PRECISE VEHICLE RECONSTRUCTION FOR AUTONOMOUS DRIVING APPLICATIONS

Max Coenen*, Franz Rottensteiner, Christian Heipke

Institute of Photogrammetry and GeoInformation, Leibniz Universität Hannover, Germany
(coenen, rottensteiner, heipke)@ipi.uni-hannover.de

Commission II, WG II/4

KEY WORDS: Object detection, 3D reconstruction, 3D modelling, pose estimation, autonomous driving

ABSTRACT:

Interactive motion planing and collaborative positioning will play a key role in future autonomous driving applications. For this purpose, the precise reconstruction and pose estimation of other traffic participants, especially of other vehicles, is a fundamental task and will be tackled in this paper based on street level stereo images obtained from a moving vehicle. We learn a shape prior, consisting of vehicle geometry and appearance features, and we fit a vehicle model to initially detected vehicles. This is achieved by minimising an energy function, jointly incorporating 3D and 2D information to infer the model's optimal and precise pose parameters. For evaluation we use the object detection and orientation benchmark of the KITTI dataset (Geiger et al., 2012). We can show a significant benefit of each of the individual energy terms of the overall objective function. We achieve good results with up to 94.8% correct and precise pose estimations with an average absolute error smaller than 3° for the orientation and 33 cm for position.

1. INTRODUCTION

Autonomous driving comes with the need to deal with highly dynamic environments. For safe navigation, interactive motion planing and collaborative positioning, 3D scene reconstruction in general and the identification and precise reconstruction of moving objects such as other cars are fundamental tasks. Interaction and collaboration between vehicles requires knowledge about their relative poses. In this context, stereo cameras provide a cost-effective solution for the perception of a vehicle's surroundings. Existing techniques for image based vehicle detection and pose estimation are often restricted to only coarse viewpoint estimation, whereas the precise determination of vehicle poses, especially of the orientation, and vehicle shape is still an open problem. We tackle this problem and propose a method for precise 3D vehicle reconstruction from street level stereo images. The poses of other vehicles w.r.t. the observing vehicle, i.e. their relative position and orientation, can directly be derived from the reconstructions. Using CAD vehicle models to learn a vehicle shape prior, we formulate a comprehensive objective function leveraging the shape prior, 3D and 2D information. This objective function is minimized by an iterative Monte Carlo sampling technique to determine the precise vehicle pose and shape.

2. RELATED WORK

This section provides a brief overview of related work for vehicle pose estimation, vehicle reconstruction and vehicle modelling.

A coarse estimation of the vehicle orientation is delivered already by a number of vehicle detection approaches, such as the viewpoint specific detectors in (Ozuysal et al. 2009; Payet and Todorovic 2011; Villamizar et al. 2011), and by some part based detectors (e.g. Felzenszwalb et al. 2010; Leibe et al. 2006). However, all these methods are solely 2D appearance based and typically only deliver 2D bounding boxes and coarse viewpoint estimates as output. The approaches of Chen et al. (2015) and Mousavian et al. (2017) deliver bounding box estimates in 3D using convolutional neural networks (CNN). However, describing objects by a box only gives a very coarse representation of their shape.

CNNs are also trained in (Tulsiani and Malik 2015; Wang et al. 2018; Xiang et al. 2017; Kundu et al. 2018) to detect objects and estimate their pose. However, the number of classes to consider in the CNN is the product of the number of pose parameters and the number of the discretised pose bins and consequently becomes extremely high for the task of precise pose estimation. Besides, in (Tulsiani and Malik, 2015) and (Kundu et al., 2018) only the angular viewpoint is estimated, neglecting the object translation, while Wang et al. (2018) exhibit failure cases especially for the estimation of the translation and Xiang et al. (2017) only achieve coarse translation estimations. We aim at obtaining precise vehicle poses, including position and orientation, in 3D space.

Another way of capturing 3D object information from images is pursued by approaches which internally enrich part-based detectors by linking 3D object knowledge to the parts and transferring this information to the objects after detection. To that end, the increasing amount of freely available CAD data is often exploited. For instance, in (Liebelt and Schmid, 2010) appearance and geometry are treated as separate learning tasks. An appearance part model is trained from real images and each part of the training data is linked with 3D geometry from synthetic models, which allows an approximate estimation of 3D pose. Similarly, Pepik et al. (2012) adapt the deformable part model (DPM) (Felzenszwalb et al., 2010). They add 3D information from CAD models to the deformable parts and incorporate 3D constraints to enforce part correspondences. Thomas et al. (2007) enrich the Implicit Shape Model (ISM) of (Leibe et al., 2006) by adding depth information from training images to the ISM and transfer the 3D information to the test images, which allows the estimation of coarse 3D pose information. While the latter approaches only use the 3D information implicitly, by transferring the learned 3D information to

*Corresponding author

the detected objects, 3D model information can be used explicitly by deriving cues from the model representation and using these cues actively for vehicle detection, reconstruction and/or to infer pose information. In (Ansari et al., 2018; Chabot et al., 2017; Murthy et al., 2017), a 3D vehicle model is fitted to vehicle keypoints detected in the image by a CNN. However, these approaches are prone to imprecise and incorrect keypoint localisations. Another commonly applied procedure is to use an arbitrary object detector to initialise or instantiate the model, followed by fine-grained model fitting or optimisation. Approaches for 3D scene reconstruction (Bao et al., 2013; Dame et al., 2013; Güney and Geiger, 2015) follow this procedure by initially detecting vehicles and subsequently integrating vehicle models into the 3D reconstruction algorithm. In (Engelmann et al., 2016) a Signed Distance Function (SDF) is used for pose and shape estimation of vehicles detected in stereo images. The SDF is fitted to detected vehicles by minimising the distance of reconstructed 3D vehicle points to the SDF. However, a SDF is a rather complex object representation and its level of detail depends on the applied voxel grid size. Active Shape Models (ASM) (Cootes et al., 1995) provide a less complex method to represent the geometry of an object class while being able to handle intra-class variability. Such models have already been used in the context of vehicle detection and pose estimation. For instance, based on 3D points from mobile laserscanning data, Xiao et al. (2016) use a 3D vehicle ASM to fit it to detected and segmented generic street scene objects. However, in (Engelmann et al., 2016) and (Xiao et al., 2016) image information is not used at all or only for the initial vehicle detection, but image cues are disregarded for model fitting. In contrast, Zia et al. (2013) only use single images and incorporate a 3D ASM into their detection approach, using the model also to derive precise object pose estimates. For this purpose, they apply a model-keypoint based multi-class classifier. However, their results show that their approach heavily depends on good pose initialisations. Similarly, Lin et al. (2014) recover the 3D vehicle geometry by fitting the 3D ASM to estimated 2D landmark locations resulting from a DPM detector. Their approach also suffers from wrongly estimated part locations resulting from the DPM. A 3D ASM is also used in (Menze et al., 2015) to be fitted to detections of vehicles obtained from stereo image pairs and object scene flow. However, using scene flow for object detection is computationally expensive. Coenen et al. (2018) combine 3D stereo information with image cues to derive an ASM representation for vehicle detections. However, their results show only limited benefit from incorporating the image information.

In this paper we want to reconstruct vehicles from street level stereo images and precisely recover their 3D pose and shape. Similarly to (Coenen et al., 2018) and (Zia et al., 2013) we make use of an ASM as vehicle shape prior. Based on initial 3D vehicle detections, we make the following contributions in this paper: **(1)** We extend the ASM of (Zia et al., 2013) by defining two groups of keypoints, one describing the geometrical shape and the other additionally representing distinct vehicle part appearances. As in (Zia et al., 2013) we learn the geometry representation of vehicles on the one hand and appearance patterns by training an image based classifier on the other hand. However, in contrast to (Zia et al., 2013) we do not use the shape defining keypoints for appearance learning, but introduce our additionally defined appearance keypoints for this purpose, believing them to be more distinctive and thus better suited for learning the classifier. **(2)** For the reconstruction of vehicles, we define a probabilistic model, incorporating different types of observations by jointly leveraging features derived from our shape prior, reconstructed 3D data, scene knowledge, and image information. The formulation of our probabilistic model is inspired by the energy formulation in (Coenen et al., 2018). However, we modify this energy function to overcome some problems reported by the authors. Further, we extend their formulation by an additional likelihood term inspired by (Zia et al., 2013). Our experiments show that these extensions lead to major enhancements for vehicle reconstruction. **(3)** In contrast to (Zia et al., 2013) we do not rely on good pose initialisations, especially initial orientation information is not needed. Instead, we define a robust model initialisation and fitting procedure based on an iterative Monte Carlo model particle sampling technique. **(4)** We do not restrict ourselves to a discretised and small number of orientation bins, but rather deliver fine-grained pose parameters and detailed vehicle shape, thus going beyond common pose estimation approaches as e.g. (Felzenszwalb et al. 2010; Tulsiani and Malik 2015; Wang et al. 2018).

3. METHOD

3.1 Overview

The goal of our method is to precisely and fully reconstruct vehicles detected from street level stereo images acquired from a moving platform with an approximately horizontal viewing direction. We want to deduce a 3D vehicle model which represents the detected vehicle best in terms of pose (position and orientation) and shape. For this purpose we learn a parametrized deformable model, used as shape prior, which we fit to the detected vehicles based on geometrical and appearance based information derived from the stereo images. A schematic overview of our method is given in Fig. 1. The input to our method are stereo images with known interior and relative orientation parameters incl. base length. The stereo image pairs are processed independently. We define the left stereo partner to be the reference image and use the ELAS matcher (Geiger et al., 2011) to derive a dense disparity map. Using the disparity image, we reconstruct 3D points ${}^{M}\mathbf{X}$ for every pixel of the reference image in the 3D model coordinate system ${}^{M}C$ via triangulation. This system is centered at the projection centre of the left camera. Its x-y plane is parallel to the image plane and its z-axis points to the viewing direction. We introduce a user-defined maximum allowable threshold for the depth precision δ_{σ_Z} of the reconstructed 3D points and discard points with inferior precision. For the vehicle detection we apply the instance segmentation approach described in (He et al., 2017). After a preprocessing step a deformable shape model is precisely fitted to each detected vehicle to recover its pose and shape. The emphasis of this paper is on this **reconstruction** step.

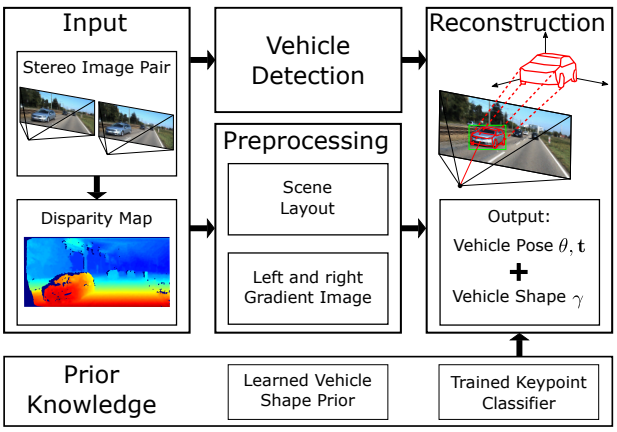


Figure 1. Overview of our framework.

3.2 Formal problem definition

Given a set of detected vehicles $\mathcal{V}$, our goal is to associate each vehicle $\mathbf{v}_k \in \mathcal{V}$ with its state vector $\mathbf{s}_k = (\mathbf{t}_k, \theta_k, \gamma_k)$. After determining the ground plane $\Omega \in \mathbb{R}^3$, we describe the vehicle pose by its 2D position $\mathbf{t}_k$ on the ground plane and its heading θ_k, i.e. the rotation angle about the normal vector of the ground plane; γ_k is a vector of shape parameters determining the shape of the 3D deformable ASM representing each vehicle (cf. Sec. 3.4.1).

3.3 Preprocessing and vehicle detection

Prior to reconstructing the vehicles we use the stereo data to derive knowledge about the 3D layout of the scene, represented by the 3D ground plane and a probabilistic free-space grid map. Using this information we reduce and constrain the parameter space of the model fitting approach.

Ground plane Ω: We apply RANSAC to the stereo point cloud ${}^{M}\mathbf{X}$ to find the ground plane Ω as plane of maximum support. All inliers of the final RANSAC consensus set are stored as ground points ${}^{M}\mathbf{X}_{\Omega} \subset {}^{M}\mathbf{X}$. Additionally to the model coordinate system ${}^{M}C$ we define a ground plane coordinate system ${}^{\Omega}C$. Its origin is the orthogonal projection of the origin of system ${}^{M}C$ to the ground plane. The y-axis is defined in the direction of the plane normal vector and the x/z-plane lies in the ground plane. We also determine parameters of a rigid transformation between the systems ${}^{M}C$ and ${}^{\Omega}C$. Using these parameters, we transform all 3D points ${}^{M}\mathbf{X}$ including the points belonging to the ground plane ${}^{M}\mathbf{X}_{\Omega}$ to the ground plane system, resulting in ${}^{\Omega}\mathbf{X}$ and ${}^{\Omega}\mathbf{X}_{\Omega}$, respectively. The subsequent steps are applied in this domain.

Probabilistic free-space grid map Φ: Assuming that vehicles are located on the ground plane and do not exceed a maximum height h_{max}, we define the 3D space above the ground plane and below the threshold h_{max} as corridor of interest for the location of vehicles. We filter all 3D points inside that corridor and store them as points of interest ${}^{\Omega}\mathbf{X}_{Int} \subset {}^{\Omega}\mathbf{X}$ with ${}^{\Omega}\mathbf{X}_{\Omega} \cup {}^{M}\mathbf{X}_{Int} = \emptyset$. Based on the points belonging to the ground plane and the extracted interest points it is possible to reason about free space in the observed scene, i.e. areas on the ground plane which are not occupied by any 3D object. To represent the free space areas we create a probabilistic free space grid map Φ. For this purpose, we create a grid in the ground plane consisting of square cells with a side length l_{Φ}. For each grid cell Φ_g with $g = 1...G$ we count the number of ground points n^g_{Ω} and the number of interest points n^g_{Int} whose vertical projection is within the respective cell. We define the probability ρ_g of each cell to be free space as

$$\rho_g = \frac{n^g_{\Omega}}{n^g_{\Omega} + n^g_{Int}}. \tag{1}$$

Grid cells without projected points are marked as *unknown*.

Gradient information: Using the Sobel operator, we extract gradient information $\nabla = ({}^{l}I_{\nabla}, {}^{r}I_{\nabla})$ where ${}^{l}I_{\nabla}$ and ${}^{r}I_{\nabla}$ are gradient magnitude images of the left (l) and right (r) image of the stereo pair, respectively. We also extract a multi-scale histogram of oriented gradient (HoG) (Dalal and Triggs, 2005) feature vector f_u for every image point u. A set of multi-scale HoG features is denoted by $\mathcal{F}$ with $f_u \in \mathcal{F}$ in the remainder of this paper.

Vehicle detection: Vehicle detections are required as input to our vehicle reconstruction approach. For this purpose we make use of the *mask R-CNN* (mRCNN), the instance segmentation network described in (He et al., 2017). Alongside its good performance it comes with the advantage of not only delivering bounding boxes but also instance segmentation masks for every vehicle. To obtain a list of detected vehicles $\mathbf{v}_k = \left({}^{\Omega}X_k, {}^{l}\mathcal{B}_k, {}^{r}\mathcal{B}_k\right)$, we associate each detection with its object points ${}^{\Omega}X_k$ being reconstructed from the pixels belonging to the respective segmentation masks, as well as with its left and right image bounding boxes ${}^{l}\mathcal{B}_k$ and ${}^{r}\mathcal{B}_k$. As we apply the detection only to the left image of the stereo pair, we make use of the dense stereo correspondences to deduce the segmentation masks and the bounding boxes on the right image.

3.4 Vehicle Reconstruction

We want to reconstruct each detected vehicle in 3D to recover its pose and shape. For this purpose, we fit a deformable vehicle model to each detection, leveraging different types of information derived from a shape prior and from the observed stereo images. We formulate a comprehensive probabilistic model, jointly fusing the different types of observations to obtain the optimal parameters for pose and shape. Details about the shape prior, probabilistic formulation and the optimisation procedure are given in the following sections.

3.4.1 Shape Prior: Similar to Zia et al. (2013) we use a 3D ASM as vehicle shape prior. The ASM is learned by applying principal component analysis (PCA) to a set of manually annotated keypoints $\mathcal{K}$ of 3D CAD vehicle models. A deformed vehicle ASM is defined by the deformed vertex positions $\nu(\gamma)$, which can be obtained by the linear combination

$$\nu(\gamma) = \mathbf{m} + \sum_j \gamma^{(j)} \lambda_j \, \mathbf{e}_j \tag{2}$$

of the mean model $\mathbf{m}$ and the eigenvectors $\mathbf{e}_j$, weighted by their corresponding eigenvalues λ_j and scaled by the object specific shape parameters $\gamma^{(j)}$. The variation of the low dimensional shape vector γ thus allows the generation of different vehicle shapes. For the number of the eigenvalues and eigenvectors to be considered in the ASM we choose $j \in \{1, 2\}$, which we found to be a proper tradeoff between the complexity of the model and the quality of the model approximation. A fully parametrised instance of a 3D vehicle ASM in the ground plane coordinate system, denoted by $M(\mathbf{s})$, can be created by computing the deformed keypoints using the shape vector γ and subsequently shifting and rotating the whole model on the ground plane according to the translation vector $\mathbf{t}$ and a rotation matrix $R_y(\theta)$ derived from the heading angle θ:

$$M_l(\mathbf{s}) = R_y(\theta) \cdot \nu_l(\gamma) + \mathbf{t}, \tag{3}$$

where l is an index for the keypoints in $\mathcal{K}$.

In (Zia et al., 2013) only a comparably low number of keypoints is used for both, the definition of the geometry and for learning an appearance model. In this work, we significantly expand this vehicle representation by defining two groups of keypoints, namely *shape keypoints* $\mathcal{K}_S$ and *appearance keypoints* $\mathcal{K}_A$. $\mathcal{K}_S$ consists of keypoints defining the outer shape of the vehicle model, i.e. it contains keypoints representing corner and boundary points of the vehicle chassis, of the vehicle body and of the tires. $\mathcal{K}_A$ contains keypoints not representing the geometrical shape of a vehicle, but related to distinct visual features such as corner points of the windshield and the rear window, centre points of wheels, the license plate, front and back light, etc. We believe this group of keypoints to lead to more distinctive and discriminative features

for appearance learning compared to the *shape keypoints*. Note that the two groups of keypoints $\mathcal{K}_S$ and $\mathcal{K}_A$ are not mutually exclusive. Besides, while in (Zia et al., 2013), the 3D shape prior is only used to constrain the 2D keypoint configuration in the image, we additionally incorporate its 3D information directly by deriving both, appearance as well as geometrical representations.

Geometrical representation: We represent the model surface by defining a triangular mesh M_Δ for the ASM shape vertices $\mathcal{K}_S$. Further, we use a subset of $\mathcal{K}_S$ to define a wireframe $M_\mathcal{W}$ of the vehicle model, consisting of two types of edges: *crease* edges that describe the outline of the vehicle and *semantic* edges, describing the boundaries between semantically different vehicle parts. Fig. 2 shows the triangulations of several deformed models and their wireframes.

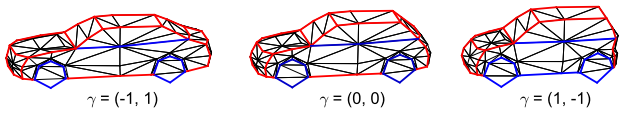
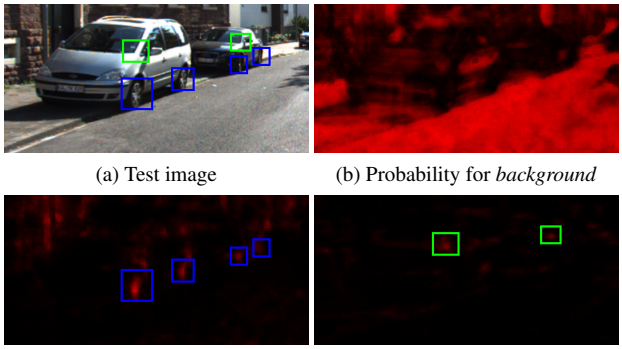


Figure 2. Exemplary ASM with their triangulated surface (black) and wireframe. Red: Crease edges, Blue: semantic edges.

Appearance representation: Inspired by Zia et al. (2013), we train a distinct appearance keypoint detector for all keypoints in $\mathcal{K}_A$, with one class per keypoint and additionally one class for the background. To this end, we manually label image points for every keypoint in real-world images showing vehicles of different types and from different viewpoints and we randomly pick training samples for the background from images not containing any vehicle. To generate training data, we extract multi-scale HoG features for every labeled point. Based on these features, we train a multi-class random forest (RF) classifier (Breiman, 2001). We choose a RF because it has shown to be one of the best classifiers when only a comparably small amount of training data is available and because it can handle multi-modal distributions in the feature space, which can result from the different vehicle viewpoints. Fig. 3 shows a test image and examples for the resulting pixel-wise RF probability map for the background class and two keypoint classes (wheels and windshield corner).



(a) Test image (b) Probability for *background*

(c) Probability for *wheels* (d) Probability for *windshield bottom left corner*

Figure 3. Examples for the output of the keypoint classifier. Brighter red denotes higher probability.

3.4.2 Shape and pose estimation: Given the vehicle detections $\mathbf{v}_k = ({}^\Omega X_k, {}^l\mathcal{B}_k, {}^r\mathcal{B}_k)$ and the information derived from the stereo image pair, we construct an observation vector $\mathbf{o}_k = (\mathbf{v}_k, \Phi, \mathcal{F}, \nabla)$ to formulate our probabilistic model. Our aim is to fit a vehicle model $M(\mathbf{s}_k)$ to each detection by finding the optimal state variables $\hat{\mathbf{s}}_k = (\hat{\mathbf{t}}_k, \hat{\theta}_k, \hat{\gamma}_k)$. Neglecting the index k to simplify our notation in the following sections, $\hat{\mathbf{s}}$ can be derived by maximising the posterior $p(\mathbf{s}|\mathbf{o})$ (MAP). Assuming a uniform prior $p(\mathbf{s})$, the posterior becomes $p(\mathbf{s}|\mathbf{o}) \propto p(\mathbf{o}|\mathbf{s})$ and the MAP corresponds to a maximum likelihood (ML) estimation. Thus, to estimate the optimal parameters, we minimize an energy function $E(\mathbf{o}, \mathbf{s})$ which corresponds to the negative logarithm of the likelihood with

$$E(\mathbf{o}, \mathbf{s}) = \underbrace{-\log p(X|\mathbf{s}) - \log p(\Phi|\mathbf{s})}_{\text{3D information}} \underbrace{- \log p(\mathcal{F}|\mathbf{s}) - \log p(\nabla|\mathbf{s})}_{\text{2D image information}}. \tag{4}$$

In $E(\mathbf{o}, \mathbf{s})$ we incorporate four individual observation likelihood terms. The first two terms consider 3D information derived from the stereo pairs, while the other ones leverage 2D image information. The individual likelihood terms are explained in the following paragraphs.

3D likelihood: The 3D likelihood depends on the distance of the 3D points X from the model surface M_Δ of the model $M(\mathbf{s})$:

$$\log p(X|\mathbf{s}) = -\frac{1}{P} \sum_{x \in X} \frac{d_{\sigma_x}(x, M_\Delta)}{2\sigma_x^2}. \tag{5}$$

In eq. 5, P is the overall number of 3D points in X and σ_x is the depth uncertainty of the individual 3D point x. As the set of 3D points X possibly contains outliers due to segmentation errors of the applied detection approach or matching errors, we use the Huber norm to calculate $d_{\sigma_x}(x, M_\Delta)$ with

$$d_{\sigma_x}(x, M_\Delta) = \begin{cases} \|x, M_\Delta\|_2^2 & \text{if } \|x, M_\Delta\|_2 \leq \sigma_x, \\ 2\sigma_x \cdot \|x, M_\Delta\|_2 - \sigma_x^2) & \text{otherwise.} \end{cases} \tag{6}$$

The Huber norm is more robust against outliers compared to the quadratic distance $\|x, M_\Delta\|_2^2$ of a 3D point x to the model surface M_Δ. This likelihood fits the 3D ASM to the 3D point cloud.

Free-space likelihood: This likelihood term uses the probabilistic free space grid map Φ. It is calculated based on the amount of overlap between the minimum enclosing 2D bounding box $M_\mathcal{B}$ of the model $M(\mathbf{s})$ on the ground plane and the free-space grid map cells Φ_g given their probability ρ_g of being free space:

$$\log p(\Phi|\mathbf{s}) = \lambda_\Phi \frac{1}{A_\mathcal{B}} \sum_{g=1}^{G} \log(1 - \rho_g) \cdot o(M_\mathcal{B}, \Phi_g) \tag{7}$$

In eq. 7, $A_\mathcal{B}$ is the area of the model bounding box. The function $o(\cdot, \cdot)$ calculates the amount of overlap between the model bounding box and a grid cell using the surveyor's area formula (Braden, 1986). The factor $\lambda_\Phi = \min(1, \frac{l_\Phi}{\sigma_M})$ is used to weight this likelihood term based on the grid cell size l_Φ and the depth uncertainty σ_M of a stereo-reconstructed point in the distance of the model $M(\mathbf{s})$. This likelihood penalises models that are partly or fully located in areas which are actually observed as being free space. It acts as substitute information for missing 3D data on the vehicle sides that are invisible to the camera.

Classification likelihood: Here, we make use of the trained RF keypoint classifier. We backproject the non-self-occluded keypoints $\mathcal{K}_A$ to the left (l) and right (r) stereo images and extract the HoG-features $\mathcal{F} = ({}^l\mathcal{F}, {}^r\mathcal{F})$ at the corresponding image positions. The classification likelihood is calculated by

$$\log p(\mathcal{F}|\mathbf{s}) = -\frac{1}{2C} \sum_{i \in \{l,r\}} \sum_{c=1}^{C} \log \frac{1 - \Psi_c({}^i f_c)}{1 - \Psi_b({}^i f_c)} \tag{8}$$

where C is the overall number of backprojected keypoints. The output of the trained keypoint classifier $\Psi_c({}^i f_c)$ denotes the classification probability of the feature vector ${}^i f_c \in {}^i\mathcal{F}$ for its corresponding keypoint class $\mathcal{K}_A^c$ and $\Psi_b({}^i f_c)$ denotes the probability of the same feature vector for the class *background*. This term thus models the likelihood of the model $M(\mathbf{s})$ based on the backprojected keypoint configuration and the observed image features at the corresponding positions.

Gradient likelihood: This likelihood term is based on the model wireframe $M_\mathcal{W}$ and the gradient images $\nabla = ({}^l I_\nabla, {}^r I_\nabla)$. Assuming that the two types of vehicle edges (*crease, semantic*) chosen to define the wireframe to correspond to large image gradients, this likelihood depends on a measure of similarity between the image gradients and the backprojected edges of the model wireframe. For this purpose, considering self-occlusion, we backproject the visible parts of the model wireframe to the left and right images, resulting in binary wireframe images with entries of 1 at pixels that are crossed by a wireframe edge and 0 everywhere else. We consider differences between the real image gradient positions and the model wireframe caused by generalisation effects of our vehicle model representation by blurring the binary wireframe images using a Gaussian filter, resulting in the left and right non-binary wireframe images ${}^l I_\mathcal{W}$ and ${}^r I_\mathcal{W}$. The generalisation error of the 3D model can be quantified by an uncertainty $\sigma_\mathcal{W}$ which is set to 10 cm in this work. We calculate a backprojection uncertainty for the model wireframe by performing error propagation on the backprojection of the centre point $X_{M(\mathbf{s})}$ of the model $M(\mathbf{s})$, setting the point's uncertainty to $\sigma_\mathcal{W}$. The size of the applied Gaussian filter is defined according to the resulting backprojection uncertainty, leading to a stronger blurring effect when the vehicle is close to the camera and vice versa. The gradient likelihood is calculated according to

$$\log p(\nabla|\mathbf{s}) = -\frac{1}{2} \sum_{i \in \{l,r\}} \log\Big(1 - BC({}^i I_\nabla, {}^i I_\mathcal{W})\Big) \quad (9)$$

in which we use the Bhattacharyya coefficient of the gradient image and the wireframe image as similarity measure with

$$BC(I_\nabla, I_\mathcal{W}) = \sum_w^W \sqrt{I_\nabla^w \cdot I_\mathcal{W}^w}. \quad (10)$$

In eq. 10, W is the overall number of pixels inside the respective detection bounding box ${}^{l,r}\mathcal{B}$ and $I_{(\cdot)}^w$ returns the value of image $I_{(\cdot)}$ at pixel w. The gradient and wireframe images are normalised such that $\sum_w^W I_{(\cdot)}^w = 1$. By not considering model parts whose backprojection falls outside $\mathcal{B}$ we reduce the misguiding effect of non-vehicle gradients. This likelihood becomes large when the backprojected wireframe corresponds well to large image gradients.

3.4.3 Inference: The objective function of eq. 4 is minimized to find the optimal pose and shape parameters for each detected vehicle. As this function is non-convex and the model parameters are continuous, we adapt the iterative Monte Carlo sampling procedure proposed in (Coenen et al., 2018) to approximate the parameter set for which the energy function becomes minimal. To this end we discretise the target parameters by generating model particles for the vehicle ASM. Starting from one or more initial parameter sets, we generate a number of particles n_p in each iteration $j \in [0, n_{it}]$ by jointly sampling the pose and shape parameters from a uniform distribution centered at the preceding parameter values. For the resampling step, we calculate the energy for every particle and introduce the best scoring particles as initial seed particles for the next iteration. In each iteration, the size of the interval from which the parameters are sampled is reduced. In the following paragraphs, more details are given.

Initialisation: For initialisation we introduce four initial model particles ${}^0\mathbf{s}_k^i = ({}^0\mathbf{t}_k, {}^0\theta_k^i, {}^0\gamma_k)$ with $i \in [1, 4]$ for every vehicle detection $\mathbf{v}_k$. To initialise the parameters of the particles we create the minimum 2D bounding box enclosing the 2D projections of the 3D vehicle points ${}^\Omega\mathbf{X}_k$ on the ground plane. We define the initial translation vector ${}^0\mathbf{t}_k$ as the bounding box centre. The particle orientations ${}^0\theta_k^i$ are set to the four orientations of the bounding box semi-axes. Due to this, in contrast to (Zia et al., 2013), we do not depend on good, and in fact not on any initial orientation estimates. The initial shape ${}^0\gamma_k$ is defined as the zero vector, thus, the initial particles correspond to the mean ASM.

Resampling: In each iteration j we want to find the n_s best scoring particles according to the particle energy in eq. 4 and forward these particles to the next iteration as seed particles. By forwarding multiple particles instead of only one we expect to be able to deal with multi-modal distributions and local minima in the objective function.

Refinement: We experienced the energy distribution to exhibit local minima at opposite directions of the vehicle orientations, i.e. when the model is rotated by about 180° compared to the real vehicle orientation, because some vehicles are almost symmetric w.r.t. their lateral axis. To overcome this problem we extend the minimisation procedure by a *refinement* step. In this step, an additional iteration is conducted after the last iteration n_{it} by introducing two particles as seed particles. The first particle is chosen as the one achieving the lowest energy within the particle set of the final iteration and the second particle is a copy of the first, rotated by 180°.

Final result: The final values for the target parameters of pose and shape are defined after the refinement step and are set to the parameters of the particle achieving the lowest energy within the particle set of the final refining iteration.

4. EVALUATION

4.1 Test Data and Test Setup

For the evaluation of our method we use stereo sequences of the KITTI Vision Benchmark Suite (Geiger et al., 2012). The data were captured by a mobile platform in an urban area. We use the object detection and object orientation estimation benchmark, which consists of 7481 stereo images with labelled objects. In our evaluation we consider all objects labelled as *car*. For every object, the benchmark provides 2D image bounding boxes, the 3D object location in model coordinates as well as the rotation angle about the vertical axis in model coordinates. According to the KITTI evaluation benchmark[1], we distinguish between three levels of difficulty (*easy*, *moderate* and *hard*) during evaluation, which mainly depend on the level of object occlusion and truncation. We require an overlap of at least 50% between the detected 2D bounding box and the reference bounding box for an object to be considered as a correct detection. The performance of the applied mRCNN is shown in Tab. 1, depicting the values for recall (percentage of reference vehicles that were detected), precision (percentage of detections that actually are vehicles) and F1 (harmonic mean of recall and precision). The detections are the input

[1] http://www.cvlibs.net/datasets/kitti

to our vehicle reconstruction approach. For the evaluation of the vehicle reconstruction, we consider all correctly detected vehicles and evaluate the reconstructed vehicle poses by comparing the 3D locations $\mathbf{t}_k$ and the orientation angles θ_k of our fitted models to the reference positions and orientations.

4.2 Parameter Settings

For the 3D reconstruction of the stereo images the maximum value δ_{σ_Z} for the standard deviation of the depth values is defined as 1.5 m. For the specific stereo setup used in Geiger et al. (2012), this leads to a maximum valid distance of the 3D points from the camera of approximately 24 m. We select the side length l_Φ of the free-space grid cells to be 25 cm. In all experiments for model fitting, we conduct $n_{it} = 12$ iterations, drawing 150 particles per iteration from $n_s = 8$ seed particles. The initial interval boundaries of the uniform distributions from which we randomly draw the particle parameters are ± 1.5 m for the location parameter $\mathbf{t}_k$, ± 2.5 for the shape parameter vector γ_k and $\pm 45°$ for the orientation θ_k. Choosing the range of $\pm 45°$ for the orientation angles of the four initial seed particles allows particles to take the whole range of possible orientations in the first iteration, which is important to be able to deal with incorrect initialisations. In each iteration j the size of the initially defined interval boundaries is decreased by a factor 0.85^j. With n_{it} = 12, this leads to a reduction of the final interval range to 14% of the initial width.

To assess the impact of the individual components in the model fitting procedure, we define different variants with different settings for the calculation of the energy function. In the **3D** variant, we only consider the 3D likelihood term for the model fitting by setting. We successively add additional likelihood terms to evaluate their individual impact. In variant **3D+F**, we add the free-space likelihood term to the energy function. With this setting we are able to evaluate the results that can be achieved by only using 3D data neglecting image information. In contrast to that, we only consider image information neglecting all 3D information in the **Img** setting by only considering the classification and gradient likelihood terms. However, 3D information is still used for the initialisation. In **3D+F+C**, we apply the classification likelihood term together with the two likelihood terms based on 3D information. To evaluate the complete energy function for the model fitting we apply the setting **Full** by leveraging all likelihood terms. In all variants listed so far, we do not apply the refinement step described in Sec. 3.4.3. The impact of this step is analysed in the last variant, referred to as **Refine**. Here, we apply the settings of the *Full* setup and additionally conduct the *refinement* step at the end. In addition to the reconstruction results we also report the values for the initial poses in **Init**.

4.3 Vehicle Reconstruction Results

To evaluate the vehicle reconstructions, we compare the resulting pose parameters from each fitted 3D vehicle model and the reference data for location and orientation of the vehicles. We consider a model to be correct in position if its distance from the reference position is smaller than $0.75\,m$. To evaluate the orientation, we report values in three stages (θ_5, θ_{10} and $\theta_{22.5}$), in which we consider an orientation to be correct if its difference from the reference is less than $5°$, $10°$ and $22.5°$, respectively. Tab. 2 contains the percentage of the correctly estimated pose parameters for the different variants described in Sec. 4.2 including the absolute mean error ε for the *Refine* variant. Comparing the results for different levels of difficulty, we can see a similar pattern of performance for all variants. That is, all variants perform best for the *easy* level and worst for the *hard* level. While the amount of correct position estimations decreases only by 8.1 % at maximum from *easy* to *hard*, the results for orientation are more sensitive to the degree of vehicle visibility, as they decrease by up to 14.6 %. Further, independently from the level of difficulty, the percentage of vehicles for which a correct position is determined only differs by about 2 % between the different approaches (excluding the *Img* setting). Again, the more sensitive parameter is the vehicle orientation. Thus, in the following paragraphs we will focus on the analysis of the orientation results. Fig. 4 shows the cumulative histogram of absolute orientation errors over the full range of 0-180° whereas in Fig. 5 we depict a histogram of absolute orientation differences for eight discrete orientation bins, both exemplarily for the *easy* level.

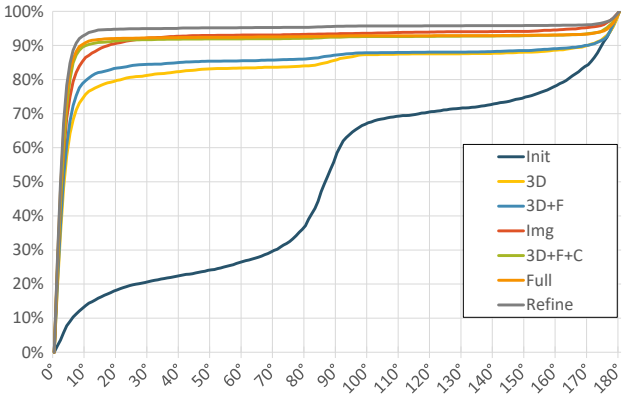


Figure 4. Cumulative histogram of absolute differences between estimated and reference orientations (*easy* level).

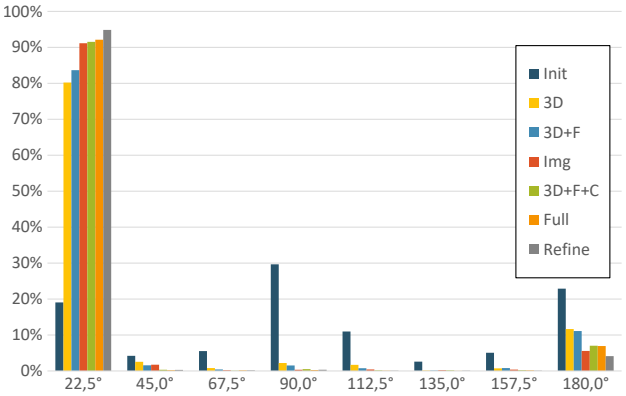


Figure 5. Histogram of absolute differences between estimated and reference orientations (*easy* level).

Init: As the quality values for the initial poses show, the initialisation in general can only be regarded as very inaccurate or even wrong, especially related to the orientations. However, as our results show, our proposed approach is quite robust against these initialisation errors as it is able to correct the majority of them.
3D: We choose the *3D* setting as baseline and achieve results with up to 64.2 % of orientation estimations whose error is smaller than 5° and 80.2 % with an error smaller than 22.5°for the *easy* category; lower results are achieved for the more challenging categories. Fig. 5 shows that the incorrect orientation estimations for this setting mainly fall into the 180° bin.
3D+F: Incorporating the free-space energy term for the vehicle reconstruction generally increases the number of correct orientation estimations by up to 4.1 %. An interesting observation can be made from Fig. 5: Considering free-space leads to a reduction of

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-IV-2-W5-21-2019 | © Authors 2019. CC BY 4.0 License.

	easy	**mod.**	**hard**
Recall [%]	98.6	95.9	85.5
Precision [%]	96.7	97.9	98.3
F1 [%]	97.7	96.9	91.5

Table 1. Vehicle detection results.

		Init	3D	3D+F	Img	3D+F+C	Full	Refine (ε)
easy	**t** [%]	65.6	81.7	82.0	61.6	81.7	81.0	80.8 (*0.33 m*)
	θ_5 [%]	9.1	64.2	67.8	75.0	80.1	82.6	84.8 (*1.9°*)
	θ_{10} [%]	13.7	75.6	79.7	86.5	90.0	90.8	93.2 (*2.3°*)
	$\theta_{22.5}$[%]	19.1	80.2	83.7	91.1	91.5	92.1	94.8 (*2.5°*)
moderate	**t** [%]	58.9	79.5	80.4	59.7	80.8	80.5	80.6 (*0.33 m*)
	θ_5 [%]	8.6	59.4	61.8	67.9	73.1	75.2	77.8 (*1.8°*)
	θ_{10} [%]	13.1	70.0	72.7	78.2	82.8	83.8	86.4 (*2.3°*)
	$\theta_{22.5}$[%]	18.5	75.1	77.3	84.0	86.0	86.3	89.0 (*2.7°*)
hard	**t** [%]	51.5	73.6	74.9	54.6	76.0	75.4	75.9 (*0.34 m*)
	θ_5 [%]	8.4	53.7	55.8	61.0	66.3	68.0	70.4 (*1.8°*)
	θ_{10} [%]	13.0	63.2	65.6	70.4	75.3	76.1	78.4 (*2.3°*)
	$\theta_{22.5}$[%]	18.1	68.2	70.3	76.5	79.0	78.8	81.6 (*2.8°*)

Table 2. Evaluation of the pose estimation results.

incorrect orientation estimations mainly in the intermediate bins while the number of orientation results in the first orientation bin increases. We consider this as the desired and natural effect of the free-space energy term. The values for this variant represent the results that can be achieved by only considering the 3D information derived from the stereo images.

Img: In contrast to the *3D+F* setting, the *Img* setting represents the results that can be achieved by only using the 2D image cues for the vehicle reconstruction. According to Tab. 2, neglecting 3D information causes a major loss of correct position estimations (61.6 % compared to 82.0 % by the *3D+F* setting for the easy case). Also, the image likelihood terms seem to be strongly beneficial for estimating the vehicle orientations. Thus, up to 7.2 % (θ_5, easy case) more correct orientation estimations can be achieved by only using image information compared with only considering 3D information. Fig. 5 shows that the image likelihood terms mainly help to distinguish between opposite vehicle heading directions as the number of orientation differences in the last orientation bin is remarkably lower compared to the *3D+F* setting. In the following settings, the effect of combining 3D and 2D information for the vehicle reconstruction is analysed.

3D+F+C: According to Tab. 2, incorporating the classification likelihood term together with the two 3D likelihood terms yields the largest improvements for the orientation estimation by decreasing the number of incorrect orientation estimations by up to 12.3 % (for the θ_5 criterion). Fig. 5 shows that the classification energy term helps to reduce false estimations of both, the last as well as the intermediate orientation bins.

Full: Additionally considering the gradient likelihood term to the objective function, small improvements of up to 2.5 % for the θ_5 criterion can be achieved.

Refine: It is the goal of the refinement step to reduce the number of orientation estimations that differ from the reference by 180°. Fig. 5 shows that the desired effect can be achieved to a certain degree. Introducing seed particles with opposite viewing directions compared to its preceding particle decreases the errors in the last orientation bin drastically. In total, the refinement step yields improvements of up to 2.7 %. Hence, regarding the orientation estimation, the distinctly best results with up to 84.8 % and even 94.8 % of correct orientation estimations w.r.t. the θ_5 and the $\theta_{22.5}$ criterion are achieved by applying our *Full* objective function including the conclusive refinement step. Further, we are able to achieve very precise pose estimations with maximum average errors of 2.8° for orientation and 34 cm for position.

For collaborative positioning tasks, the estimation of the orientation is the more sensitive factor: for an exemplary vehicle distance of 10 m, an orientation error of 2.8° leads to around 50 cm error in positioning compared to the 34 cm resulting from the position estimation error. Using the *Refine* setting we are able to outperform the orientation estimation results of Coenen et al. (2018) by 12.4 % and to achieve an equivalent amount of correct orientations as in Zia et al. (2013) but with a 1.5° smaller average error and without depending on good pose initialisations. Though, it has to be noted that in Zia et al. (2013) no stereo information was used. Fig. 6 shows some qualitative results of our vehicle reconstruction approach.

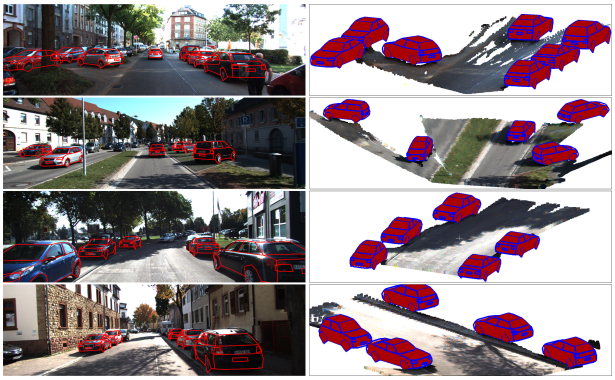


Figure 6. Qualitative results. *Left:* Backprojected model wireframes. *Right:* 3D view on the reconstructed scene; estimated ground plane and the 3D vehicle models are shown

5. CONCLUSION

We developed an approach for shape and pose aware vehicle reconstruction from street-level stereo images. We were able to show the benefits of the different constituents of our objective function w.r.t. the pose estimation results: While the *classification likelihood* term leads to the largest improvements, the *gradient likelihood* term is able to reduce incorrect orientation estimations in the intermediate orientation bins and the *refinement step* in the last bin, respectively. Until now, the stereo image pairs are processed individually. In the future, the reconstruction framework will be extended to multiple image pairs of subsequent time steps, thus fitting a vehicle model to observations from different epochs simultaneously. This might help to reduce the overall computational cost by incorporating prior estimation results of previous epochs, using a reasonable vehicle motion model. Further, in the future we will make use of the shape estimation results to reason about vehicle categories, the vehicle type and to recognize individual vehicles.

ACKNOWLEDGEMENTS

This work was supported by the German Research Foundation (DFG) as a part of the Research Training Group i.c.sens [GRK2159].

REFERENCES

Ansari, J. A., Sharma, S., Majumdar, A., Murthy, J. K. and Krishna, K. M., 2018. The Earth ain't Flat: Monocular Reconstruction of Vehicles on Steep and Graded Roads from a Moving Camera. In: *IEEE International Conference on Intelligent Robots and Systems (IROS)*, pp. 8404–8410.

Bao, S. Y., Chandraker, M., Lin, Y. and Savarese, S., 2013. Dense Object Reconstruction with Semantic Priors. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1264–1271.

Braden, B., 1986. The Surveyor's Area Formula. *The College Mathematics Journal* 17(4), pp. 326–337.

Breiman, L., 2001. Random Forests. *Machine Learning* 45, pp. 5–32.

Chabot, F., Chaouch, M., Rabarisoa, J., Teuliere, C. and Chateau, T., 2017. Deep MANTA: A Coarse-to-Fine Many-Task Network for Joint 2D and 3D Vehicle Analysis from Monocular Image. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1827–1836.

Chen, X., Kundu, K., Zhu, Y., Berneshawi, A. G., Ma, H., Fidler, S. and Urtasun, R., 2015. 3d Object Proposals for accurate Object Class Detection. In: *Advances in Neural Information Processing Systems*, Vol. 28, pp. 424–432.

Coenen, M., Rottensteiner, F. and Heipke, C., 2018. Recovering the 3D Pose and Shape of Vehicles from Stereo Images. In: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. IV-2, pp. 73–80.

Cootes, T. F., Taylor, C. J. and Cooper, D. H., 1995. Active Shape Models - Their Training and Application. *Computer Vision and Image Understanding (CVIU)* 61(1), pp. 38–59.

Dalal, N. and Triggs, B., 2005. Histograms of oriented Gradients for Human Detection. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Vol. 1, pp. 886–893.

Dame, A., Prisacariu, V. A., Ren, C. Y. and Reid, I., 2013. Dense Reconstruction using 3D Object Shape Priors. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1288–1295.

Engelmann, F., Stückler, J. and Leibe, B., 2016. Joint Object Pose Estimation and Shape Reconstruction in urban Street Scenes using 3D Shape Priors. In: *Pattern Recognition*, Lecture Notes in Computer Science, Vol. 9796, pp. 219–230.

Felzenszwalb, P. F., Girshick, R. B., McAllester, D. and Ramanan, D., 2010. Object Detection with discriminatively trained part-based Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32(9), pp. 1627–1645.

Geiger, A., Lenz, P. and Urtasun, R., 2012. Are we ready for autonomous Driving? The KITTI Vision Benchmark Suite. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3354–3361.

Geiger, A., Roser, M. and Urtasun, R., 2011. Efficient Large-Scale Stereo Matching. In: *Computer Vision – ACCV 2010*, Lecture Notes in Computer Science, Vol. 6492, pp. 25–38.

Güney, F. and Geiger, A., 2015. Displets: Resolving stereo Ambiguities using Object Knowledge. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4165–4175.

He, K., Gkioxari, G., Dollár, P. and Girshick, R., 2017. Mask r-cnn. In: *IEEE International Conference on Computer Vision (ICCV)*, pp. 2980–2988.

Kundu, A., Li, Y. and Rehg, J. M., 2018. 3D-RCNN: Instance-level 3D Object Reconstruction via Render-and-Compare. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3559–3568.

Leibe, B., Leonardis, A. and Schiele, B., 2006. *An Implicit Shape Model for combined Object Categorization and Segmentation*. Lecture Notes in Computer Science, Vol. 4170, Springer Berlin Heidelberg, pp. 508–524.

Liebelt, J. and Schmid, C., 2010. Multi-view Object Class Detection with a 3D geometric Model. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1688–1695.

Lin, Y.-L., Morariu, V. I., Hsu, W. and Davis, L. S., 2014. Jointly Optimizing 3D Model Fitting and Fine-Grained Classification. In: *European Conference on Computer Vision (ECCV)*, pp. 466–480.

Menze, M., Heipke, C. and Geiger, A., 2015. Joint 3d Estimation of Vehicles and Scene Flow. In: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. II-3, pp. 427–434.

Mousavian, A., Anguelov, D., Flynn, J. and Koseck, J., 2017. 3D Bounding Box Estimation using Deep Learning and Geometry. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5632–5640.

Murthy, J. K., Krishna, G. V. S., Chhaya, F. and Krishna, K. M., 2017. Reconstructing Vehicles from a single Image: Shape Priors for Road Scene Understanding. In: *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 724–731.

Ozuysal, M., Lepetit, V. and Fua, P., 2009. Pose Estimation for Category specific multiview Object Localization. In: *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPR Workshops)*, pp. 778–785.

Payet, N. and Todorovic, S., 2011. From Contours to 3D Object Detection and Pose Estimation. In: *IEEE International Conference on Computer Vision (ICCV)*, pp. 983–990.

Pepik, B., Stark, M., Gehler, P. and Schiele, B., 2012. Teaching 3D Geometry to deformable Part Models. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3362–3369.

Thomas, A., Ferrari, V., Leibe, B., Tuytelaars, T. and van Gool, L., 2007. Depth-From-Recognition: Inferring Meta-data by Cognitive Feedback. In: *IEEE 11th International Conference on Computer Vision (ICCV)*, pp. 1–8.

Tulsiani, S. and Malik, J., 2015. Viewpoints and Keypoints. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1510–1519.

Villamizar, M., Grabner, H., Moreno-Noguer, F., Andrade-Cetto, J., van Gool, L. and Sanfeliu, A., 2011. Efficient 3D Object Detection using multiple Pose-Specific Classifiers. In: *British Machine Vision Conference*, pp. 20.1–20.10.

Wang, Y., Tan, X., Yang, Y., Liu, X., Ding, E., Zhou, F. and Davis, L. S., 2018. 3D Pose Estimation for Fine-Grained Object Categories. In: *European Conference on Computer Vision Workshops (ECCVW)*.

Xiang, Y., Choi, W., Lin, Y. and Savarese, S., 2017. Subcategory-Aware Convolutional Neural Networks for Object Proposals and Detection. In: *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 924–933.

Xiao, W., Vallet, B., Schindler, K. and Paparoditis, N., 2016. Street-Side Vehicle Detection, Classification and Change Detection using mobile Laser Scanning Data. *ISPRS Journal of Photogrammetry and Remote Sensing* 114, pp. 166–178.

Zia, M. Z., Stark, M., Schiele, B. and Schindler, K., 2013. Detailed 3D Representations for Object Recognition and Modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35(11), pp. 2608–2623.