

# USING LABEL NOISE ROBUST LOGISTIC REGRESSION FOR AUTOMATED UPDATING OF TOPOGRAPHIC GEOSPATIAL DATABASES

A. Maas<sup>\*</sup>, F. Rottensteiner, C. Heipke

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

Commission VII, WG VII/5

**KEY WORDS:** Change detection, label noise, logistic regression, supervised classification

## ABSTRACT:

Supervised classification of remotely sensed images is a classical method to update topographic geospatial databases. The task requires training data in the form of image data with known class labels, whose generation is time-consuming. To avoid this problem one can use the labels from the outdated database for training. As some of these labels may be wrong due to changes in land cover, one has to use training techniques that can cope with wrong class labels in the training data. In this paper we adapt a label noise tolerant training technique to the problem of database updating. No labelled data other than the existing database are necessary. The resulting label image and transition matrix between the labels can help to update the database and to detect changes between the two time epochs. Our experiments are based on different test areas, using real images with simulated existing databases. Our results show that this method can indeed detect changes that would remain undetected if label noise were not considered in training.

## 1. INTRODUCTION

Topographical databases are very important for applications such as navigation or city planning. Keeping such a database up-to-date manually has been estimated to require up to 40% of the costs for the original data acquisition (Champion, 2007), which indicates that this process should be automated. In this context, the primary data source most frequently used is remotely sensed imagery. If the sensor data used for the original data acquisition are not available, a typical work flow for automated updating of topographic databases starts with the classification of the new sensor data. In a second step, the classification results are compared with the database in order to detect areas of change e.g. (Vosselman et al., 2004). Based on the detected changes, the database can then be updated. For the first step, supervised classification algorithms are frequently used because they are more easily transferred from one data set to another one. The reason for this is that supervised methods rely on representative training data to train the underlying classifier, thus adapting it to changes in the appearance of the objects.

This flexibility comes at the cost that training data, consisting of image subsets with known object labels, have to be generated in advance, typically in a time-consuming and costly manual process. Thus, it would be desirable to reduce the amount of training data required. One strategy to achieve this aim is to use the existing database to provide the necessary class labels. Such a procedure has to take into account that the database may be outdated, so that some of the class labels derived from the original map might be wrong. However, in general changes will only affect a relatively small part of a scene, so that one can assume the majority of the class labels to be correct.

In this paper we propose a new supervised classification method that uses existing database information for training. Unlike most of the existing work in this context, we do not just eliminate

training samples having a wrong label as outliers, but we resort to a training method that is tolerant to these errors. In particular, we use a label noise tolerant version of logistic regression (Bootkrajang & Kabán, 2012), a probabilistic method that does not only reduce the impact of label noise in the training process, but also delivers an estimate about the amount of change in a scene. After applying the resultant classifier to the new sensor data, we can compare the classification results to the original database and, thus, obtain change in land cover. No manually labelled training data are required. Our method is evaluated using several data sets with different degrees of simulated changes to show the benefits, but also the limitations of the proposed method.

## 2. RELATED WORK

The detection of changes between a current image and an existing database is a basic step in the updating process. The overview in (Jianya et al., 2008) distinguishes three basic strategies for change detection. The first group of methods compares the image data of two epochs directly, based on features such as band ratios, to detect changes, e.g. (Subudhi et al., 2014). The second group of methods compares the results of an independent classification of the images from both epochs, whereas the third and most general group of methods integrates all known data simultaneously for multitemporal classification. In a probabilistic context, this leads to models such as Markov chains, where transition probabilities between epochs are considered, potentially in combination with a local context model, e.g. (Hoberg et al., 2015). As we assume sensor data to be unavailable for the time of the original database acquisition, our approach is based on the second strategy in this paper; thus, we compare the classification results of the data acquired at the second epoch to the original database. To obtain the training samples required for good classification and change detection results, the database is used. Such derived training data are

---

<sup>\*</sup> Corresponding author

affected by gross errors called *label noise*, which has to be considered in the training process.

Training under label noise is a well-studied problem, for example in fields such as epidemiology, econometrics and computer-aided diagnoses. Fréney and Verleysen (2014) differentiate three types of statistical models for label noise. The *noisy completely at random* (NCAR) model assumes the occurrence of a label error to be independent from all other variables, including the class label. It is characterized by a single parameter, namely the probability of an error. In the *noisy at random* (NAR) model, the probability of an error depends on the class labels, so that it is parameterized by a square transition matrix whose elements describe the probability for a specific type of error affecting two classes. The most complex model is called *noisy not at random* model and additionally considers dependencies between labelling errors and the observed data.

According to Fréney and Verleysen (2014), there are three strategies for dealing with label noise. The first one is to use a classifier that is robust to label noise by design, e.g. random forests. However, such methods still have problems with a large amount of label noise. The second strategy tries to identify the incorrect training samples in order to remove them from the training set before the actual training procedure. The authors state that such *data cleansing* approaches tend to eliminate too many instances, which may lead to a decreased classification performance. Finally, the third strategy consists of learning algorithms that are tolerant to noisy training data. In this context probabilistic approaches can be distinguished from non-probabilistic ones. Probabilistic models learn the parameters of a noise model, e.g. the elements of the transition matrix for the NAR model, jointly with the parameters of a classifier that would best separate the data based on the (unknown) true labels of the training samples. This strategy is followed by Bootkrajang and Kabán (2012), using logistic regression as a base classifier. Li et al. (2007) achieve a similar result on the bases of a kernel Fisher discriminant. Bootkrajang and Kabán (2012) report results for the classification of entire images with the purpose of image revival, but not for a classification on a pixel level. Non-probabilistic methods focus on making non-probabilistic classifiers such as support vector machines (SVM) tolerant to label noise (An & Liang, 2013), but typically do not estimate the parameters of a noise model.

To the knowledge of the authors most existing approaches for considering label noise in remote sensing are based on data cleansing. For instance, Radoux et al. (2014), deriving training data from an existing map, present two techniques for eliminating outliers. The first one removes training samples near the boundaries of land cover types, whereas the other one assumes a Gaussian distribution of spectral signatures and removes outliers based on a statistical test. These methods seem to be tailored to data of low ground sampling distance (300m). It is doubtful whether the model assumptions can be transferred to high resolution data, where each class may correspond to multiple clusters in feature space. A similar method was used for map updating in (Radoux & Defourmy, 2010), using Kernel density estimation for deriving probability densities. The error rates in the original data were relatively low. Jia et al. (2014) use all pixels from an existing map for training and eliminate samples that receive another class than the one indicated in the original data. Büschenfeld (2013) uses land cover data from a geographical information system (GIS) for generating training samples. He iteratively applies SVM, eliminating training samples that are assigned to another class than indicated by the

observed label or that show a high uncertainty. The entities to be classified are land cover objects from the GIS.

Mnih and Hinton (2012) are among the few authors using maps for label noise tolerant training. Their method is based on deep learning, but they only present a solution for a binary classification problem. Bruzzone and Persello (2009) propose a context-sensitive semi-supervised SVM, which is supposed to be robust to label noise. This improvement is realized by including information of the pixels in the neighbourhood of the training samples in the learning process. However, no probabilistic label noise model is used. The authors claim that such a strategy allows the use of existing maps for training, but this topic is not elaborated further.

This paper presents a new method for change detection between an outdated database and current remotely sensed images without using manually labelled training data, just relying on the database. Unlike most existing work, we apply label noise tolerant classification to cope with incorrect training labels. In particular, we apply label noise tolerant logistic regression (Bootkrajang & Kabán, 2012), a probabilistic method that also provides estimates for the parameters of a NAR model which can be interpreted as probabilities for specific types of change. In order to consider local context, we integrate this classifier in a Conditional Random Field (CRF) model. The resulting label image is compared with the database to detect changes. We evaluate our method for different degrees of label noise and compare the results to those achieved by random forests.

### 3. LABEL NOISE TOLERANT CHANGE DETECTION

#### 3.1 General Idea

For our task we assume the existing database and the image data to be available in raster format, defined on the same grid. The data consist of  $N$  pixels, each pixel  $n$  represented by a feature vector  $\mathbf{x}_n = [x_n^1, \dots, x_n^F]^T$  of dimension  $F$ . The existing database contains an *observed class label*  $C_n^o \in \mathbb{C} = \{C^1, \dots, C^K\}$  for each pixel  $n$ , where  $\mathbb{C}$  denotes the set of classes and  $K$  is the total number of classes. As the database may be outdated, the observed label may differ from the unknown current label  $C_n \in \mathbb{C}$  of that pixel that corresponds to the updated database information and has to be estimated from the data. Changes are determined implicitly as pixels where the observed label differs from the current one ( $C_n^o \neq C_n$ ).

The current class labels are determined in a supervised classification of the remote sensing data. Our classification method is based on a CRF. In this CRF, we use logistic regression for the association potentials. The core of our method and our main contribution is that we use the observed class labels as the labels of the training data and deal with the problem of wrong training labels under the assumption that they form a specific kind of label noise. Thus, in principle we can use all pixels as training data, though we have to select a subset of them for computational efficiency. The training procedure itself is the one proposed by Bootkrajang and Kabán (2012). It delivers the parameters of logistic regression and an estimate for the parameters of a model for label noise, namely the NAR model. This procedure is described in Section 3.2. In the CRF-based classification itself, the logistic regression classifier is combined with a model for local context to achieve a smooth classification result. Details are explained in Section 3.3.

### 3.2 Label Noise Tolerant Logistic Regression

Multiclass logistic regression is a probabilistic discriminative classifier that directly models the posterior probability  $p(C_n|\mathbf{x}_n)$  of a class label  $C_n$  given the feature vector  $\mathbf{x}_n$ . A feature space transformation  $\Phi(\mathbf{x}_n)$  is applied to achieve non-linear decision boundaries in the original feature space. That is, rather than to  $\mathbf{x}_n$ , the classification is applied to a vector  $\Phi(\mathbf{x}_n)$  which has a higher dimension than  $\mathbf{x}_n$  and whose components may be arbitrary functions of  $\mathbf{x}_n$ . For instance, one can use quadratic expansion, i.e.  $\Phi(\mathbf{x}_n)$  contains the original features as well as all squares and mixed products of features. In addition,  $\Phi(\mathbf{x}_n)$  contains a bias feature that is assumed to be a constant with value 1 without loss of generality. The model of the posterior is based on the softmax function (Bishop, 2006):

$$p(C_n = C^k|\mathbf{x}_n) = \frac{\exp(\mathbf{w}_k^T \cdot \Phi(\mathbf{x}_n))}{\sum_{j=1}^K \exp(\mathbf{w}_j^T \cdot \Phi(\mathbf{x}_n))}, \quad (1)$$

where  $\mathbf{w}_k$  is a vector of weight coefficients for a particular class  $C^k$  that is related to the parameters of the separating hyperplanes in the transformed feature space. As the sum of the posterior over all classes has to be 1, these weight vectors are not independent. This fact is considered by setting  $\mathbf{w}_1$  to  $\mathbf{0}$ . The remaining weights are collected in a joint parameter vector  $\mathbf{w}=(\mathbf{w}_2^T, \dots, \mathbf{w}_K^T)^T$  to be determined from training data.

It is our goal to train a classifier that directly delivers the current labels  $C_n$ . However, our training data consist of  $N$  independent pairs  $(\mathbf{x}_n, C_n^o)$  of a feature vector and the corresponding observed class label from the existing database. In order to determine the most probable values of  $\mathbf{w}$ , we have to optimise the posterior of  $\mathbf{w}$  given the training data (Bishop, 2006):

$$p(\mathbf{w}|\mathbf{x}_1, \dots, \mathbf{x}_N, C_1^o, \dots, C_N^o) \propto p(\mathbf{w}) \cdot \prod_{n,k} p(C_n^o = C^k|\mathbf{x}_n, \mathbf{w}) \quad (2)$$

In the presence of label noise, the observed label  $C_n^o$  is not necessarily the label  $C^k$  which should be determined by maximising the posterior in eq. 1. Bootkrajang and Kabán (2012) propose to determine the probability  $p(C_n^o = C^k|\mathbf{x}_n, \mathbf{w})$  required for training as the marginal distribution of the observed labels  $C_n^o$  over all possible states of the unknown current labels  $C_n$ . This leads to (Bootkrajang & Kabán, 2012):

$$p(C_n^o = C^k|\mathbf{x}_n, \mathbf{w}) = \sum_a p(C_n^o = C^k|C_n = C^a) \cdot p(C_n = C^a|\mathbf{x}_n, \mathbf{w}) \quad (3)$$

or  $S_{nk} = \sum_a \gamma_{na} \cdot f_{na}$ , where we introduced the short-hands  $S_{nk} = p(C_n^o = C^k|\mathbf{x}_n, \mathbf{w})$ ,  $\gamma_{na} = p(C_n^o = C^k|C_n = C^a)$  and  $f_{na} = p(C_n = C^a|\mathbf{x}_n, \mathbf{w})$ .

In eq. 3  $f_{na} = p(C_n = C^a|\mathbf{x}_n, \mathbf{w})$  is identical to the posterior for the unknown current label  $C_n$ , modelled according to eq. 1. The probabilities  $p(C_n^o = C^k|C_n = C^a)$  are the parameters of the NAR model according to (Frénay & Verleysen, 2014), describing how probable it is to observe label  $C^k$  if the true label indicated by the feature vector is  $C^a$ . From the point of view of change detection, these probabilities are closely related to the probability of a change from class  $C^k$  to  $C^a$ , though the direction of change according to the definition in eq. 3 is actually inverted. If we differentiate  $K$  classes, there are, consequently,  $K \times K$  such transition probabilities, which we can collect in a  $K \times K$  transition matrix  $\Gamma$  with  $\Gamma(a,k) = \gamma_{ak}$ .

We use a Gaussian prior with zero mean and isotropic covariance  $\sigma \cdot \mathbf{I}$ , where  $\mathbf{I}$  is a unit matrix, for the regularisation term  $p(\mathbf{w})$  in eq. 2. Finding the maximum of eq. 2 is equivalent

to finding the minimum of the negative logarithm, thus of  $E(\mathbf{w}) = -\ln[p(\mathbf{w}|\mathbf{x}_1, \dots, \mathbf{x}_N, C_1^o, \dots, C_N^o)]$ . Plugging eq. 3 into eq. 2 and taking the negative logarithm yields

$$E(\mathbf{w}) = -\sum_{n=1}^N \sum_{k=1}^K t_{nk} \cdot \ln S_{nk} + \frac{\mathbf{w}^T \cdot \mathbf{w}}{2\sigma^2} \rightarrow \min, \quad (4)$$

where  $t_{nk}$  is an indicator variable taking the value 1 if  $C_n^o = C^k$  and 0 otherwise. As  $E(\mathbf{w})$  is nonlinear, minimisation has to be carried out iteratively. Starting from initial values  $\mathbf{w}^0$ , we apply gradient descent, estimating the parameter vector  $\mathbf{w}^\tau$  in iteration  $\tau$  using the Newton-Raphson method (Bishop, 2006):

$$\mathbf{w}^\tau = \mathbf{w}^{\tau-1} - \mathbf{H}^{-1} \nabla E(\mathbf{w}), \quad (5)$$

where  $\nabla E(\mathbf{w}) = [\nabla_{\mathbf{w}_2} E(\mathbf{w})^T, \dots, \nabla_{\mathbf{w}_K} E(\mathbf{w})^T]^T$  is the gradient vector and  $\mathbf{H}$  is the Hessian matrix. The gradient is the concatenation of all derivatives by the class-specific parameter vectors  $\mathbf{w}_j$ , with (Bootkrajang & Kabán, 2012)

$$\nabla_{\mathbf{w}_j} E(\mathbf{w}) = \sum_{n=1}^N (f_{nj} - \tilde{t}_{nj}) \cdot \Phi(\mathbf{x}_n) + \frac{1}{\sigma^2} \cdot \mathbf{w}, \quad (6)$$

where  $\tilde{t}_{nj} = f_{nj} \cdot \sum_{k=1}^K (\gamma_{jk} \cdot \frac{t_{nk}}{S_{nk}})$ . The Hessian Matrix consists of  $(K-1) \times (K-1)$  blocks  $\mathbf{H}_{ij} = \nabla_{\mathbf{w}_i} \nabla_{\mathbf{w}_j} E(\mathbf{w})$  with (Bootkrajang & Kabán, 2012)

$$\mathbf{H}_{ij} = \sum_{n=1}^N (f_{ni} \cdot f_{nj} \cdot \xi + I_{ij} \cdot (f_{nj} - \tilde{t}_{nj})) \cdot \Phi(\mathbf{x}_n) \cdot \Phi(\mathbf{x}_n)^T + \frac{\delta(i=j)}{\sigma^2} \cdot \mathbf{I}, \quad (7)$$

where  $\mathbf{I}$  is a unit matrix with elements  $I_{ij}$ ,  $\delta(\cdot)$  is the Kronecker delta function delivering a value of 1 if the argument is true and 0 otherwise, and

$$\xi = \sum_{k=1}^K \left( \gamma_{jk} \cdot \gamma_{ik} \cdot \frac{t_{nk}}{S_{nk}} \right) - 1. \quad (8)$$

Optimising for the unknown weights by gradient descent as just described requires knowledge about the elements of the transition matrix  $\Gamma$ , i.e. the parameters of the noise model. However, these parameters are unknown. Bootkrajang and Kabán (2012) propose an iterative procedure similar to expectation maximisation (EM). Starting from initial values for  $\Gamma$ , e.g. based on the assumption that there is not much change (leading to large values on the main diagonal only), the optimal weights can be determined. Using these weights, one can update the parameters of  $\Gamma$  according to (Bootkrajang & Kabán, 2012)

$$\gamma_{jk}^{new} = \frac{1}{c} \cdot \gamma_{jk}^{old} \cdot \sum_{n=1}^N \frac{t_{nk}}{S_{nk}} \cdot \gamma_{nj}, \quad (9)$$

$$\text{where } c = \sum_{l=1}^K \left[ \gamma_{jl}^{old} \cdot \sum_{n=1}^N \left( f_{nj} \cdot \frac{t_{nl}}{S_{nl}} \right) \right].$$

These values for the transition matrix can be used for an improved estimation of  $\mathbf{w}$ , and so forth. Thus, the update of  $\mathbf{w}$  and  $\Gamma$  alternate until a stopping criterion, e.g. related to the change of the parameters between two consecutive iterations, is reached. Note that the parameters  $\mathbf{w}$  thus obtained are related to a classifier delivering the posterior for the unknown current labels  $C_n$  (eq. 1).

### 3.3 The Conditional Random Field (CRF)

CRF are graphical models which can be used to consider local context in a probabilistic framework (Kumar & Hebert, 2006). In our application, the nodes of the graphical models are the

current labels  $C_n$  introduced in Section 3.1 that correspond to the updated database information. The edges of the graph model dependencies between the random variables corresponding to the nodes. We connect direct neighbours on the basis of a 4-neighbourhood of the image grid by edges. Rather than classifying each pixel  $n$  of an image individually based on locally observed features, the entire configuration of labels, collected in a vector  $\mathbf{C} = (C_1, \dots, C_N)^T$  is determined simultaneously using all the observed data  $\mathbf{x}$ , i.e. all features observed at the individual pixels. CRF are discriminative models, so that the posterior  $P(\mathbf{C}|\mathbf{x})$  for the entire classified image  $\mathbf{C}$  given the data  $\mathbf{x}$  is modelled directly according to (Kumar & Hebert, 2006):

$$P(\mathbf{C}|\mathbf{x}) = \frac{1}{Z} \exp \left[ \sum_n A(C_n, \mathbf{x}) + \sum_{n,m \in \epsilon} I(C_n, C_m, \mathbf{x}) \right] \quad (10)$$

In eq. 10,  $Z$  is a normalization constant which is not considered further in the classification process because we are only interested in determining the label configuration  $\mathbf{C}$  for which  $P(\mathbf{C}|\mathbf{x}) \rightarrow \max$ . The association potential  $A$  is the link between the data  $\mathbf{x}$  and the label  $C_n$  of pixel  $n$ .  $A(C_n, \mathbf{x})$  may depend on the entire input image  $\mathbf{x}$ , which is often considered by using site-wise feature vectors  $\mathbf{x}_n(\mathbf{x})$  that may be functions of the entire image. Any discriminative classifier can be used in this context; we use logistic regression, thus  $A(C_n, \mathbf{x}) = \ln p(C_n|\mathbf{x}_n)$ , where  $p(C_n|\mathbf{x}_n)$  is determined according to eq. 1. The definition of  $\mathbf{x}_n = \mathbf{x}_n(\mathbf{x})$  depends on the available data.

The terms  $I(C_n, C_m, \mathbf{x})$  are called the interaction potentials; they describe the context model. The sum over the interaction potentials is taken over all pairs of pixels  $n, m$  connected by an edge; thus,  $\epsilon$  is the set of edges in the graph. The interaction potentials also depend the data  $\mathbf{x}$ . We use the context-sensitive Potts model for the interaction potential, which results in a data-dependant smoothing of the resultant image (Shotton et al., 2009):

$$I(C_n, C_m, \mathbf{x}) = \delta(C_n, C_m) \cdot \left( \beta_0 + \beta_1 \cdot \exp \left( - \frac{\|\mathbf{x}_n - \mathbf{x}_m\|^2}{2\sigma_D^2} \right) \right) \quad (11)$$

Again,  $\delta(\cdot)$  is the Kronecker delta function, whereas the coefficients  $\beta_0$  and  $\beta_1$  influence the overall degree of smoothing and the impact of the data-dependant term, respectively. The parameter  $\sigma_D$  is the average squared gradient of the image.

We train the association potentials independently from the interaction terms, using the method described in Section 3.2. The parameters of the interaction potentials ( $\beta_0$  and  $\beta_1$ ) could be determined by a procedure such as cross-validation, but we use values found empirically. For the determination of the optimal configuration of labels given the model of the posterior we use loopy belief propagation (Frey & MacKay, 1998).

## 4. EXPERIMENTS

### 4.1 Test Data and Test Setup

We use three data sets in our experiments. The first dataset consists of a part of the Vaihingen data contained in the ISPRS 2D Labelling Challenge (Wegner et al., 2015). We use eleven of the patches provided for the test, each about 2,000 x 2,500 pixels in size. For each patch, a colour infrared true orthophoto (TOP) and a digital surface model (DSM) are made available, both with a ground sampling distance of 9 cm. Furthermore, all

the patches used in this paper belong to the training set of the labelling challenge, so that reference data are available in the same grid as the other data. The reference differentiates the six classes *impervious surfaces*, *building*, *low vegetation*, *tree*, *car*, and *clutter/background*. As cars are not considered to be contained in topographic databases, we merged this class with *impervious surfaces*. For each pixel, we defined a five-dimensional feature vector  $\mathbf{x}_n(\mathbf{x})$  consisting of values for the normalised difference vegetation index (NDVI), the normalised digital surface model (nDSM), indicating the heights above ground, the red band of the TOP, smoothed by a Gaussian filter with  $\sigma=2$ , and hue as well as saturation obtained from the TOP, both smoothed by a Gaussian filter with  $\sigma=10$ . These features were selected from a larger pool of generic features based on the feature importance analysis of a random forest classifier (Breiman, 2001).

The other two data sets, subsets of the data used in (Hoberg et al., 2015), are based on satellite imagery. Data set 2 consists of a subset of a Landsat image of an area near Herne, Germany, covering 8.6 x 5.9 km<sup>2</sup> with a GSD of 30 m (about 350 x 300 pixels), acquired in 2010. Only the red, green and near infrared bands are available to us. The reference contains three classes, namely *residential area*, *forest* and *cropland*. Data set 3 consists of a RapidEye image with a GSD of 5 m of an area near Husum, Germany, also acquired in 2010. The area covered by this image is about 3.500 x 1.900 pixels or 16.8 x 9.6 km<sup>2</sup>, and, again, only the red, green and near infrared bands are available to us. The reference contains the classes *residential area*, *rural street*, *forest* and *cropland*. For both data sets we selected seven features, namely the original grey values in the three available bands, the results of a colour space transform applied to the three-band false colour infrared images (intensity, hue, saturation), and the NDVI.

We carried out two series of experiments. The first series, only based on one patch (patch 3) of the Vaihingen data, focussed on the evaluation of the method for label noise tolerant training described in Section 3.2. In these experiments, we did not use the CRF, but only the local classifier. The reference was used to obtain training labels, but these labels were contaminated randomly with varying degrees of label noise with different properties before training. In addition to noise tolerant logistic regression (LN), we also trained a standard multiclass logistic regression (MLR) and a Random Forest (RF) (Breiman, 2001) classifier using the training data thus derived, and we applied these classifiers to the image data. The results of all classifiers are compared to the reference to obtain confusion matrices and derived metrics such as completeness, correctness, quality and overall accuracy (OA), e.g. (Rutzinger et al., 2009). These experiments are designed to investigate the potential and limitations of the noise tolerant training procedure. RF was chosen for comparison as a representative example for a discriminative classifier that is supposed to be robust to some degree of label noise (Frénay & Verleysen, 2014). These experiments are presented in Section 4.2.

The second set of experiments is designed to evaluate the CRF-based method for change detection, and it uses all three data sets. For that purpose, we manually changed the existing database information used as reference, simulating realistic scenarios of change, e.g. urban redevelopment projects. After that, the classifier was trained using the simulated data, and it was applied to the feature vectors derived from the sensor data. We compare the results to the reference, deriving the same quality metrics as mentioned previously. This set of experiments highlights the feasibility of the overall approach for

different scenarios, also in terms of image resolution. These experiments are presented in Section 4.3.

For the Vaihingen data set we used quadratic expansion for the feature space mapping  $\Phi(\mathbf{x}_n)$  (cf. Section 3.2), whereas for Herne and Husum we used the original features. The standard deviation  $\sigma$  of the regularisation term in eq. 4 was set to  $\sigma=10$  in all experiments. If not stated otherwise, the initial values for the transition matrix  $\Gamma$  were  $\gamma_{ij} = 0.8$  for  $i = j$  and  $\gamma_{ij} = 0.2/(K-1)$  for  $i \neq j$ , where  $K$  is the number of classes, corresponding to a situation in which 80% of the pixels are expected to remain unchanged. The initial values for the weights  $\mathbf{w}$  in label noise robust training were determined by standard logistic regression training without assuming label noise. In cases involving the CRF, the parameters of the interaction potential were set to  $\beta_0 = 5.5$  and  $\beta_1 = 4.5$ , respectively.

#### 4.2 Evaluation of Label Noise Tolerant Training

In the first part of this set of experiments, we contaminated the training data by label noise according to the NCAR model. That is, the label noise was assumed to be independent of the class labels, so that the true transition matrix contained identical values  $\gamma_{ij} = 1-\eta$  for  $i = j$  and  $\gamma_{ij} = \eta / (K-1)$  for  $i \neq j$ , where  $\eta$  characterises the percentage of erroneous training labels. Training labels were changed randomly according to that model to simulate label noise. We varied  $\eta$  from 0% to 50% in steps of 10%. For training we used 30% of the pixels, which we chose randomly from all available data, taking care to have approximately equal numbers of training samples per class. The RF classifier used for comparison consisted of 300 trees of a maximum depth of 25. A node was split if it contained more than 5 training samples. Each experiment was repeated 20 times, using different training pixels and changing different class labels to simulate label noise. We report the average overall accuracy obtained for all pixels in the scene and also give error bars to indicate maximum and minimum numbers over all 20 test runs per experiment.

The results of the first part of this set of experiments are shown in figure 1. In the absence of label noise, RF delivers slightly better results than both versions of logistic regression, all classifiers achieving an OA of about 84%. With  $\eta = 10\%$  of wrong labels, RF still delivers results on par with the label noise tolerant logistic regression, but then the OA is decreasing down to 67% for 50% label noise. Standard logistic regression (MLR) turns out to be more robust than RF, with a decrease of only 2-3% even for large amounts of label noise. Label noise robust logistic regression is hardly affected at all, performing at the same level more or less independently of  $\eta$ . This good performance may be partly caused by the fact that this type of label noise corresponds well with the initialisation of the transition matrix (cf. Section 4.1). Note that the error bars in figure 1 are very small.

However, the NCAR model is not very realistic for the task at hand. Some changes in topography are more likely to occur than others, so that the elements of the transition matrix may vary to a larger degree. This is why in the second part of this set of experiments, we simulated label noise according to the NAR model, where the likelihood of a change depends on the class labels. Again we used the variable  $\eta$  to characterise the amount of label noise in the training data, but the true transition matrix was generated in a different way. For each row  $i$  we randomly selected the probability for each class transition in that row ( $\gamma_{ij}$  for  $i \neq j$ ) so that the sum  $S = \sum \gamma_{ij} \leq \eta$ . The element of the main

diagonal was then set to  $\gamma_{ii} = 1 - S$ . Thus,  $\eta$  can be interpreted as the maximum percentage of change, and the classes are also affected by change to different degrees. Label noise was simulated according to the transition matrix just described. We varied the values of  $\eta$  as in the case of the NCAR experiment, using 30% of the data for training and carrying out 20 tests for each value of  $\eta$ . We also varied the transition matrix in each of these tests. To assess the influence of the initial values for the elements of  $\Gamma$ , we carried out a second set of tests for  $\eta = 50\%$  with label noise tolerant logistic regression based on the initial values  $\gamma_{ij} = 0.5$  for  $i = j$  and  $\gamma_{ij} = 0.5/(K-1)$  for  $i \neq j$ ; this version is referred to as version LN50. The overall accuracy values achieved in these experiments are shown in figure 2.

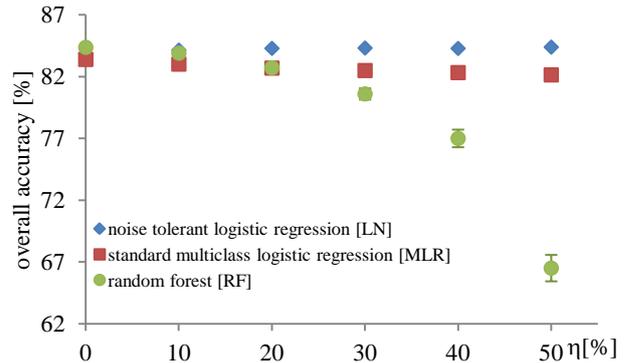


Figure 1. Overall accuracy as a function of the amount of label noise  $\eta$  (NCAR model) for three different classifiers.

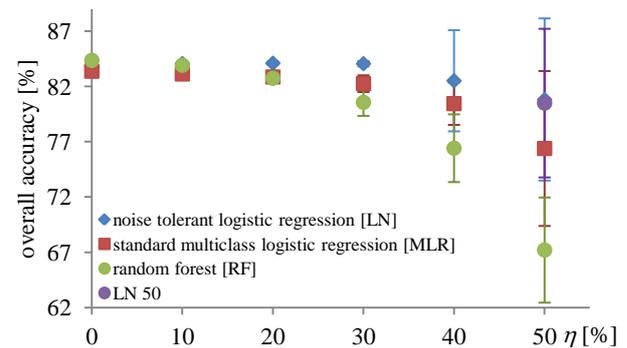


Figure 2. Overall accuracy as a function of the maximum amount of label noise  $\eta$  (NAR model). Note that LN50 was only tested for  $\eta = 50\%$ .

Analysing figure 2, the first observation is that the error bars become longer with increasing amount of label noise. This is partly due to the fact that the actual amount of label noise may vary ( $\eta$  only is the maximum amount of label noise), but it is also caused by the impact of different true transition probabilities: the changing label noise with respect to the class labels does affect classification accuracy. Apart from that, we observe a similar behaviour as in the previous test as far as the average OA of RF is concerned. For the standard logistic regression (MLR), the decrease in OA is larger than in the NCAR case. This is to be expected: whereas the decision boundary between two classes is not likely to change if the label noise is equally distributed on either side of it, unbalanced label noise will shift it towards the class having more wrong labels. For the label noise tolerant version (LN), hardly any decrease in OA can be observed for  $\eta \leq 30\%$ . There is a tendency towards a smaller OA for larger amounts of label noise, indicating that unbalanced label noise also affects that method, but nevertheless LN consistently outperforms the other classifiers, and even for  $\eta = 50\%$  the loss in average accuracy is only about

4%. The experiment with the different initialisation for  $\eta = 50\%$  (LN50) achieves nearly identical results although one could have expected it to perform better because the initial values are closer to the true ones. This indicates the robustness of the method with respect to the initialisation of  $\Gamma$ .

We cannot give a detailed analysis of completeness and correctness here for lack of space. These quality metrics follow similar trends as the average OA. Buildings and impervious surfaces obtain better quality measures than the other classes, but this observation is independent from the classifier used. In general, the experiments presented in this section indicate that label noise robust logistic regression is well-suited to cope with even relatively large amounts of label noise.

We also analysed the differences between the estimated transition matrices and the true ones that were used to simulate label noise. Figure 3 shows the median of the absolute differences between estimated and true matrix entries for the simulations based on the NAR model. Again we observe that the errors increase with the amount of label noise. For  $\eta = 50\%$ , the median absolute difference is about 5%, identical to about 10% of the amount of label noise, which we consider to be relatively accurate. The maximum differences, not shown in the figure for lack of space, show a higher rate of increase with increasing amount of label noise. For  $\eta = 50\%$ , the average maximum error over 20 tests is about 20%.

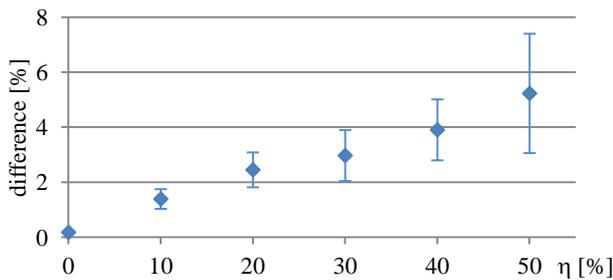


Figure 3. Median of absolute differences between the estimated and the true elements of the transition matrix (LN).

### 4.3 Evaluation of Change Detection

In Section 4.2, the label noise was distributed uniformly in the image, which is not very realistic when considering real changes in topography. For the experiments reported in this section, we simulated realistic changes in all our data sets, changing about 15%-25% of the scene. For three of the Vaihingen test patches we simulated two outdated databases with different distributions of change (cf. figure 4; for Vaihingen we refer to patches by their numbers as given in the benchmark documentation (Wegner et al., 2015), using underscores to differentiate variants of the outdated database. Thus, 30\_1 refers to the first variant of the database for patch 30). In this set of experiments, we used all pixels of the outdated database for training and applied the CRF-based classifier to the data. We compare the versions LN and MLR for the association potential of the CRF only to assess the benefits of the version with label noise tolerant training over standard training of logistic regression. The resulting values of overall accuracy for all test areas are presented in figure 5.

Figure 5 shows that the overall accuracy achieved in classification if the label noise robust version of logistic regression is used for the association potential of the CRF (LN) is better than the one for MLR for nearly all the cases. The exceptions are the second variants of the (simulated) outdated databases for areas 30, 32 and 37 in Vaihingen, characterised by

different distributions of label noise in the scene (cf. figure 4 for the difference in area 30). In general, the improvement of LN over MLR is in the order of about 1-2% in Vaihingen. This corresponds to the scenario with approximately 20% label noise observed in Section 4.2, where the differences between these two versions were not yet very pronounced. The improvement of LN over MLR is slightly more obvious for the data sets based on satellite imagery. Note that the disadvantages of MLR may also be mitigated by the smoothing effects due to the CRF. An example for the results achieved by LN is shown in figure 6.

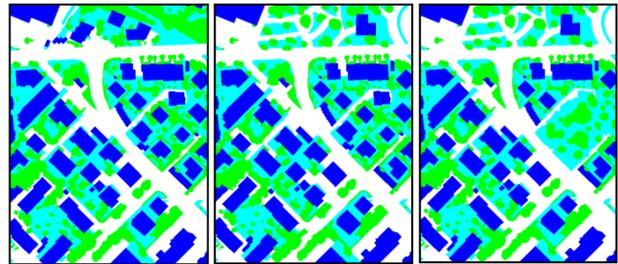


Figure 4. Area 30 in Vaihingen. Left: reference; centre: simulated topographic database (variant 30\_1), right: simulated topographic database (variant 30\_2). Colours: white: *impervious surface*; blue: *building*; green: *tree*; cyan: *low vegetation*.

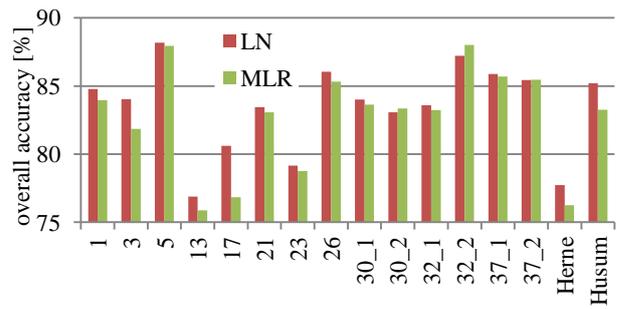


Figure 5. Overall accuracy [%] achieved for all test sites for two versions LN and MLR. Except for Herne and Husum, the numbers are the patch numbers of the Vaihingen data (Wegner et al., 2015).

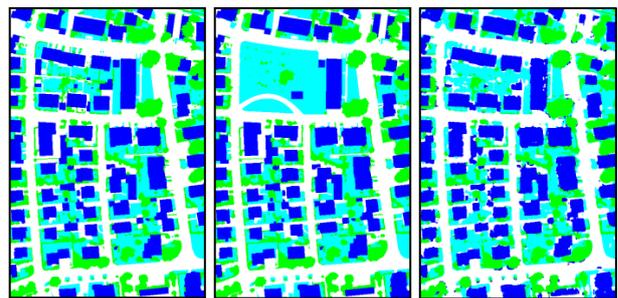


Figure 6. Vaihingen, area 3. Left: reference; centre: simulated database; right: classification results (LN). The colour code is identical to figure 4.

A typical reason for problems of label noise robust training, occurring in the three cases where MLR achieves a slightly better result than LN, is a change that produces an object that is not represented by the correct training data for that class. An example is the construction of new buildings having an atypical roof material that is not used for any other building in the scene. An analysis of the estimated transition matrices indicates that clusters of atypical pixels that all correspond to label noise can

cause an overestimation of the off-diagonal elements of these matrices that obviously lead to errors in classification. Again, we cannot give a detailed analysis of completeness and correctness per class for lack of space; an example, comparing the two versions LN and MLR for all classes except *clutter/background* (which only occurs in one patch and, thus, is not representative) is shown in figure 7. The figure indicates that there is a different trade-off between type 1 and type 2 errors. However, the quality, being a compound measure integrating both completeness and correctness, is consistently higher for LN than for MLR in this case.

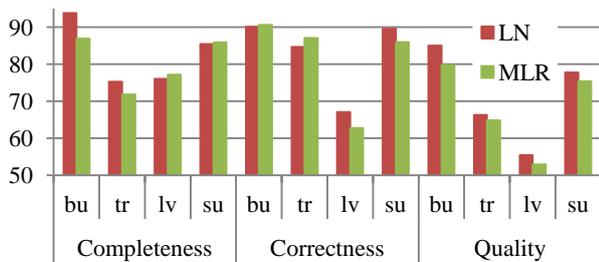


Figure 7. Completeness, correctness and quality per class (bu: *building*, tr: *tree*, lv: *low vegetation*; su: *impervious surfaces*) for the results in area 3 (Vaihingen). The results for version LN correspond to figure 6 (right).

So far, the evaluation has concentrated on the entire image, thus also integrating unchanged pixels that were also used in the training process. Figure 8 shows the overall accuracy achieved in all tests, only taking into account the pixels affected by a simulated change. In the majority of the examples, LN delivers better results than MLR. The improvement can reach 10% (Vaihingen, area 17), but a more realistic number would be 1-3%, which also applies to the satellite images. In the second variant of changes for area 32 (32\_2), where MLR is better than LN by about 5%, there was a large new part of an industrial building with atypical roof material that could not be detected correctly, a problem already discussed above.

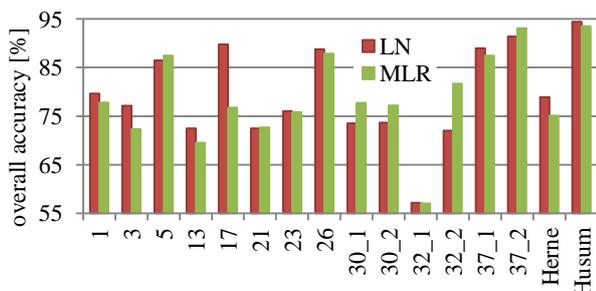


Figure 8. Overall accuracy for all tests only taking into account the pixels affected by a change.

Again, we also evaluated the differences between the estimated and the true elements of the transition matrices; figure 9 shows the median of these differences for all tests. In general, the median difference is below 3%, corresponding well to the scenario in figure 3 given the amount of simulated changes. Only the differences for the data set from Husum are atypically large. The median of the differences is about 5.5%. This problem can be attributed to the class *rural streets* contained in that data set. The maximum error in the transition matrix is related to a transition from *rural street* to *residential*, the corresponding probability being estimated as 86% although no such changes were simulated in the data. The estimation of the transition matrix may have been negatively affected by the fact

that only a small number of pixels belongs to class *rural streets*, which, thus, is underrepresented in the data set.

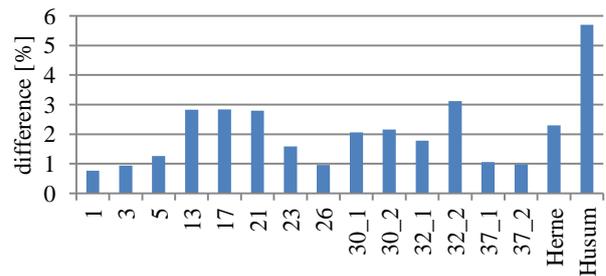


Figure 9. Median of absolute differences between the estimated and the true elements of the transition matrix (LN).

## 5. CONCLUSION

In this paper we have presented a method for change detection based on CRF that uses an outdated topographic database to derive training labels for the supervised classification of new data, without any need for training data generated manually. The method takes into account the unavoidable errors in the database by using a model of logistic regression which can deal with label noise. Comparing the classification results with the database, changes having taken place between the original acquisition of the database and the time epoch when the sensor data were acquired.

In our experiments we tested label noise tolerant logistic regression under varying degrees of both, class-independent and class-dependent label noise. In both scenarios label noise tolerant logistic regression delivered very promising results. Even in the presence of up to 50% wrong training labels the classification accuracy was only affected to a small degree compared to a classifier trained on 100% correct labels, whereas the quality of a random forest classifier, supposed to be robust to some degree of label noise, deteriorated by a much larger margin. Applying the CRF-based classification to scenes with simulated realistic changes in the database, the use of label noise tolerant logistic regression for the association potentials increased the classification accuracy over a standard logistic regression for the changed areas by 1-3% in most cases. However, these experiments also showed the limitations of the training method. For instance, we consider the NAR model, which forms the basis of that method, to be too simplistic, neglecting the fact that in our application erroneous training labels occur in local clusters and, thus, are spatially correlated. As far as the estimated transition matrices are concerned, they are reasonably accurate in most cases. However, underrepresented classes or large amounts of label noise may affect the estimation in a negative way.

In our future work we want to expand the model underlying the label noise tolerant classifier so that it can take into account the fact that wrong labels may appear in local clusters. Furthermore, here we used values determined on an empirical basis for the interaction model of the CRF; a joint training procedure similar to the one described in (Kumar & Hebert, 2006) might lead to improved results. We also want to expand our experiments, not restricting ourselves to the relatively small training patches as we did in this paper, so that the problem of atypical objects can be overcome and the results become more representative for different types of imagery. Additionally, we want to expand our experiments to data with real changes instead of simulated ones in order to test our method in a scenario that is more realistic

with respect to the type and extent of change as well as to the level of detail and number of classes in the existing database.

Finally, we observe that multitemporal classification requires temporal transition probabilities, which are frequently hand-crafted based on heuristic models, e.g. (Hoberg et al., 2015). Our experience with estimating transition matrices presented in this paper gives us reason to believe that our method could be the basis for an empirical estimation of these important model parameters in such a multitemporal context.

### ACKNOWLEDGEMENT

The Vaihingen data set was provided by the German Society for Photogrammetry, Remote Sensing and Geoinformation (DGPF) (Cramer, 2010):  
<http://www.ifp.uni-stuttgart.de/dgpf/DKEP-Allg.html>.

### REFERENCES

- An, W., Liang, M., 2013. Fuzzy support vector machine based on within-class scatter for classification problems with outliers or noises. *Neurocomputing* 110(2013):101-110.
- Bishop, C. M., 2006. Pattern recognition and machine learning. 1<sup>st</sup> edn, Springer, New York (NY), USA.
- Bootkrajang, J., Kabán, A., 2012. Label-noise robust logistic regression and its applications. In: *Proceedings of the 2012 European Conference on Machine Learning and Knowledge Discovery in Databases*, Vol. I, pp. 143-158
- Breiman, L., 2001. Random forests. *Machine Learning*, 45(1):5–32.
- Bruzzone, L., Persello, C., 2009. A novel context-sensitive semisupervised SVM classifier robust to mislabeled training samples. *IEEE Transactions on Geoscience and Remote Sensing* 47(7):2142-2154.
- Büschfeld, T., 2013. Klassifikation von Satellitenbildern unter Ausnutzung von Klassifikationsunsicherheiten. PhD thesis, Fortschritt-Berichte VDI, Reihe 10 Informatik / Kommunikation, Vol. 828, Institute of Information Processing, Leibniz Universität Hannover, Germany.
- Champion, N., 2007. 2D Building change detection from high resolution aerial images and correlation digital surface models. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* Vol. XXXVI-3 / W49A, pp. 197-202.
- Cramer, M., 2010. The DGPF test on digital aerial camera evaluation – overview and test design. *Photogrammetrie Fernerkundung Geoinformation* 2(2010):73–82.
- Fréney, B., Verleysen, M., 2014. Classification in the presence of label noise: a survey. *IEEE Transactions on Neural Networks on Learning Systems* 25(5):845-869.
- Frey, B. J., MacKay, D. J. C., 1998. A revolution: belief propagation in graphs with cycles. In: *Proceedings of the Neural Information Processing Systems Conference*, pp. 479–485.
- Hoberg, T., Rottensteiner, F., Feitosa, R. Q., Heipke, C., 2015. Conditional random fields for multitemporal and multiscale classification of optical satellite imagery. *IEEE Transactions on Geosciences and Remote Sensing* 53(2):659-673.
- Jia, K., Liang, S., Wei, X., Zhang, L., Yao, Y., Gao, S., 2014. Automatic land-cover update approach integrating iterative training sample selection and a markov random field model. *Remote Sensing Letters* 5(2):148-156.
- Jianya, G., Haigang, S., Guorui, M., Qiming, Z., 2008. A review of multi temporal remote sensing data change detection algorithms. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XXXVII-B7*, pp. 757-762.
- Kumar, S., Hebert, M., 2006. Discriminative random fields. *International Journal of Computer Vision* 68(2): 179-201.
- Li, Y., Wessels, L. F. A., de Ridder, D., Reinders, M. J. T., 2007. Classification in the presence of class noise using a probabilistic kernel Fisher method. *Pattern Recognition* 40(2007):3349-3357.
- Lu, D., Mausel, P., Brondizio, E., Moran, E., 2004. Change detection techniques. *International Journal of Remote Sensing* 25(12):2365-2401.
- Mnih, V., Hinton, G., 2012. Learning to label aerial images from noisy data. In: *Proc. 29th International Conference on Machine Learning*, pp. 567-574.
- Radoux, J., Defourny, P., 2010. Automated image-to-map discrepancy detection using iterative trimming. *Photogrammetric Engineering & Remote Sensing* 76(2):173-181.
- Radoux, J., Lamarche, C., Van Bogaert, E., Bontemps, S., Brockmann, C., Defourny, P., 2014. Automated training sample extraction for global land cover mapping. *Remote sensing* 6(2014):3965-3987.
- Rutzinger, M., Rottensteiner, F. and Pfeifer, N., 2009. A comparison of evaluation techniques for building extraction from airborne laser scanning. *IEEE Journal of Selected Topics in Applied Earth Observation and Remote Sensing* 2(1):11-20.
- Shotton, J., Winn, J., Rother, C., and Criminisi, A., 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.
- Subudhi, B. N., Bovolo, F., Ghosh, A., Bruzzone, L., 2014. spatio-contextual fuzzy clustering with markov random field model for change detection in remotely sensed images. *Optics and Laser Technology* 57(2014):284-292.
- Vosselman, G., Gorte, B., Sithole, G., 2004. Change detection for updating medium scale maps using laser altimetry. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* vol XXXV-B3, pp. 207-212.
- Wegner, J.D., Rottensteiner, F., Gerke, M., Sohn, G., 2015. The ISPRS 2D Labelling Challenge. <http://www2.isprs.org/commissions/comm3/wg4/semantic-labeling.html>. Accessed 11/12/ 2015.