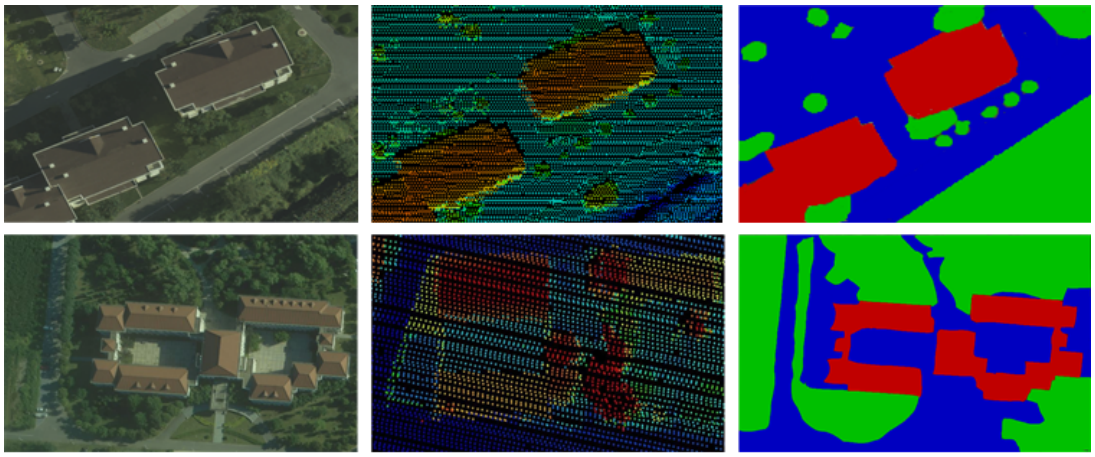


Figure 13.5: The classification result from the MSMSH-CRF model on the Beijing Airborne Data (Zhang *et al.*, 2013). *Left*: remote sensing image, *Middle*: LiDAR point cloud, *Right*: classification result (red - building, blue - road, green - vegetation).

Table 13.1: Average pixelwise accuracy of three methods on the Beijing Airborne Data.

Method	Accuracy (%)
(Shotton *et al.*, 2009)	64.2
(Zhang *et al.*, 2013)	73.6
Ours	83.7

Table 13.2: Pixelwise accuracy of the MSMSH-CRF classification on the Beijing Airborne Data. The confusion matrix shows classification accuracy for each class (rows) and is row-normalized to sum to 100%. Row labels indicate the true class, and column labels indicate the predicted class.

	building	road	vegetation
building	78.3	11.9	9.8
road	9.5	85.9	4.6
vegetation	9.7	8.7	81.6

Table 13.3: The confusion matrix: pixelwise accuracy of the standard CRF classification (Shotton *et al.*, 2009) on the Beijing Airborne Data.

	building	road	vegetation
building	63.7	19.2	17.1
road	22.4	67.0	10.6
vegetation	11.3	15.2	73.5

13.4.2 Results

Figure 13.5 shows the example results of MSMSH-CRF classification method. The average pixelwise accuracy on the testing set is given in Table 13.1. The average classification accuracy of our method is 83.7%, which has 10.1% gain w.r.t. the accuracy of the MSHCRF model (Zhang *et al.*, 2013) and 19.5% gain w.r.t. the accuracy of the standard CRF model (Shotton *et al.*, 2009). The parameter, learned by cross validation on the training set, are $\theta_{PLF} = 0.22, \theta_{PTF} = 0.18, \theta_3 = 0.15, \theta_p = 0.2$, and $\theta_l = 0.25$. For the fairness of comparison, both the training set and the testing set are same for MSMSH-CRF, MSHCRF and standard CRF respectively.

Table 13.2 shows the confusion matrix obtained by applying standard MSMSH-CRF model to the whole test dataset. Accuracy values in the table are computed as the percentage of image pixels assigned to the correct class label, ignoring pixels labeled as void in the ground truth. Compared to the confusion matrices of standard CRF model and MSHCRF model in Table 13.3 and Table 13.4 respectively, the MSMSH-CRF model yields significant improvement on all three classes for integrating multi-scale hierarchical information of the regions in the images. Table 13.5 shows the performance comparison when dropping one types of potentials in the MSMSH-CRF model.

13.5 Conclusion

In conclusion, this work presents a novel multi-source multi-scale hierarchical conditional random field model for automatic classification of remote sensing images. The main contributions of this work are

Table 13.4: The confusion matrix: pixelwise accuracy of the MSHCRF classification (Zhang *et al.*, 2013) on the Beijing Airborne Data.

	building	road	vegetation
building	70.1	15.8	14.1
road	14.4	77.3	8.3
vegetation	12.3	13.8	73.9

Table 13.5: The performance comparison when dropping one types of potentials in the MSMSH-CRF model.

Potentials	Accuracy (%)
With all potentials	83.7
Set $E_2 = 0$ in Eq. (13.2)	63.9
Set $E_3 = 0$ in Eq. (13.2)	73.6
Set $E_4 = 0$ in Eq. (13.2)	70.1

summarized as follows: a novel CRF-based modeling scheme exploiting the complementarity of multi-source data such as the texture in remote sensing images and the elevation in LiDAR data. To exploit different levels of contextual information in images, the multi-scale hierarchical potentials are proposed in our model, which is then enhanced by evidence aggregation from a local to global level. Considering the interrelation of the same objects in remote sensing images and LiDAR data, multi-source hierarchical potentials are proposed in our model to make full use of the category consistency of multi-source data. We have evaluated the precision and robustness of the proposed approach on airborne data, which shows that the proposed method outperforms standard CRF method. However, feature extraction is crucial to the final classification accuracy. Feature selection is done in an ad-hoc fashion in the current stage. In our future work, we are interested in automatic feature selection that may further improve the classification performance.

13. MULTI-SOURCE MULTI-SCALE HIERARCHICAL CONDITIONAL RANDOM FIELD MODEL FOR REMOTE SENSING IMAGE CLASSIFICATION

Chapter 14

Integration of Gaussian Process and Markov Random Field for Hyperspectral Image Classification

In this chapter, we propose a framework GP-MRF, which combines Gaussian processes (GPs) and Markov random field (MRF) for accurate classification of hyperspectral remote sensing image (HSI) data. This method exploits the relationship between adjacent pixels and integrates it into spectral information to obtain spectral-spatial classification. This framework consists of two steps. Firstly, a GP classifier (GPC) yields pixelwise predictive probability for each class. Secondly, an MRF is applied to extract spatial contextual information in the label map achieved in the first step. Then the classification results are inferred from the spectral-spatial information. By means of MRF regularization an enhanced classification result has been obtained. The experiments are performed on three hyperspectral benchmark datasets. The results from the GPC are compared with those obtained by state-of-the-art classification approaches and demonstrate that, GP model is a competitive tool for classification of HSI in terms of accuracy. Furthermore, the experimental results indicate that our proposed method GP-MRF improves the classification accuracy of conventional GPC. This research appears at the IEEE Joint Urban Remote Sensing Event (JURSE) (Liao *et al.*, 2015).

14.1 Introduction

The abundant spectral information contained in hyperpsectral data enable the characterization, identification, and classification of the land-covers with improved accuracy and robustness. However, several critical problems are unavoidable in classification of HSI, among which: 1) a great number of spectral bands and relatively a small number of labeled training samples, which poses the well-known Hughes phenomenon (Hughes, 1968); 2) the spatial variability of the spectral signature; 3) noisy environment; 4) The scene of different objects made by the same or similar material (e.g. the roofs of some buildings and the roads can be made by the same material, asphalt) makes it hard to distinguish different land-covers. Therefore, the contextual information is necessary for classification task of HSI.

In recent years, some state-of-the-art methods have been successfully applied in the remote sensing community to classification task, such as support vector machines (SVMs) (Melgani & Lorenzo, 2004) and random forests (RFs) (Ham *et al.*, 2005). In particular, the kernel-based methods represented by SVMs have been proved as an excellent classification approach for HSI in terms of accuracy and robustness (Camps-Valls & Bruzzone, 2005; Melgani & Lorenzo, 2004). The kernel-based methods have the

inherent virtues: 1) handling high dimensional input spaces efficiently; 2) dealing with noisy samples in a robust way; 3) work with a relatively low number of labeled training samples. These properties make them well-suited to tackle the classification problems of HSI. GPs are another representative of potentially promising kernel-based methods. They have been successfully applied to HSI classification and yielded comparable or even better performance than SVM in terms of accuracy (Zhao *et al.*, 2008). Moreover, they provide truly probabilistic outputs with an explicit degree of prediction uncertainty. In contrast to non-probabilistic approaches, the probabilistic techniques have various advantages in practical recognition circumstances (Kumar, 2005). Furthermore, there exist algorithms for GP hyperparameter learning which are lacking in the SVM framework. Therefore, GP is more likely to yield better classification results. However, Bayesian GP methods have not received much attention in remote sensing community.

In order to alleviate the aforementioned spatial problems, it is necessary to exploit spatial contextual information to enhance the classification accuracy that is only based on spectral information. Markov random fields (MRFs) are effective probabilistic models to integrate spatial correlation of neighbors in a label image into a decision rule (Li, 2009). The maximum a posterior (MAP) decision rule is typically used in this framework (Solberg *et al.*, 1996). In the MRF model, we assume that the class distribution of each pixel depends on a certain degree on its adjacent pixels. This assumption is reasonable because of two practical reasons: 1) adjacent pixels have mixed spectral response on the center pixel, especially the pixels near the borders (spatial boundaries); 2) in a HSI over an urban or suburban region, each land-cover type mostly arises in form of a patch, lump or local region. In mostly pixelwise classification results of HSI we observe that, many scattered pixels are assigned different labels from its adjacent pixels, or a small plot among a big region is classified as another land-cover type. Such classification results are normally susceptible. By means of combination of spectral information with spatial contextual information to construct a new decision rule the classification results can be modify and the accuracy will be clearly enhanced.

In this work, we present a GP- and MRF-based (GP-MRF) method for spectral-spatial classification of HSI. Firstly, a GP model is applied to obtain the label image of HSI and yield predictive probability of each pixel for each class. Secondly, spatial contextual information is extracted by MRF model based on the label map. Finally, the spectral information is integrated into spatial contextual information to construct a new decision rule and each pixel will be reclassified. The second and third steps will be repeated until the the results satisfy a predefined criterion.

This work is outlined as follows. Section 14.2 briefly reviews the formulation of GPC and MRF, then discusses how to combine this two methods. Section 14.3 presents and discusses the experimental results. We conclude the work in Section 14.4.

14.2 GP-MRF Model

14.2.1 GP Model for Classification

Given a training set $(\mathbf{X}, \mathbf{Y}) = \{\mathbf{X}_n, Y_n\}_{n=1}^{N}$, where N is the number of labeled samples and Y_n is the corresponding class label that indicates the land-cover type. Each vector $\mathbf{X}_n \in R^d$ represents the spectral d bands of a pixel in a HSI. Our task is labeling a new test sample set $\mathbf{x} = \{\mathbf{x}_m\}_{m=1}^{M}$, where M is the number of test samples, by computing the probability $P(y|\mathbf{X}, \mathbf{Y}, \mathbf{x})$ belonging to a class. For simple illustrating the binary classification with target $y_i \in \{-1, +1\}$ is considered here. The binary classification is easily extended to multiple classification by using the one-against-all or one-against-one strategy.

GP models generate a discrete label y_i for a data point $\mathbf{x}_i$ via a continuous latent variable f_i (Rasmussen & Williams, 2006). A likelihood model $p(\mathbf{y}|\mathbf{f})$ characterizes the monotonic relationship between latent variable $\mathbf{f}$ and the probably observed annotation $\mathbf{y}$. Several forms of squashing functions

are available for such likelihood model. In particular the logistic and probit function are the most popular. In this work, the probit function is considered.

$$p(y_i = +1|f_i) = \varphi(y_i f_i) \tag{14.1}$$

where φ is the Gaussian cumulative distribution function with the form:

$$\varphi(z) = \int_{-\infty}^{z} \frac{1}{\sqrt{2\pi}} exp(-\frac{x^2}{2})dx \tag{14.2}$$

To make a probability prediction for $\mathbf{x}$ an integrating over the latent variable f is executed as follows:

$$p(y_i = +1|\mathbf{X}, \mathbf{Y}, \mathbf{x}_i) = \int p(y_i|f_i)p(f_i|\mathbf{X}, \mathbf{Y}, \mathbf{x}_i)df_i \tag{14.3}$$

where $p(f_i|\mathbf{X}, \mathbf{Y}, \mathbf{x}_i)$ is the distribution of latent variable f_i corresponding to $\mathbf{x}_i$. It can be obtained by integrating over $\mathbf{F} = (F_1, \ldots, F_n)$, which is the latent variable corresponding to training set $(\mathbf{X}, \mathbf{Y})$:

$$p(f_i|\mathbf{X}, \mathbf{Y}, \mathbf{x}_i) = \int p(f_i|\mathbf{X}, \mathbf{Y}, \mathbf{x}_i, \mathbf{F})p(\mathbf{F}|\mathbf{X}, \mathbf{Y})d\mathbf{F} \tag{14.4}$$

where $p(\mathbf{F}|\mathbf{X}, \mathbf{Y}) = p(\mathbf{F}|\mathbf{Y})p(\mathbf{F}|\mathbf{X})$ / $p(\mathbf{Y}|\mathbf{X})$ is the posterior over the latent variables. $p(\mathbf{Y}|\mathbf{X})$ is the marginal likelihood (evidence), $p(\mathbf{F}|\mathbf{X})$ is the GP prior over the latent function, which in GP model is a jointly zero mean Gaussian distribution and with the covariance given by the kernel K.

The non-Gaussian likelihood in Eq. (14.4) makes the integral analytically intractable. We have to resort to either analytical approximation of integrals or Monte Carlo methods. Two well known analytical approximation methods are suitable for this task, namely *Laplace* (Williams & David, 1998) and *Expectation Propagation* (EP) (Minka, 2001). They both approximate the non-Gaussian joint posterior with a Gaussian one. In this work we adopt the *Laplace* method since its relative lower computation cost than EP with comparable accuracy. As introduced in (Rasmussen & Williams, 2006) the posterior mean and variance for f_i are obtained as follow:

$$\mu_i = \mathbf{k}(\mathbf{x}_i)^T \mathbf{K}^- \widetilde{\mathbf{F}} \tag{14.5}$$

$$\sigma^{\mathbf{2}}{}_i = \mathbf{k}(\mathbf{x}_i, \mathbf{x}_i) - \mathbf{k}(\mathbf{x}_i)^T (\mathbf{K} + \mathbf{W}^-)\mathbf{k}(\mathbf{x}_i) \tag{14.6}$$

where $\mathbf{W} \triangleq -\nabla\nabla logp(\mathbf{Y}|\widetilde{\mathbf{F}})$ is diagonal. $\mathbf{K}$ denotes a $N - by - N$ covariance matrix between N training points. $\mathbf{k}(\mathbf{x}_i)$ is a covariance vector between N training points X and a test points $\mathbf{x}_i$ and $\mathbf{k}(\mathbf{x}_i, \mathbf{x}_i)$ is covariance matrix for test point $\mathbf{x}_i$ and $\widetilde{\mathbf{F}} = \arg\max_{\mathbf{F}} p(\mathbf{F}|\mathbf{X}, \mathbf{Y})$. Given the mean and variance of f_i, we compute the prediction probability in Eq. (14.3).

The covariance function is the crucial ingredient in GP predictor and its hyperparameters Θ crucially affects its performance. The Gaussian radial basis function (RBF) is one of the most widely used kernels since its robustness for different types of data and given as follow:

$$K_{RBF}(\mathbf{x}_i, \mathbf{x}_j) = \sigma^2 exp(-\frac{\| \mathbf{x}_i - \mathbf{x}_j \|^2}{2^2}) \tag{14.7}$$

$\Theta = [\sigma, l]$ is the hyperparameter set for RBF, of which *l* in the function is the characteristic lengthscale, which informally can be roughly considered as the distance you have to move in input space for the function value to become uncorrelated. The smaller *l* we choose, the more rapidly the function varies. In this case, all of the training points are more correctly classified. Moreover, if *l* varies with input di-

Input: $P_L(x_i|y_i)$: the the likelihood function for pixel i belonging to a class y_i;
Im: the label map from GPC;
Output: optimal $y^* \rightarrow$ new label map
1: initial the minimal global energy E_{min};
2: compute spectral energy $E_{spectral}$;
3: find the neighbourhoods $\mathcal{N}$ for each site;
4: **repeat**
5: compute spatial energy $E_{spatial}$ based on Im;
6: compute local energy $E(y_i)$ for each site;
7: assign the new label y_i^* corresponding to $\min E(y_i)$ to the site i and update label map Im;
8: compute the global energy $E(y)$ and compare with E_{min};
9: **if** $E(y) \leq E_{min}$ **then**
10: $E_{min} \leftarrow E(y)$
11: **end if**
12: **until** E_{min} convergence

Figure 14.1: GP-MRF

mensions (i.e. input bands), e.g. $l = [l_1, \ldots, l_d]$, there is another kernel called the Automatic Relevance Determination (ARD) which is derived form RBF:

$$K_{ARD}(\mathbf{x}_i, \mathbf{x}_j) = \sigma^2 exp(-\sum_{b=1\ldots d} \frac{\| x_i^b - x_j^b \|^2}{2_b^2}) \tag{14.8}$$

x_i^b indicates the bth band of the ith input point. The ARD has been proved to be an effective kernel successfully removing irrelevant information (Rasmussen & Williams, 2006; Williams & David, 1998). It provides a parametrization scheme for automatic feature reduction especially for the high-dimensional challenge such as HSI with more than one hundred bands.

14.2.2 MRF-based Regularization

In the aforementioned pixelwise classification, only the spectral information is considered. However the spectral response can be affected by other spectrum from adjacent pixels. Therefore, it is necessary to regularize the pixelwise classification results with MAP-MRF framework (Geman & Geman, 1984).

Markov Random Fields are a probabilistic framework that incorporate the spatial information from a set of cliques in images, whose basic principle is that each pixel interacts only with its neighboring pixels (Li, 2009). In other words, a pixel more possibly has the same label as its neighborhoods. Because of formulating MRF models within Bayesian framework, the optimal solution is the *Maximum a Posteriori* (MAP) and is obtained by maximizing the posterior probability $Pr(\mathbf{y}|\mathbf{x})$:

$$\mathbf{y}^* = \arg\max_{\mathbf{y} \in \mathcal{Y}} Pr(\mathbf{y}|\mathbf{x}) \tag{14.9}$$

where $\mathbf{x}$ is the observation and $\mathbf{y}$ is the possible labeling.

Based on the *Hammersley-Clifford theorem* (Hammersley & Clifford, 1971), we consider the MAP

solution as the minimization of an energy cost function (Yang & Förstner, 2011a):

$$E = E_{spectral} + E_{spatial} \tag{14.10}$$

$E_{spectral}$ is the spectral energy defined by the likelihood function as:

$$E_{spectral} = -ln\{P_L(x_i|y_i)\} \tag{14.11}$$

where x_i is the site of the ith pixel in the label map, y_i is one of the possible label for site x_i, and the likelihood function $P_L(x_i|y_i)$ have been already yielded by GPC (i.e. $P(y_i|\mathbf{x}_i)$), which means the predictive probability of x_i belonging to the class y_i. The second term of Eq. (14.10) is spatial energy and its standard expression is:

$$E_{spatial} = \sum_{j \in \mathcal{N}} \beta(1 - \delta(y_i, y_j)), j \in \mathcal{N} \tag{14.12}$$

where $\delta(.,.)$ is the Kronecker delta function ($\delta(a,b) = 1$ if $a = b$, else $\delta(a,b) = 0$) and β is a non-negative parameter controlling the weight of spatial energy. $\mathcal{N}$ is the neighborhood system, which in this work is 8-connected. y_i is the label of the center pixel x_i and y_j is the label of its jth neighboring pixel.

We adopt the *Iterative Conditional Modes* (ICM) (Prince, 2012) to solve the optimization problem. We compute the local energy $E(x_i)$ of each pixel belonging to each label. A pixel is assigned to the label with smallest energy and it gets the local optimization. The local energies were summed up as global energy. Based on the updated label map the above procedure will be repeated. The optimization can be achieved until the global energy is convergence. The procedure is detailed in Figure 14.1.

14.3 Experimental Results

In the experiments, three hyperspectral datasets-INDIAN PINES, UNIVERSITY OF PAVIA, and CENTER OF PAVIA will be used in this work. These datasets have been widely used as benchmark (Melgani & Lorenzo, 2004; Zhao *et al.*, 2008) in the study of HSI classification. The INDIAN PINES data set was acquired by the AVIRIS in 1992 and taken over a predominately agricultural region in NW Indiana, USA. The data set has 145×145 pixels and 200 channels. Seven of the 16 different land-cover classes in the original ground-truth were removed, which can offer only a few training samples (this makes the experimental analysis more significant from the statistical viewpoint) (Melgani & Lorenzo, 2004). The CENTER OF PAVIA image remains 102 channels after removing some noisy bands and lies around the center of Pavia with 1096×492 pixels. The ground-truth consists of 9 land-cover classes. The UNIVERSITY OF PAVIA data set has 103 channels with 610×340 pixels and also 9 land-cover classes.

In the experiments, both the RBF and ARD kernel were adopted in the GP model for comparison purpose and the hyperperameters were optimized by *Conjugate Gradient* method (Nocedal & Wright, 2006) based on the *Laplace* method. In order to simplify the classification and balanced samples problems, the one-against-one strategy was applied. The algorithm (Wu *et al.*, 2004) was used to estimate the predictive probability of the test samples belonging to each class from the results of one-against-one strategy.

The original image and ground truth of Indian Pines dataset are shown in Figure 14.2(a) and Figure 14.2(b) respectively. The classification results of GPC are shown in Figure 14.2(c). Many scattered pixels or small patches are labeled as different classes from their adjacent pixels by GPC. These labels are unconvinced as we have discussed in Section I. Figure 14.2(d) shows the improved classification

Table 14.1: Individual class percentage accuracies of the Indian Pines data set with different classifiers.

Class	SVM	RF	GP_{RBF}	GP_{ARD}	GP-MRF
C1 Corn-notill	85.67	80.33	88.38	90.61	**93.81**
C2 Corn-mintill	87.47	69.74	83.85	90.26	**99.05**
C3 Grass-pasture	92.92	92.95	97.36	96.48	**97.53**
C4 Grass-Trees	98.88	98.25	99.13	98.83	**99.81**
C5 Hay-windrowed	99.60	**100**	**100**	**100**	**100**
C6 Soybean-notill	85.59	84.87	86.84	89.25	**97.80**
C7 Soybean-mintill	89.04	72.66	88.21	**92.54**	90.47
C8 Soybean-clean till	83.82	91.24	92.81	94.60	**98.47**
C9 Woods	99.37	99.15	99.50	99.16	**99.91**

Table 14.2: OA and AA in percentage of GP (RBF), GP (ARD) and GP-MRF (ARD) for different datasets.

Algorithm	INDIAN		UNIVERSITY		CENTER	
	OA	AA	OA	AA	OA	AA
GP (RBF)	84.50	89.39	90.09	92.35	98.33	96.53
GP (ARD)	87.26	91.41	89.82	92.11	98.41	96.60
GP-MRF (ARD)	95.60	97.42	96.9	97.53	97.48	99.13

results by MRF based on the results of GPC. The label image is refined by MRF. In this experiment, the RBF kernel was used in GP model. 200 points for each class from these datasets were randomly selected as training samples used for learning the classifiers and the residual were test samples for assessing their performance. Figure 14.3 shows the classification results of the University of Pavia dataset.

Table 14.1 shows the individual class accuracy of SVM, RF, GP (RBF), GP (ARD) and GP-MRF (ARD) from the Indian Pines data set. In order to objectively compare the performances between different classifiers, we used the same size of training and test samples as (Camps-Valls & Bruzzone, 2005) and quoted the experimental results of SVM (RBF). The results show that the GPC performs competitively or even better than the state-of-the-art methods SVM and RF in terms of accuracy. The comparison between the GPC (RBF) and GPC (ARD) proves the previous discussion in Section 14.2: the ARD kernel outperforms RBF kernel for classification of HSI. However, in order to optimize more parameters for ARD kernel, more input dimensions increase the training time rapidly. Finally, the results of GP-MRF (ARD) demonstrate that our proposed approach can significantly increase the classification accuracy of the individual class.

Table 14.2 compares the results in terms of overall accuracy (OA) and average accuracy (AA) between GP (RBF), GP (ARD) and GP-MRF (ARD) in three different datasets. The results further prove that, our proposed approach can effectively improve the accuracies of classification for HSI over urban/suburban regions. 200 points for each class from these datasets were randomly selected as training samples and the residual were regarded as test samples.

Finally, Figure 14.4 investigates the performances of GP-MRF (ARD) in terms of global classification accuracy with different weight parameter $\beta = [0.5, 1, 2, 3, 4, 5]$ for spatial information in Eq. (14.12). We draw the conclusion that the OA is not significantly different over the given values. Our method is robust to the choice of β.

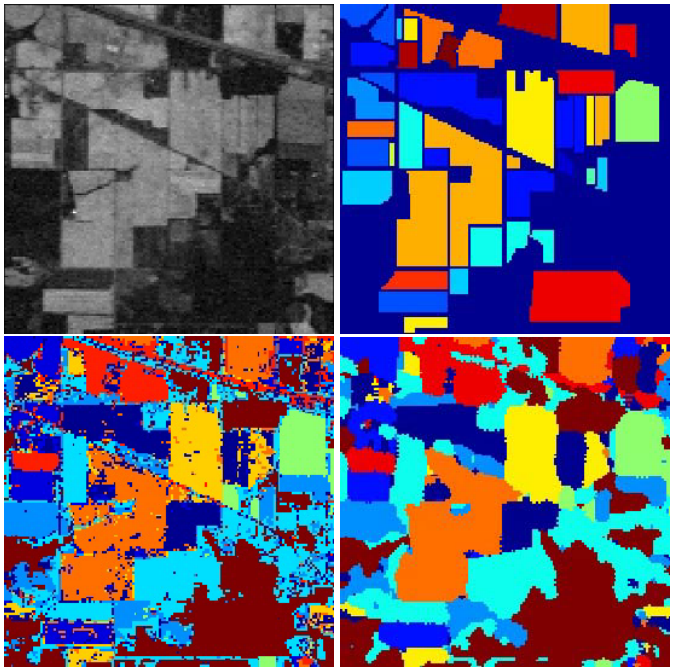


Figure 14.2: Hyperspectral image classification results of Indian Pines dataset. From Left to Right, from Top to Bottom: (a) Data of Indian Pines, (b) ground truth, (c) classification result of GPC (ARD) and (d) classification result of GP-MRF (ARD).

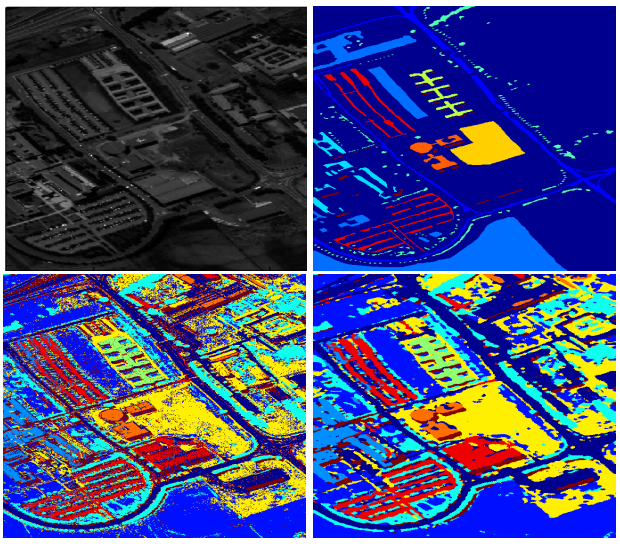


Figure 14.3: Hyperspectral image classification results of the University of Pavia dataset. From Left to Right, from Top to Bottom: (a) Data of the University of Pavia, (b) ground truth, (c) classification result of GPC (ARD) and (d) classification result of GP-MRF (ARD).

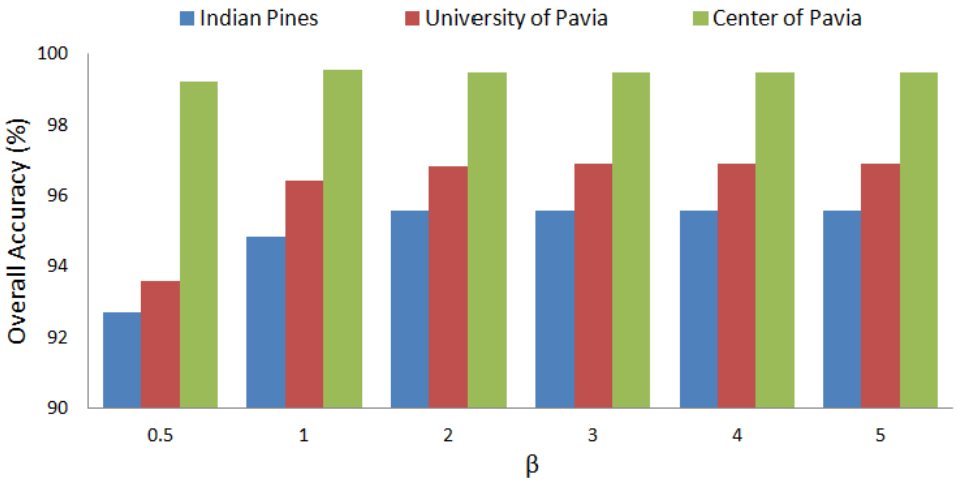


Figure 14.4: Overall accuracy in percentage for different values of β for different datasets.

14.4 Conclusion

In this work we proposed a novel framework GP-MRF, which combines the GPC and MRF to enhance the classification accuracies. The GP-MRF framework integrates the spectral information into spatial information and effectively classifies the HSI over urban/suburban regions without selection or reduction of data dimensionality.

We evaluated the performance of GP-MRF in three hyperspectral datasets and the results demonstrated that MRFs utilize the relationship between the adjacent pixels to improve the classification accuracy of HSI on the basis of GPC classification. We used GPC to preliminarily classify original data and obtain label image and predictive probability of each pixel belonging to each class which will be applied in the step of MRF. The experiment shows that our approach yields accurate classification results and is robust for classifying different kinds of HSI.

Part IV

APPENDIX

Bibliography

Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., & Süsstrunk, S. 2012. SLIC Superpixels Compared to State-of-the-Art Superpixel Methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **34**(11), 2274–2282. 36, 40

Aharon, M., Elad, M., & Bruckstein, A. 2006. K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation. *IEEE Transactions on Signal Processing*, **54**(11), 4311–4322. 145, 153

Al-Shaikhli, Saif, Yang, Michael Ying, & Rosenhahn, Bodo. 2014a. Brain Tumor Classification Using Sparse Coding and Dictionary Learning. *In: IEEE International Conference on Image Processing (ICIP)*. 12, 27, 141

Al-Shaikhli, Saif, Yang, Michael, & Rosenhahn, Bodo. 2014b. Coupled Dictionary Learning for Multi-Label Brain Tumor Segmentation in Flair MRI images. *In: International Symposium on Visual Computing (ISVC)*. 13, 27, 149

Al-Shaikhli, Saif, Yang, Michael Ying, & Rosenhahn, Bodo. 2014c. Multi-Region Labeling and Segmentation Using a Graph Topology Prior and Atlas Information in Brain Images. *Computerized Medical Imaging and Graphics Journal*, **38**(8), 725–734. 11, 26, 123, 143, 144

Alexe, Bogdan, Deselaers, Thomas, & Ferrari, Vittorio. 2012. Measuring the Objectness of Image Windows. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **34**(11), 2189–2202. 101

Aljabar, Paul, Heckemann, Rolf A, Hammers, Alexander, Hajnal, Joseph V, & Rueckert, Daniel. 2009. Multi-atlas based Segmentation of Brain Images: Atlas Selection and its Effect on Accuracy. *NeuroImage*, **46**(3), 726–738. 124, 135, 136, 138, 140

Andrews, Shawn, McIntosh, Chris, & Hamarneh, Ghassan. 2011. Convex Multi-region Probabilistic Segmentation with Shape Prior in the Isometric Log-ratio Transformation Space. *Pages 2096–2103 of: IEEE International Conference on Computer Vision (ICCV)*. 123, 124

Andrieu, C., de Freitas, N., Doucet, A., & Jordan, M. I. 2003. An Introduction to MCMC for Machine Learning. *Machine Learning*, **50**(1-2), 5–43. 106, 107, 108, 109

Arbelaez, P., Maire, M., Fowlkes, C., & Malik, J. 2011. Contour Detection and Hierarchical Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **33**(5), 898–916. 3, 19, 36, 39, 41, 92

Babenko, Boris, Yang, Ming-Hsuan, & Belongie, Serge. 2011. Visual Tracking with Online Multiple Instance Learning. *Pages 983–990 of: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 115

BIBLIOGRAPHY

Baddeley, A., & van Lieshout, M. 1993. Stochastic Geometry Models in High-level Vision. *Statistics and Images*, 231–256. 164

Bagon, S., Boiman, O., & Irani, M. 2008. What Is a Good Image Segment? A Unified Approach to Segment Extraction. *Pages 30–44 of: European Conference on Computer Vision (ECCV)*. 37

Baltsavias, E.P. 2004. Object Extraction and Revision by Image Analysis Using Existing Geodata and Knowledge: Current Status and Steps Towards Operational Systems. *ISPRS Journal of Photogrammetry and Remote Sensing*, **58**(3-4), 129–151. 164

Barth, A., Siegemund, J., Meissner, A., Franke, U., & Förstner, W. 2010. Probabilistic Multi-Class Scene Flow Segmentation for Traffic Scenes. *Pages 503–512 of: Annual Symposium of the German Association for Pattern Recognition (DAGM)*. 50

Batra, Dhruv, Kowdle, Adarsh, Parikh, Devi, Luo, Jiebo, & Chen, Tsuhan. 2011. Interactively Co-segmentating Topically Related Images with Intelligent Scribble Guidance. *International Journal of Computer Vision*, **93**(3), 273–292. 95

Bauer, Stefan, Nolte, Lutz-P, & Reyes, Mauricio. 2011. Fully Automatic Segmentation of Brain Tumor Images Using Support Vector Machine Classification in Combination with Hierarchical Conditional Random Field Regularization. *Pages 354–361 of: Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. 141

Bazin, Pierre-Louis, & Pham, Dzung. 2008. Homeomorphic Brain Image Segmentation with Topological and Statistical Atlases. *Medical Image Analysis*, **12**(5), 616–625. 124

Benedek, Csaba, Descombes, Xavier, & Zerubia, Josiane. 2010. Building Detection in a Single Remotely Sensed Image with a Point Process of Rectangles. *Pages 1417–1420 of: International Conference on Pattern Recognition (ICPR)*. 167

Berger, Cyrille. 2012. Toward Rich Geometric Map for SLAM: Online Detection of Planes in 2D LIDAR. *In: Proceedings of the International Workshop on Perception for Mobile Robots Autonomy (PEMRA)*. 181

Besag, J. 1974. Spatial Interaction and the Statistical Analysis of Lattice Systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, 192–236. 2, 7, 18, 22, 80, 164

Besag, J. 1986. On the Statistical Analysis of Dirty Pictures (with discussion). *Journal of the Royal Statistical Society Series B*, **48**(3), 259–302. 2, 18, 19, 176

Besse, Frederic, Rother, Carsten, Fitzgibbon, Andrew, & Kautz, Jan. 2012. PMBP: PatchMatch Belief Propagation for Correspondence Field Estimation. *In: British Machine Vision Conference (BMVC)*. 105, 107

Bishop, Christopher. 2006. *Pattern Recognition and Machine Learning*. Springer. 176

Bleyer, Michael, Rother, Carsten, Kohli, Pushmeet, Scharstein, Daniel, & Sinha, Sudipta. 2011. Object Stereo - Joint Stereo Matching and Object Segmentation. *Pages 3081–3088 of: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 70

Botev, Z, Grotowski, J, & Kroese, D. 2010. Kernel Density Estimation via Diffusion. *Annals of Statistics*, **38**(5), 2916–2957. 97

Boyd, Stephen, Parikh, Neal, Chu, Eric, Peleato, Borja, & Eckstein, Jonathan. 2011. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Foundations and Trends in Machine Learning*, **3**(1), 1–122. 73

Boykov, Yuri, & Jolly, Marie-Pierre. 2001. Interactive Graph Cuts for Optimal Boundary and Region Segmentation of Objects in N-D Images. *Pages 105–112 of: IEEE International Conference on Computer Vision (ICCV)*. 95, 182, 184

Boykov, Yuri, & Kolmogorov, Vladimir. 2004. An Experimental Comparison of Min-Cut/Max-Flow Algorithms for Energy Minimization in Vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **26**, 1124–1137. 96, 97

Boykov, Yuri, Veksler, Olga, & Zabih, Ramin. 2001. Fast Approximate Energy Minimization via Graph Cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **23**, 1222–1239. 9, 19, 25, 72, 73, 93, 96, 97, 103, 104, 164, 171, 184

Brendel, William, & Todorovic, Sinisa. 2009. Video Object Segmentation by Tracking Regions. *Pages 833–840 of: IEEE International Conference on Computer Vision (ICCV)*. 92

Brox, T., & Malik, J. 2010. Object Segmentation by Long Term Analysis of Point Trajectories. *Pages 282–295 of: European Conference on Computer Vision (ECCV)*. 8, 24, 90, 92, 94, 96, 98, 99, 100, 101

Brox, T., & Malik, J. 2011. Large Displacement Optical Flow: Descriptor Matching in Variational Motion Estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **33**(3), 500–513. 93, 98

Camps-Valls, G., & Bruzzone, L. 2005. Kernel-based Methods for Hyperspectral Image Classification. *IEEE Transactions on Geoscience and Remote Sensing*, **43**(6), 1351–1362. 15, 30, 189, 194

Cao, T, Jojic, V., Modla, S., Powell, D., Czymmek, K., & Niethammer, M. 2013. Robust Multimodal Dictionary Learning. *Pages 259–266 of: Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. 150

Chai, Dengfeng, Förstner, Wolfgang, & Yang, Michael Ying. 2012. Combine Markov Random Fields and Marked Point Processes to Extract Building from Remotely Sensed Images. *Pages 365–370 of: ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences; ISPRS Congress*. 14, 29, 163

Chan, Tony, & Vese, Luminita. 2001. Active Contours without Edges. *IEEE Transactions on Image Processing*, **10**(2), 266–277. 124, 136, 138, 140

Chen, Liang-Chieh, Fidler, Sanja, Yuille, Alan L, & Urtasun, Raquel. 2014. Beat the MTurkers: Automatic Image Labeling from Weak 3D Supervision. *In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2. 5, 21, 70

Chockalingam, Prakash, Pradeep, S. Nalin, & Birchfield, Stan. 2009. Adaptive Fragments-based Tracking of Non-rigid Objects Using Level Sets. *Pages 1530–1537 of: IEEE International Conference on Computer Vision (ICCV)*. 91, 98, 99

Cocosco, Chris A, Kollokian, Vasken, Kwan, Remi K-S, Pike, G Bruce, & Evans, Alan C. 1997. BrainWeb: Online Interface to a 3D MRI Simulated Brain Database. *Page 425 of: NeuroImage*, vol. 5. 130, 134, 135, 136, 137, 138, 139, 140, 147

Cocosco, Chris A, Zijdenbos, Alex P, & Evans, Alan C. 2003. A Fully Automatic and Robust Brain MRI Tissue Classification Method. *Medical Image Analysis*, **7**(4), 513–527. 124, 142

Comaniciu, Dorin, & Meer, Peter. 2002. Mean Shift: A Robust Approach Toward Feature Space Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **24**(5), 603–619. 19, 36, 41, 85, 181

Cordier, Nicolas, Menze, Bjoern, Delingette, Hervé, Ayache, Nicholas, *et al.* 2013. Patch-based Segmentation of Brain Tissues. *Pages 6–17 of: MICCAI Challenge on Multimodal Brain Tumor Segmentation*. 150, 157, 158, 159

Criminisia, A., Zisserman, A., Van Gool, L., Bramble, S., & Compton, D. 1999. New Approach to Obtain Height Measurements from Video. *In: Proceedings of SPIE - The International Society for Optical Engineering*, vol. 3576. 47

Dalponte, Michele, Bruzzone, Lorenzo, & Gianelle, Damiano. 2012. Tree Species Classification in the Southern Alps Based on the Fusion of Very High Geometrical Resolution Multispectral / Hyperspectral Images and LiDAR Data. *Remote Sensing of Environment*, **123**, 258–270. 175

Debard, G., Karsmakers, P., Deschodt, M., Vlaeyen, E., Dejaeger, E., Milisen, K., Goedem, T., Vanrumste, B., & Tuytelaars, T. 2012. Camera-Based Fall Detection on Real World Data. *Pages 356–375 of: Outdoor and Large-Scale Real-World Scene Analysis*. 43

Delong, Andrew, Osokin, Anton, Isack, Hossam N, & Boykov, Yuri. 2012. Fast Approximate Energy Minimization with Label Costs. *International Journal of Computer Vision*, **96**(1), 1–27. 53, 155

Deng, F., Li, S., & Su, G. 2012. Classification of Remote Sensing Optical and LiDAR Data Using Extended Attribute Profiles. *IEEE Journal of Selected Topics in Signal Processing*, **6**(7), 856–865. 175

Diebel, James, & Thrun, Sebastian. 2005. An Application of Markov Random Fields to Range Sensing. *Pages 291–298 of: Advances in Neural Information Processing Systems (NIPS)*. 5, 21, 70, 72, 73

Dragon, Ralf, Rosenhahn, Bodo, & Ostermann, Jörn. 2012. Multi-scale Clustering of Frame-to-Frame Correspondences for Motion Segmentation. *Pages 445–458 of: European Conference on Computer Vision (ECCV)*. 92

Drauschke, M., & Förstner, W. 2011. A Bayesian Approach for Scene Interpretation with Integrated Hierarchical Structure. *Pages 1–10 of: Annual Symposium of the German Association for Pattern Recognition (DAGM)*. 82, 84, 85

Duan, Genquan, Ai, Haizhou, Cao, Song, & Lao, Shihong. 2012. Group Tracking: Exploring Mutual Relations for Multiple Object Tracking. *Pages 129–143 of: European Conference on Computer Vision (ECCV)*. 105, 115

Duarte-Carvajalino, J. M., & Sapiro, G. 2009. Learning to Sense Sparse Signals: Simultaneous Sensing Matrix and Sparsifying Dictionary Optimization. *IEEE Transactions on Image Processing*, **18**(7), 1395–1408. 142

Duda, Richard O., & Hart, Peter E. 1972. Use of the Hough Transformation to Detect Lines and Curves in Pictures. *Communications of the ACM*, **15**(1), 11–15. 47

Edgcomb, A., & Vahid, F. 2012. *Estimating Daily Energy Expenditure from Video for Assistive Monitoring, Technical report*. `http://www.static.cs.ucr.edu/store/techreports/UCR-CS-2012-09260.pdf`. 54

Egenhofer, Max J, & Herring, John. 1990. Categorizing Binary Topological Relations Between Regions, Lines, and Points in Geographic Databases. *Technical Report, Department of Surveying Engineering, University of Maine*. 125

Ellis, Liam, & Zografos, Vasileios. 2012. Online Learning for Fast Segmentation of Moving Objects. *In: Asian Conference on Computer Vision (ACCV)*. 97

Endres, I., & Hoiem, D. 2010. Category Independent Object Proposals. *Pages 575–588 of: European Conference on Computer Vision (ECCV)*. 36

Felzenszwalb, P., & Huttenlocher, D. 2004a. Efficient Belief Propagation for Early Vision. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 261–268. 164

Felzenszwalb, Pedro F., & Huttenlocher, Daniel P. 2004b. Efficient Graph-Based Image Segmentation. *International Journal of Computer Vision*, **59**(2), 167–181. 3, 19, 36, 50

Festa, J., Pereira, S., Mariz, J., Sousa, N., & Silva, C. 2013. Automatic Brain Tumor Segmentation of Multi-Sequence MR Images Using Random Decision Forests. *Pages 23–26 of: NCI-MICCAI BRATS*. 150, 157, 158, 159

Fischl, Bruce, Salat, David H, Busa, Evelina, Albert, Marilyn, Dieterich, Megan, Haselgrove, Christian, van der Kouwe, Andre, Killiany, Ron, Kennedy, David, Klaveness, Shuna, *et al.* 2002. Whole Brain Segmentation: Automated Labeling of Neuroanatomical Structures in the Human Brain. *Neuron*, **33**(3), 341–355. 124

Fragkiadaki, Katerina, Zhang, Weiyu, Zhang, Geng, & Shi, Jianbo. 2012a. Two-Granularity Tracking: Mediating Trajectory and Detection Graphs for Tracking under Occlusions. *Pages 552–565 of: European Conference on Computer Vision (ECCV)*. 92

Fragkiadaki, Katerina, Zhang, Geng, & Shi, Jianbo. 2012b. Video Segmentation by Tracing Discontinuities in a Trajectory Embedding. *Pages 1846–1853 of: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 92

Fulkerson, B., Vedaldi, A., & Soatto, S. 2009. Class Segmentation and Object Localization with Superpixel Neighborhoods. *Pages 670–677 of: IEEE International Conference on Computer Vision (ICCV)*. 84

Geiger, Andreas, Lenz, Philip, & Urtasun, Raquel. 2012. Are We Ready for Autonomous Driving? The KITTI Vision Benchmark Suite. *Pages 3354–3361 of: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 74

Geman, Stuart, & Geman, Donald. 1984. Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **6**, 721–741. 2, 18, 164, 192

Georgescu, Bogdan, Shimshoni, Ilan, & Meer, Peter. 2003. Mean Shift Based Clustering in High Dimensions: A Texture Classification Example. *Pages 456–463 of: IEEE International Conference on Computer Vision (ICCV)*. 181

Gladis Pushpa Rathi, V., & Palani, S. 2012. Linear Discriminant Analysis For Brain Tumor Classification Using Feature Selection. *International Journal of Communications and Engineering*, **5**(5), 130–134. 141

Golub, Gene H., & Loan, Charles F. Van. 1996. *Matrix Computations*. 3rd edn. The Johns Hopkins University Press. 38

Gong, Xiaojin, Ren, Jianqiang, Lai, Baisheng, Yan, Chaohua, & Qian, Hui. 2014. Guided Depth Upsampling via A Cosparse Analysis Model. *Pages 738–745 of: IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 5, 21, 70, 75, 76

Gooya, A., Pohl, K., Bilello, M., Biros, G., & Davatzikos, C. 2011. Joint Segmentation and Deformable Registration of Brain Scans Guided by a Tumor Growth Model. *Pages 532–540 of: Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. 150

Gooya, A., Pohl, K., Bilello, M., Cirillo, L., Biros, G., Melhem, E., & Davatzikos, C. 2012. GLISTR: Glioma Image Segmentation and Registration. *IEEE Transactions on Medical Imaging*, **31**(10), 1941–1954. 149, 150

Gould, Stephen, Rodgers, Jim, Cohen, David, Elidan, Gal, & Koller, Daphne. 2008. Multi-Class Segmentation with Relative Location Prior. *International Journal of Computer Vision*, **80**(3), 300–316. 61, 84

Grabner, H., Gall, J., & Van Gool, L. J. 2011. What Makes a Chair a Chair? *Pages 1529–1536 of: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 45, 46

Green, Peter. 1995. Reversible Jump Markov Chain Monte Carlo Computation and Bayesian Model Determination. *Biometrika*, **82**(4), 711–732. 165

Greenspan, Hayit, Goldberger, Jacob, & Mayer, Arnaldo. 2004. Probabilistic Space-Time Video Modeling via Piecewise GMM. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **26**(3), 384–396. 95

Greig, D., Porteous, B., & Seheult, A. 1989. Exact Maximum A Posteriori Estimation for Binary Images. *Journal of the Royal Statistical Society. Series B (Methodological)*, 271–279. 171

Grompone, R., Jakubowics, J., Morel, J., & Randall, G. 2010. LSD: A Fast Line Segment Detector with a False Detection Control. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **32**(4), 722–732. 180

Grompone von Gioi, R., Jakubowicz, J., Morel, J., & Randall, G. 2012. LSD: a Line Segment Detector. *Image Processing On Line*, **2**, 35–55. 48, 49

Grundmann, Matthias, Kwatra, Vivek, Han, Mei, & Essa, Irfan A. 2010. Efficient Hierarchical Graph-based Video Segmentation. *Pages 2141–2148 of: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 8, 23, 89, 92

Guillemaut, Jean-Yves, & Hilton, Adrian. 2011. Joint Multi-layer Segmentation and Reconstruction for Free-viewpoint Video Applications. *International Journal of Computer Vision*, **93**(1), 73–100. 70, 72

Guimond, Alexandre, Meunier, Jean, & Thirion, Jean-Philippe. 2000. Average Brain Models: A Convergence Study. *Computer Vision and Image Understanding*, **77**(2), 192–210. 130

Gupta, A., Satkin, S., Efros, A. A., & Hebert, M. 2011. From 3D Scene Geometry to Human Workspace. *Pages 1961–1968 of: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 45, 46

Ham, Jisoo, Chen, Yangchi, Crawford, Melba M, & Ghosh, Joydeep. 2005. Investigation of the random forest framework for classification of hyperspectral data. *IEEE Transactions on Geoscience and Remote Sensing*, **43**(3), 492–501. 189

Hammersley, J. M., & Clifford, P. 1971. Markov Field on Finite and Lattices. *Unpublished.* 164, 192

Han, Ju, Chang, Hang, Loss, Leandro, Zhang, Kai, Baehner, Frederick L, Gray, Joe W, Spellman, Paul, & Parvin, Bahram. 2011. Comparison of sparse coding and kernel methods for histopathological classification of gliobastoma multiforme. *Pages 711–714 of: IEEE International Symposium on Biomedical Imaging: From Nano to Macro (ISBI)*. 142, 147, 148

Haralick, Robert M, Shanmugam, Karthikeyan, & Dinstein, Its' Hak. 1973. Textural features for image classification. *IEEE Transactions on Systems, Man and Cybernetics*, **3**(6), 610–621. 144

Harrison, Alastair, & Newman, Paul. 2010. Image and sparse laser fusion for dense scene reconstruction. *Pages 219–228 of: Field and Service Robotics*. 75, 76

He, X., Zemel, R., & Carreira-perpinan, M. 2004. Multiscale Conditional Random Fields for Image Labeling. *Pages 695–702 of: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 84

Hedau, Varsha, Hoiem, Derek, & Forsyth, David A. 2009. Recovering the spatial layout of cluttered rooms. *Pages 1849–1856 of: IEEE International Conference on Computer Vision (ICCV)*. 4, 20, 44, 45, 53, 55, 56, 58, 60, 61, 62, 67

Hedau, Varsha, Hoiem, Derek, & Forsyth, David A. 2010. Thinking Inside the Box: Using Appearance Models and Context Based on Room Geometry. *Pages 224–237 of: European Conference on Computer Vision (ECCV)*. 45

Hoiem, Derek, Efros, Alexei A., & Hebert, Martial. 2007. Recovering Surface Layout from an Image. *International Journal of Computer Vision*, **75**(1), 151–172. 45

Hu, W., Xie, D., & Tan, T. 2004. A Hierarchical Self-organizing Approach for Learning the Patterns of Motion Trajectories. *IEEE Transactions on Neural Networks*, **15**(1), 135–144. 44

Huang, Wenqi, Gong, Xiaojin, & Xiang, Zhiyu. 2014. Road scene segmentation via fusing camera and lidar data. *Pages 1008–1013 of: IEEE International Conference on Robotics and Automation (ICRA)*. 5, 21, 70, 71, 74

Huang, Wenqi, Gong, Xiaojin, & Yang, Michael Ying. 2015. Joint Object Segmentation and Depth Upsampling. *IEEE Signal Processing Letters*, **22**(2), 192–196. 2, 6, 18, 21, 69

Huang, Xin, Zhang, Liangpei, & Gong, Wei. 2011. Information Fusion of Aerial Images and LIDAR Data in Urban Areas: Vector-stacking, Re-classification and Post-processing Approaches. *International Journal of Remote Sensing*, **32**(1), 69–84. 175

Hughes, G. 1968. On the mean accuracy of statistical pattern recognizers. *IEEE Transactions on Information Theory*, **14**(1), 55–63. 189

Hwang, Jenq-Neng, Lay, Shyh-Rong, & Lippman, A. 1994. Nonparametric Multivariate Density Estimation: A Comparative Study. *IEEE Transactions on Signal Processing*, **42**(10), 2795–2810. 97

Ihler, Alexander, & McAllester, David. 2009. Particle Belief Propagation. *Pages 256–263 of: International Conference on Artificial Intelligence and Statistics (AISTATS)*. 9, 25, 103, 104

Jia, Zhaoyin, Gallagher, Andrew, Saxena, Ashutosh, & Chen, Tsuhan. 2013. 3D-Based Reasoning with Blocks, Support, and Stability. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 4, 20, 43

Jiang, S., Wu, Y., Huang, M., Yang, W., Chen, W., & Feng, Q. 2013. 3D brain tumor segmentation in multimodal MR images based on learning population-and patient-specific feature sets. *Computerized Medical Imaging and Graphics*, **37**(7), 512–521. 149

Jiang, Yun, Lim, Marcus, & Saxena, Ashutosh. 2012. Learning Object Arrangements in 3D Scenes using Human Context. *In: International Conference on Machine Learning (ICML)*. 43

Johnson, K.A., & Becker, J.A. 1995. *Whole brain atlas database*. `http://www.med.harvard.edu/aanlib/home.html`. 147, 155, 156, 158, 159

Junejo, I., & Foroosh, H. 2006. Robust Auto-Calibration from Pedestrians. *Pages 92–97 of: IEEE International Conference on Video and Signal Based Surveillance (AVSS)*. 47

Kang, Min-Koo, Kim, Daeyoung, & Yoon, Kuk-Jin. 2014. Adaptive Support of Spatial-Temporal Neighbors for Depth Map Sequence Up-sampling. *IEEE Signal Processing Letters*, **21**(2), 150–154. 70

Kapur, Tina, Grimson, W Eric L, Wells III, William M, & Kikinis, Ron. 1996. Segmentation of brain tissue from magnetic resonance images. *Medical Image Analysis*, **1**(2), 109–127. 124

Kaus, M., Warfield, S.K., Nabavi, A., Black, P.M., Jolesz, F.A., & Kikinis, R. 2001. Automated segmentation of MR images of brain tumors. *Radiology*, **218**(2), 586–591. 147

Khalatbari, Kimia. 1999. *MedPix Medical Image Database*. 134, 135, 136, 137, 138, 139, 140

Klein, Allison, Sloan, Peter-Pike, Finkelstein, Adam, & Cohen, Michael. 2002. Stylized video cubes. *Pages 15–22 of: SIGGRAPH/Eurographics Symposium on Computer Animation*. 92

Kohlberger, Timo, Sofka, Michal, Zhang, Jingdan, Birkbeck, Neil, Wetzl, Jens, Kaftan, Jens, Declerck, Jérôme, & Zhou, S Kevin. 2011. Automatic multi-organ segmentation using learning-based segmentation and level set optimization. *Pages 338–345 of: Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. 124

Kolmogorov, Vladimir. 2006. Convergent Tree-Reweighted Message Passing for Energy Minimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **28**, 1568–1583. 9, 19, 25, 103, 104

Kolmogorov, Vladimir, & Rother, Carsten. 2007. Minimizing Nonsubmodular Functions with Graph Cuts-A Review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **29**, 1274–1279. 19

Kong, Deming, Xu, Lijun, Li, Xiaolu, & Xing, Weiwei. 2012. Estimation of Cluster Centers on Building Roof from LiDAR Footprints. *Pages 254–258 of: IEEE International Conference on Imaging Systems and Techniques*. 181

Koppula, H. S., Anand, A., Joachims, T., & Saxena, A. 2011. Semantic Labeling of 3D Point Clouds for Indoor Scenes. *Pages 244–252 of: Advances in Neural Information Processing Systems (NIPS)*. 45

Korč, Filip, & Förstner, Wolfgang. 2009. eTRIMS Image Database for interpreting images of man-made scenes. *In: TR-IGG-P-2009-01, Department of Photogrammetry, University of Bonn*. 85, 86, 88

Kothapa, Rajkumar, Pacheco, Jason, & Sudderth, Erik. 2011. *Max-Product Particle Belief Propagation*. Tech. rept. Brown University. 104, 105, 107

Kroon, D.J., & Slump, C.H. 2009. MRI modalitiy transformation in demon registration. *Pages 963–966 of: IEEE International Symposium on Biomedical Imaging: From Nano to Macro (ISBI)*. 131, 132

Kumar, M Pawan, Torr, Philip HS, & Zisserman, Andrew. 2010. Objcut: Efficient Segmentation Using Top-down and Bottom-up Cues. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **32**(3), 530–545. 165

Kumar, Sanjiv. 2005. *Models for learning spatial interactions in natural images for context-based classification*. Ph.D. thesis, Carnegie Mellon University. 190

Kumar, Sanjiv, & Hebert, Martial. 2003a. Discriminative Random Fields: A Discriminative Framework for Contextual Interaction in Classification. *Pages 1150–1157 of: IEEE International Conference on Computer Vision (ICCV)*, vol. 2. 2, 18, 80

Kumar, Sanjiv, & Hebert, Martial. 2003b. Man-Made Structure Detection in Natural Images using a Causal Multiscale Random Field. *Pages 119–126 of: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 81

Ladickỳ, Lubor, Sturgess, Paul, Russell, Chris, Sengupta, Sunando, Bastanlar, Yalin, Clocksin, William, & Torr, Philip HS. 2012. Joint optimization for object class segmentation and dense stereo reconstruction. *International Journal of Computer Vision*, **100**(2), 122–133. 5, 21, 69, 70, 72, 74, 75, 76, 77

Lafarge, Florent, Descombes, Xavier, Zerubia, Josiane, & Pierrot-Deseilligny, Marc. 2008. Automatic Building Extraction from DEMs Using an Object Approach and Application to the 3D-city Modeling. *ISPRS Journal of Photogrammetry and Remote Sensing*, **63**(3), 365–381. 165

Lafarge, F., Gimelfarb G., & Descombes, X. 2010. Geometric Feature Extraction by a Multi-marked Point Process. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, **32**(9), 1597–1609. 167

Lafferty, J., McCallum, A., & Pereira, F. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. *Pages 282–289 of: International Conference on Machine Learning (ICML)*. 2, 18, 47, 50, 80, 176

Lee, D., Hebert, M., & Kanade, T. 2009. Geometric Reasoning for Single Image Structure Recovery. *In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 47, 49

Lee, D., Gupta, A., Hebert, M., & Kanade, T. 2010. Estimating Spatial Layout of Rooms using Volumetric Reasoning about Objects and Surfaces. *Pages 1288–1296 of: Advances in Neural Information Processing Systems (NIPS)*. 44, 45, 53, 55, 58, 60, 67

Lee, Yong Jae, Kim, Jaechul, & Grauman, Kristen. 2011. Key-segments for video object segmentation. *Pages 1995–2002 of: IEEE International Conference on Computer Vision (ICCV)*. 8, 23, 89, 92, 93, 95, 97, 101

Li, Bing Nan, Chui, Chee Kong, Chang, Stephen, & Ong, Sim Heng. 2011. Integrating spatial fuzzy clustering with level set methods for automated medical image segmentation. *Computers in Biology and Medicine*, **41**(1), 1–10. 124, 136, 138, 140

Li, Stan. 1994. A Markov Random Field Model for Object Matching under Contextual Constraints. *Pages 866–869 of: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 164

Li, Stan Z. 2009. *Markov Random Field Modeling in Image Analysis*. Vol. 26. Springer. 164, 169, 190, 192

Liao, Wentong, Tang, Jun, Rosenhahn, Bodo, & Yang, Micheal Ying. 2015. Integration of Gaussian Process and MRF for Hyperspectral Image Classification. *In: IEEE Joint Urban Remote Sensing Event*. 16, 31, 189

Lin, Wen-Chieh, & Liu, Yanxi. 2006. Tracking dynamic near-regular textures under occlusion and rapid movements. *Pages 44–55 of: European Conference on Computer Vision (ECCV)*. 104, 105, 112

Linguraru, Marius George, Pura, John A, Pamulapati, Vivek, & Summers, Ronald M. 2012. Statistical 4D graphs for multi-organ abdominal segmentation from multiphase CT. *Medical Image Analysis*, **16**(4), 904–914. 124

Liu, Fei, Xu, Dongxiang, Yuan, Chun, & Kerwin, William S. 2006. Image segmentation based on Bayesian network-Markov random field model and its application to in vivo plaque composition. *Pages 141–144 of: IEEE International Symposium on Biomedical Imaging: From Nano to Macro (ISBI)*. 81

Liu, Hui, Zhan, Yunfeng, & Zhang, Caiqing. 2011. Medical Image Segmentation Based on Contextual Label Tree for Retrieval. *Journal Computational Information System*, **7**(5), 1472–1478. 124

Liu, Qiong, Yang, You, Ji, Rongrong, Gao, Yue, & Yu, Li. 2012. Cross-view down/up-sampling method for multiview depth video coding. *IEEE Signal Processing Letters*, **19**(5), 295–298. 70

Lowe, D.G. 2004. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, **60**(2), 91–110. 38

Ludwig, O., Delgado, D., Goncalves, V., & Nunes, U. 2009. Trainable classifier-fusion schemes: An application to pedestrian detection. *Pages 1–6 of: IEEE Intelligent Transportation Systems (ITSC)*. 113

Ma, Tianyang, & Latecki, Longin Jan. 2012. Maximum weight cliques with mutex constraints for video object segmentation. *Pages 670–677 of: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 92

Mansouri, Abdol-Reza, Mitiche, Amar, & Vázquez, Carlos. 2006. Multiregion competition: A level set extension of region competition to multiple region image partitioning. *Computer Vision and Image Understanding*, **101**(3), 137–150. 124, 134, 135, 136, 137, 138, 140

Martin, David, Fowlkes, Charless, Tal, Doron, & Malik, Jitendra. 2001. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. *Pages 416–423 of: IEEE International Conference on Computer Vision (ICCV)*, vol. 2. 74

Mastin, Andrew, Kepner, Jeremy, & Fisher, J. 2009. Automatic Registration of LIDAR and Optical Images of Urban Scenes. *Pages 2639–2646 of: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 178

Matas, Jiri, Galambos, Charles, & Kittler, Josef. 2000. Robust Detection of Lines Using the Progressive Probabilistic Hough Transform. *Computer Vision and Image Understanding*, **78**(1), 119–137. 47, 48, 49

Mayer, Helmut. 1999. Automatic Object Extraction from Aerial Imagery - a Survey Focusing on Buildings. *Computer Vision and Image Understanding*, **74**(2), 138–149. 13, 28, 164

McKenna, J., & Charif, Nait. 2004. Summarising contextual activity and detecting unusual inactivity in a supportive home environment. *Pattern Analysis and Applications*, **7**, 386–401. 3, 20, 43, 44, 45

Meier, Raphael, Bauer, Stefan, Slotboom, Johannes, Wiest, Roland, & Reyes, Mauricio. 2013. A hybrid model for multimodal brain tumor segmentation. *Multimodal Brain Tumor Segmentation*, 31. 157, 158, 159

Melgani, F., & Lorenzo, B. 2004. Classification of Hyperspectral Remote Sensing Images with Support Vector Machines. *IEEE Transactions on Geoscience and Remote Sensing*, **42**(8), 1778–1790. 15, 30, 189, 193

Menze, Bjoern, Jakab, Andras, Bauer, Stefan, Kalpathy-Cramer, Jayashree, Farahani, Keyvan, & et al. 2014. The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS). *IEEE Transactions on Medical Imaging*, 33. 150, 151, 155, 157, 158, 159

Minka, T.P. 2001. *A family of algorithms for approximate Bayesian inference*. Ph.D. thesis, Massachusetts Institute of Technology. 191

Mishra, Akshaya, Wong, Alexander, Bizheva, Kostadinka, & Clausi, David A. 2009. Intra-retinal layer segmentation in optical coherence tomography images. *Optics Express*, **17**(26), 23719–23728. 124

Modestino, J. W., & Zhang, J. 1992. A Markov Random Field Model-Based Approach to Image Interpretation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **14**(6), 606–615. 80

Moon, N., Bullitt, E., Leemput, K. V., & Gerig, G. 2002. Automatic brain and tumor segmentation. *Pages 372–379 of: Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. Springer. 142, 149

Mortensen, Eric N., & Jia, Jin. 2006. Real-Time Semi-Automatic Segmentation Using a Bayesian Network. *Pages 1007–1014 of: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 81

Müller, O., Yang, M. Y., & Rosenhahn, B. 2013. `http://www.tnt.uni-hannover.de/papers/view_paper.php?id=996`. 104

Müller, Oliver, Yang, Michael Ying, & Rosenhahn, Bodo. 2013. Slice Sampling Particle Belief Propagation. *Pages 1129–1136 of: IEEE International Conference on Computer Vision (ICCV)*. 2, 10, 18, 26, 103

Murphy, Kevin P., Weiss, Yair, & Jordan, Michael I. 1999. Loopy Belief Propagation for Approximate Inference: An Empirical Study. *Pages 467–475 of: Uncertainty in Artificial Intelligence (UAI)*. 93

Neal, Radford M. 2003. Slice sampling. *Annals of Statistics*, **31**(3), 705–767. With discussions and a rejoinder by the author. 103, 105, 106, 108

Ng, Andrew Y., Jordan, Michael I., & Weiss, Yair. 2001. On Spectral Clustering: Analysis and an algorithm. *Pages 849–856 of: Advances in Neural Information Processing Systems (NIPS)*. 38

Niemeyer, Joachim, Rottensteiner, Franz, & Soergel, Uwe. 2014. Contextual Classification of Lidar Data and Building Object Detection in Urban Areas. *ISPRS Journal of Photogrammetry and Remote Sensing*, **87**, 152–165. 176

Nocedal, J., & Wright, S.J. 2006. *Numerical Operation*. Springer. 193

Nocera, Lucien, & Gee, James C. 1997. Robust partial-volume tissue classification of cerebral MRI scans. *Pages 312–322 of: SPIE 3034, Medical Imaging*. 124

Ochs, Peter, & Brox, Thomas. 2011. Object Segmentation in Video: A Hierarchical Variational Approach for Turning Point Trajectories into Dense Regions. *Pages 1583–1590 of: IEEE International Conference on Computer Vision (ICCV)*. 92, 98, 99, 100

Okada, Toshiyuki, Linguraru, Marius George, Yoshida, Yasuhide, Hori, Masatoshi, Summers, Ronald M, Chen, Yen-Wei, Tomiyama, Noriyuki, & Sato, Yoshinobu. 2012. Abdominal multi-organ segmentation of CT images based on hierarchical spatial modeling of organ interrelations. *Pages 173–180 of: Abdominal Imaging Computational and Clinical Applications*. Springer. 124

Oriolo, G., Ulivi, G., & Vendittelli, M. 1998. Real-time Map Building and Navigation for Autonomous Robots in Unknown Environments. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, **28**(3), 316–333. 43

Ortner, Mathias, Descombes, Xavier, & Zerubia, Josiane. 2007. Building Outline Extraction from Digital Elevation Models Using Marked Point Processes. *International Journal of Computer Vision*, **72**(2), 107–132. 165, 167

Ortner, Mathias, Descombes, Xavier, & Zerubia, Josiane. 2008. A Marked Point Process of Rectangles and Segments for Automatic Analysis of Digital Elevation Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **30**(1), 105–119. 165

Otsu, Nobuyuki. 1979. A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man and Cybernetics*, **9**(1), 62–66. 125, 130, 143

Park, Jaesik, Kim, Hyeongwoo, Tai, Yu-Wing, Brown, Michael S, & Kweon, Inso. 2011. High Quality Depth Map Upsampling for 3Dd-TOF Cameras. *Pages 1623–1630 of: IEEE International Conference on Computer Vision (ICCV)*. 70

Parmehr, Ebadat G, Zhang, Chunsun, & Fraser, Clive S. 2012. Automatic Registration of Multi-source Data Using Mutual Information. *Pages 301–308 of: ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, ISPRS Congress*. 175

Peng, Jian, Hazan, Tamir, McAllester, David, & Urtasun, Raquel. 2011. Convex Max-Product Algorithms for Continuous MRFs with Applications to Protein Folding. *In: International Conference on Machine Learning (ICML)*. 9, 25, 104

Perez, M., Chan, J., & Sahli, H. 2012. Multiscale Conditional Random Fields for Supervised Region Based Labeling and Classification. *Pages 1769–1772, of: IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. 176

Pero, Luca Del, Guan, Jinyan, Brau, Ernesto, Schlecht, Joseph, & Barnard, Kobus. 2011. Sampling Bedrooms. *Pages 2009–2016 of: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 45

Pero, Luca Del, Bowdish, Joshua, Fried, Daniel, Kermgard, Bonnie, Hartley, Emily, & Barnard, Kobus. 2012. Bayesian geometric modeling of indoor scenes. *Pages 2719–2726 of: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 45

Plath, Nils, Toussaint, Marc, & Nakajima, Shinichi. 2009. Multi-Class Image Segmentation Using Conditional Random Fields and Global Classification. *Pages 817–824 of: International Conference on Machine Learning (ICML)*. 81, 84, 85

Price, Brian L., Morse, Bryan S., & Cohen, Scott. 2009. LIVEcut: Learning-based Interactive Video Segmentation by Evaluation of Multiple Propagated Cues. *Pages 779–786 of: IEEE International Conference on Computer Vision (ICCV)*. 91

Prince, S.J. 2012. *Computer Vision: Models, Learning, and Inference*. Cambridge University Press. 193

Qian, C., & Yasuda, K. 2008. Particle Swarm Optimization via Successive Optimization in its Parameter Space. *Pages 932 –937 of: International Conference on Systems, Man and Cybernetics*. 54

Rasmussen, C.E., & Williams, C.K.I. 2006. *Gaussian Processes for Machine Learning*. The MIT Press. 190, 191, 192

Reina, Amelio Vázquez, Avidan, Shai, Pfister, Hanspeter, & Miller, Eric L. 2010. Multiple Hypothesis Video Segmentation from Superpixel Flows. *Pages 268–281 of: European Conference on Computer Vision (ECCV)*. 92, 93

Ren, Xiaofeng, & Malik, Jitendra. 2007. Tracking as Repeated Figure/Ground Segmentation. *Pages 1–8 of: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 91

Reynolds, J., & Murphy, K. 2007. Figure-ground segmentation using a hierarchical conditional random field. *Pages 175–182 of: Canadian Conference on Computer and Robot Vision*. 85

Reza, S., & Iftekharuddin, K. 2013. Multi-class Abnormal Brain Tissue Segmentation Using Texture Features. *Pages 38–42 of: NCI-MICCAI BRATS*. 150, 157, 158, 159

Rother, Carsten, Kolmogorov, Vladimir, & Blake, Andrew. 2004. GrabCut: Interactive Foreground Extraction Using Iterated Graph Cuts. *ACM Transactions on Graphics*, **23**, 309–314. 35, 95, 164

Rubio, José C., Serrat, Joan, & López, Antonio M. 2012. Video Co-segmentation. *Pages 1–12 of: Asian Conference on Computer Vision (ACCV)*. 92

Russell, Bryan C., Freeman, William T., Efros, Alexei A., Sivic, Josef, & Zisserman, Andrew. 2006. Using Multiple Segmentations to Discover Objects and their Extent in Image Collections. *Pages 1605–1614 of: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 80

Rusu, Radu Bogdan. 2009. *Semantic 3D Object Maps for Everyday Manipulation in Human Living Environments*. Ph.D. thesis, Computer Science Department, TUM, Germany. 64

Sabuncu, Mert R, Yeo, BT Thomas, Van Leemput, Koen, Fischl, Bruce, & Golland, Polina. 2010. A generative model for image segmentation based on label fusion. *IEEE Transactions on Medical Imaging*, **29**(10), 1714–1729. 124

Sachdeva, J., Kumar, V., Gupta, I., Khandelwal, N., & C.K., Ahuja. 2013. Segmentation, Feature Extraction, and Multiclass Brain Tumor Classification. *Journal of Digital Imaging*, **26**(6), 1141–1150. 141, 149

Saleemi, I., Hartung, L., & Shah, M. 2010. Scene Understanding by Statistical Modeling of Motion Patterns. *Pages 2069–2076 of: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 4, 20, 43

Salzmann, Mathieu, & Urtasun, Raquel. 2012. Beyond Feature Points: Structured Prediction for Monocular Non-rigid 3D Reconstruction. *Pages 245–259 of: European Conference on Computer Vision (ECCV)*. 9, 25, 104, 105, 112

Sarkar, S., & Boyer, K. L. 1993. Integration, Inference, and Management of Spatial Information Using Bayesian Networks: Perceptual Organization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **15**, 256–274. 6, 22, 80, 81

Schindler, Konrad. 2012. An Overview and Comparison of Smooth Labeling Methods for Land-cover Classification. *IEEE Transactions on Geoscience and Remote Sensing*, **50**(11), 4534–4545. 176

Schnitzspan, P., Fritz, M., & Schiele, B. 2008. Hierarchical Support Vector Random Fields: Joint Training to Combine Local and Global Features. *Pages 527–540 of: European Conference on Computer Vision (ECCV)*. 81

Schuster, M., Okerman, J., Nguyen, H., Rehg, J., & Kemp, C. 2010. Perceiving clutter and surfaces for object placement in indoor environments. *Pages 152–159 of: IEEE-RAS International Conference on Humanoid Robots, Humanoids*. 45

Schwing, A. G., & Urtasun, R. 2012. Efficient Exact Inference for 3D Indoor Scene Understanding. *Pages 299–313 of: European Conference on Computer Vision (ECCV)*. 45

Selvaraj, H., Thamarai, S., Selvathi, D., & Gewali, L. 2007. Brain MRI Slices Classification Using Least Squares Support Vector Machine. *International Journal of Intelligent Computing in Medical Sciences and Image Processing*, **1**(1), 21–33. 142

Sengupta, Sunando, Greveson, Eric, Shahrokni, Ali, & Torr, Philip HS. 2013. Urban 3d semantic modelling using stereo vision. *Pages 580–585 of: IEEE International Conference on Robotics and Automation (ICRA)*. 5, 21, 70

Shattuck, David W, Sandor-Leahy, Stephanie R, Schaper, Kirt A, Rottenberg, David A, & Leahy, Richard M. 2001. Magnetic resonance image tissue classification using a partial volume model. *NeuroImage*, **13**(5), 856–876. 123, 124

Shen, Wei, Zhang, Jin, & Yuan, Feng. 2011. A new algorithm of building boundary extraction based on LIDAR data. *Pages 1–4 of: IEEE International Conference on Geoinformatics*. 181

Shi, Jianbo, & Malik, Jitendra. 2000. Normalized Cuts and Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **22**(8), 888–905. 3, 19, 35, 36, 38, 41, 92, 95, 96

Shimizu, Akinobu, Ohno, Rena, Ikegami, Takaya, Kobatake, Hidefumi, Nawano, Shigeru, & Smutek, Daniel. 2007. Segmentation of multiple organs in non-contrast 3D abdominal CT images. *International Journal of Computer Assisted Radiology and Surgery*, **2**(3-4), 135–142. 124

Shimoni, Michal, Tolt, Gustav, Perneel, Christiaan, & Ahlberg, Jörgen. 2011. Detection of vehicles in shadow areas using combined hyperspectral and lidar data. *Pages 4427–4430 of: IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. 175

Shitrit, H. B., Berclaz, J., Fleuret, F., & Fua, P. 2011. Tracking Multiple people under Global Appearance Constraints. *Pages 137–144 of: IEEE International Conference on Computer Vision (ICCV)*. 105

Shoaib, M., Dragon, R., & Ostermann, J. 2009. Shadow Detection for Moving Humans Using Gradient-Based Background Subtraction. *In: ICASSP International Conference on Acoustics, Speech and Signal Processing*. 46

Shoaib, M., Dragon, R., & Ostermann, J. 2011. Context-aware visual analysis of elderly activity in a cluttered home environment. *EURASIP Journal on Advances in Signal Processing*, **129**, 1–14. 46

Shoaib, Muhammad, Yang, Michael Ying, Rosenhahn, Bodo, & Ostermann, Jörn. 2014. Estimating Layout of Cluttered Indoor Scenes Using Trajectory-based Priors. *Image Vision Computing*, **32**(11), 870–883. 5, 21, 43

Shotton, Jamie, Winn, John, Rother, Carsten, & Criminisi, Antonio. 2009. Textonboost for image understanding: Multi-class object recognition and segmentation by jointly modeling texture, layout, and context. *International Journal of Computer Vision*, **81**(1), 2–23. 176, 178, 180, 184, 185, 186

Silberman, N., Hoiem, D., Kohli, P., & Fergus, R. 2012. Indoor Segmentation and Support Inference from RGBD Images. *Pages 746–760 of: European Conference on Computer Vision (ECCV)*. 45, 61

Snyder, Wesley E. 2002. *NC State University Image Analysis Laboratory Database*. 134, 135, 138, 139, 140

Sokolova, M., & Lapalme, G. 2009. A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, **45**(4), 427–437. 147

Solberg, A.H.S., Torfinn, T., & Jain, A.K. 1996. A Markov Random Field Model for Classification of Multisource Satellite Imagery. *IEEE Transactions on Geoscience and Remote Sensing*, **34**(1), 100–113. 190

Soni, Ameet. 2007. *Brain Tissue Classification of Magnetic Resonance Images Using Conditional Random Fields*. Tech. rept. Department of Computer Sciences, University of Wisconsin-Madison. 124

Sowmya, Arcot, & Trinder, John. 2000. Modelling and Representation Issues in Automated Feature Extraction from Aerial and Satellite Images. *ISPRS Journal of Photogrammetry and Remote Sensing*, **55**(1), 34–47. 164

Strauss, David. 1975. A Model for Clustering. *Biometrika*, **62**(2), 467–475. 167

Sudderth, Erik B., Ihler, Alexander T., Isard, Michael, Freeman, William T., & Willsky, Alan S. 2010. Nonparametric Belief Propagation. *Communications of the ACM*, **53**(10), 95–103. 9, 25, 104

Sutton, C., & McCallum, A. 2005. Piecewise training for undirected models. *Pages 568–575 of: Uncertainty in Artificial Intelligence (UAI)*. 183

Suzuki, Miyuki, Linguraru, Marius George, Summers, Ronald M, & Okada, Kazunori. 2012. Analyses of missing organs in abdominal multi-organ segmentation. *Pages 256–263 of: Abdominal Imaging. Computational and Clinical Applications*. Springer. 124

Tallón, Miguel, Babacan, S Derin, Mateos, Javier, Do, Minh N, Molina, Rafael, & Katsaggelos, Aggelos. 2012. Upsampling and denoising of depth maps via joint-segmentation. *Pages 245–249 of: European Signal Processing Conference (EUSIPCO)*. 70

Tappen, Marshall, & Freeman, William. 2003. Comparison of Graph Cuts with Belief Propagation for Stereo, Using Identical MRF Parameters. *Pages 900–907 of: IEEE International Conference on Computer Vision (ICCV)*. 164

Taylor, C., & Cowley, A. 2012. Parsing Indoor Scenes Using RGB-D Imagery. *In: Robotics: Science and Systems*. 45

Thiagarajan, J., Ramamurthy, K., Rajan, D., & Spanias, A. 2013. Kernel Sparse Models for Automatic Tumor Segmentation. *International Journal on Artificial Intelligence Tools*, 1–12. 141, 150

Thrun, S., Martin, C., Liu, Yufeng, Hahnel, D., Emery-Montemerlo, R., Chakrabarti, D., & Burgard, W. 2004. A Real-time Expectation-Maximization Algorithm for Acquiring Multiplanar Maps of Indoor Environments with Mobile Robots. *IEEE Transactions on Robotics and Automation*, **20**(3), 433–443. 4, 20, 43

Tournaire, O., Brédif, M., Boldo, D., & Durupt, M. 2010. An Efficient Stochastic Approach for Building Footprint Extraction from Digital Elevation Models. *ISPRS Journal of Photogrammetry and Remote Sensing*, **65**(4), 317–327. 165

Tsai, David, Flagg, Matthew, Nakazawa, Atsushi, & Rehg, James M. 2012. Motion Coherent Tracking Using Multi-label MRF Optimization. *International Journal of Computer Vision*, **100**(2), 190–202. 91, 98, 99, 100, 101

Tsai, G., Xu, Changhai, Liu, Jingen, & Kuipers, B. 2011. Real-time Indoor Scene Understanding Using Bayesian Filtering with Motion Cues. *Pages 121–128 of: IEEE International Conference on Computer Vision (ICCV)*. 45

Tsotsos, J.K. 1988. A 'Complexity Level' Analysis of Immediate Vision. *International Journal of Computer Vision*, **2**(1), 303–320. 6, 22, 79

Tu, Zhuowen, & Zhu, Song-Chun. 2002. Image Segmentation by Data-driven Markov Chain Monte Carlo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **24**(5), 657–673. 171

Tustison, N., Wintermark, M., Durst, C., & Avants, B. 2013. ANTs and Arboles. *Pages 47–50 of: NCI-MICCAI BRATS*. 150, 158, 159

Vazquez, Carlos, Mitiche, Amar, & Ayed, Ismail Ben. 2004. Image segmentation as regularized clustering: A fully global curve evolution method. *Pages 3467–3470 of: IEEE International Conference on Image Processing (ICIP)*, vol. 5. 124, 134, 135, 136, 137, 138, 140

Vedaldi, A., & Fulkerson, B. 2008. VLFeat: An open and portable library of computer vision algorithms. *http://www.vlfeat.org/*. 38

Vijayanarasimhan, Sudheendra, & Grauman, Kristen. 2012. Active Frame Selection for Label Propagation in Videos. *Pages 496–509 of: European Conference on Computer Vision (ECCV)*. 91

Vincent, Luc, & Soille, Pierre. 1991. Watersheds in Digital Spaces: An Efficient Algorithm Based on Immersion Simulations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **13**(6), 583–598. 85

Wainwright, Martin J., Jaakkola, Tommi S., & Willsky, Alan S. 2005. Map Estimation via Agreement on Trees: Message-passing and Linear Programming. *IEEE Transactions on Information Theory*, **51**, 3697–3717. 19, 103, 104

Walsh, B. 2004. Markov Chain Monte Carlo and Gibbs Sampling. *In: Lecture Notes for EEB 581 version 26*. 104, 112

Wang, H., Gould, S., & Koller, D. 2010. Discriminative Learning with Latent Variables for Cluttered Indoor Scene Understanding. *Pages 497–510 of: European Conference on Computer Vision (ECCV)*. 44, 45

Wang, Huayan, Gould, Stephen, & Roller, Daphne. 2013. Discriminative learning with latent variables for cluttered indoor scene understanding. *Communications of the ACM*, **56**(4), 92–99. 20, 44

Wang, Jiahui, Li, Feng, & Li, Qiang. 2009. Automated segmentation of lungs with severe interstitial lung disease in CT. *Medical Physics*, **36**(10), 4592–4599. 136

Warfield, Simon K, Kaus, Michael, Jolesz, Ferenc A, & Kikinis, Ron. 2000. Adaptive, template moderated, spatially varying statistical classification. *Medical Image Analysis*, **4**(1), 43–55. 147, 150, 155, 156, 158

Weiss, Nick, Rueckert, Daniel, & Rao, Anil. 2013. Multiple Sclerosis Lesion Segmentation Using Dictionary Learning and Sparse Coding. *Pages 735–742 of: Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. 142, 147, 148, 150

Williams, C.K., & David, B. 1998. Bayesian classification with Gaussian processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **20**(12), 1342–1351. 191, 192

Winkler, Gerhard. 2003. *Image Analysis, Random Fields and Markov Chain Monte Carlo Methods: A Mathematical Introduction*. Vol. 27. Springer. 164

Winn, John, & Shotton, Jamie. 2006. The Layout Consistent Random Field for Recognizing and Segmenting Partially Occluded Objects. *Pages 37–44 of: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 165

Wu, C., & Aghajan, H. 2011. User-centric Environment Discovery with Camera Networks in Smart Homes. *IEEE Transactions on Systems, Man, and Cybernetics Part A*, **41**(2), 375–383. 46

Wu, T.F., Lin, C.J., & Weng, R.C. 2004. Probability estimates for multi-class classification by pairwise coupling. *The Journal of Machine Learning Research*, **5**, 975–1005. 142, 193

Xu, Ken, Stewart, James, & Fiume, Eugene. 2002. Constraint-based Automatic Placement for Scene Composition. *Pages 25–34 of: Graphics Interface*. 43

Xue, Jianru, Zheng, Nanning, Geng, J., & Zhong, Xiaopin. 2008. Tracking Multiple Visual Targets via Particle-Based Belief Propagation. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, **38**(1), 196 –209. 105

Yang, Michael Ying. 2011. Hierarchical and Spatial Structures for Interpreting Images of Man-made Scenes Using Graphical Models. *Ph. D. thesis, Department of Photogrammetry, University of Bonn*. 18

Yang, Michael Ying. 2013. Image Segmentation by Bilayer Superpixel Grouping. *Pages 552–556 of: IAPR Asian Conference on Pattern Recognition*. 3, 19, 35

Yang, Michael Ying. 2015. A Generic Probabilistic Graphical Model for Region-based Scene Interpretation. *In: International Conference on Computer Vision Theory and Applications (VISAPP)*. 2, 7, 18, 23, 79

Yang, Michael Ying, & Förstner, Wolfgang. 2011a. A Hierarchical Conditional Random Field Model for Labeling and Classifying Images of Man-made Scenes. *Pages 196 – 203 of: International Conference on Computer Vision, IEEE/ISPRS Workshop on Computer Vision for Remote Sensing of the Environment*. 176, 193

Yang, Michael Ying, & Förstner, Wolfgang. 2011b. Regionwise Classification of Building Facade Images. *Pages 209 – 220 of: Photogrammetric Image Analysis (PIA2011)*. LNCS 6952. Springer. 176

Yang, Michael Ying, & Rosenhahn, Bodo. 2014. Video Segmentation with Joint Object and Trajectory Labeling. *Pages 831–838 of: IEEE Winter Conference on Applications of Computer Vision (WACV)*. 2, 9, 18, 24, 89

Yang, Michael Ying, Förstner, Wolfgang, & Drauschke, Martin. 2010. Hierarchical Conditional Random Field for Multi-class Image Classification. *Pages 464–469 of: International Conference on Computer Vision Theory and Applications (VISAPP)*. 84

Yang, Qingxiong, Yang, Ruigang, Davis, James, & Nistér, David. 2007. Spatial-depth super resolution for range images. *Pages 1–8 of: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 5, 21, 70

Yao, Jian, Fidler, Sanja, & Urtasun, Raquel. 2012. Describing the scene as a whole: Joint object detection, scene classification and semantic segmentation. *Pages 702–709 of: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 70

Yedidia, J.S., Freeman, W.T., & Weiss, Y. 2000. Generalized Belief Propagation. *Pages 689–695 of: Advances in Neural Information Processing Systems*, vol. 13. 19

Yu, Stella X., & Shi, Jianbo. 2003. Multiclass Spectral Clustering. *Pages 313–319 of: IEEE International Conference on Computer Vision*. 38

Yuen, Jenny, Russell, Bryan C., Liu, Ce, & Torralba, Antonio. 2009. LabelMe Video: Building a Video Database with Human Annotations. *Pages 1451–1458 of: IEEE International Conference on Computer Vision (ICCV)*. 91

Zabih, Ramin, & Kolmogorov, Vladimir. 2004. Spatially Coherent Clustering Using Graph Cuts. *Pages 437–444 of: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2. 164

Zhang, Lei, & Ji, Qiang. 2010. Image Segmentation with a Unified Graphical Model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **32**(8), 1406–1425. 81

Zhang, T., Anand, A., Joachims, T., & Saxena, A. 2011. Learning Semantic Scene Models by Object Classification and Trajectory Clustering. *Pages 244–252 of: Advances in Neural Information Processing Systems (NIPS)*. 4, 20, 43, 44

Zhang, Z., Zhou, M., Tang, L., & Li, C. 2012. Automatic detection and mapping of urban buildings in high resolution remote sensing images. *Pages 5721–5724 of: IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. 176

Zhang, Zheng, Yang, Michael Ying, & Zhou, Mei. 2013. Multi-Source Hierarchical Conditional Random Field Model for Feature Fusion of Remote Sensing Images and LiDAR Data. *Pages 389–392 of: International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences; ISPRS Hannover Workshop*. 14, 30, 176, 184, 185, 186

Zhang, Zheng, Yang, Michael Ying, & Zhou, Mei. 2015. Multi-Source Multi-Scale Hierarchical Conditional Random Field Model for Remote Sensing Image Classification. *In: ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, PIA15+HRIGI15*. 15, 30, 175

Zhao, Kaiguang, Popescu, Sorin, & Zhang, Xuesong. 2008. Bayesian learning with Gaussian processes for supervised classification of hyperspectral data. *Photogrammetric Engineering & Remote Sensing*, **74**(10), 1223–1234. 190, 193

Zhao, L., Sarikaya, D., & Corso, J. 2013a. Automatic Brain Tumor Segmentation with MRF on Supervoxels. *NCI-MICCAI BRATS*, 51–54. 150, 158, 159

Zhao, L., Wu, W., & Corso, J. 2013b. Semi-automatic Brain Tumor Segmentation by Constrained MRFs Using Structural Trajectories. *Pages 567–575 of: Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. 150

Zhu, Jiejie, Wang, Liang, Gao, Jizhou, & Yang, Ruigang. 2010. Spatial-temporal fusion for high accuracy depth maps using dynamic MRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **32**(5), 899–909. 5, 21, 70

Zhu, Song Chun, & Yuille, Alan. 1996. Region competition: Unifying snakes, region growing, and Bayes/MDL for multiband image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **18**(9), 884–900. 133

Zou, Kelly, Warfield, Simon K, Bharatha, Aditya, Tempany, Clare, Kaus, Michael R, Haker, Steven J, Wells III, William M, Jolesz, Ferenc A, & Kikinis, Ron. 2004. Statistical validation of image segmentation quality based on a spatial overlap index. *Academic Radiology*, **11**(2), 178–189. 136