IDENTIFICATION OF MISCLASSIFIED PIXELS IN SEMANTIC SEGMENTATION WITH UNCERTAINTY EVALUATION

Lina E. Budde [1] *, Dimitri Bulatov [2], Dorota Iwaszczuk [1]

[1]Technical University of Darmstadt, Dept. of Civil and Environmental Engineering Sciences, Remote Sensing and Image Analysis, Darmstadt, Germany - (lina.budde, dorota.iwaszczuk)@tu-darmstadt.de
[2] Fraunhofer IOSB (Institute of Optronics, System Technologies and Image Exploitation), Ettlingen, Germany - dimitri.bulatov@iosb.fraunhofer.de

Commission II, WG II/6

KEY WORDS: Deep Learning, Markov Random Field Optimization, Uncertainty, Monte-Carlo Dropout, Aerial Images, ISPRS 2D Semantic Labeling.

ABSTRACT:

Classification, and in particular semantic segmentation, plays a major role in remote sensing. In remote sensing, the classes usually correspond to landcover or landuse types while the data elements are image pixels. The results are so-called semantically segmented pixels describing the content of the data for each pixel. The identification of misclassified pixels is essential to perceive the overall performance of the classification algorithm. In the case of semantic segmentation, it is typically done with ground truth labels. However, such ground truth labels are rare and mostly reserved for training only. Especially deep learning approaches are data-hungry algorithms requesting a lot of labeled examples. In this work, we explore the possibility of using Monte-Carlo dropout for the identification of model-induced misclassifications. In particular, we obtain uncertainty measures from several inferences induced by the Monte-Carlo dropout. Furthermore, we examine how Markov Random Field optimization can reduce the number of misclassifications and facilitate their identification. The extent to which uncertainties provide information about misclassifications is assessed. Our results allow detecting 51 % of the misclassifications using uncertainties. Application of Markov Random Field optimization leads to a reduction of the percentage of misclassifications while detecting 0.4 % more misclassifications as without.

1. INTRODUCTION

1.1 Motivation

Deep Learning approaches have become extremely popular in recent years for various tasks in the field of image analysis (Goodfellow et al., 2016). Such approaches are also state of the art in semantic segmentation (Long et al., 2015; Ronneberger et al., 2015; Badrinarayanan et al., 2015). In this context, the development of so-called convolutional neural networks for image analysis played a crucial role. They enable efficient processing of large amounts of image data. In remote sensing, especially through satellite imagery, a lot of large-scale and, in some cases, very high-resolution image data is available. Semantic segmentation of aerial imagery helps, for example, in agriculture and urban development by recognizing land cover (Kampffmeyer et al., 2016), which constitutes our main field of research. The quality of semantic segmentation depends largely on the amount and quality of available labeled data. Despite a large number of available images, there is usually a lack of reliable ground truth data necessary for supervised training. Especially in the case of convolutional neural networks, where the number of parameters to be estimated is extremely high, many authors (Tong et al., 2020; Marmanis et al., 2016) rely on pre-training the network with slightly different, but abundantly labeled data for the first couple of layers and fine-tune it at the end with the labeled data at hand. However, this strategy has its limitations because the vast majority of parameters to be determined is contained in the latter, e.g. fully connected layers. Thus, many authors try to use the already classified data in a very reliable way as new training data (Wang et al., 2017).

1.2 Previous work

Clearly, automatically generated land cover maps should be as error-free as possible. However, ground truth data have been used to detect errors so far. This ground truth data is withheld from the training and reduces the amount of usable training data. The above examples show that we need additional methods for good quality estimation of semantic segmentation and to discover the faulty results more easily. Determining a model uncertainty offers the chance to identify areas of possible misclassification without ground truth data. However, such uncertainty information is not automatically available in neural networks (Kendall and Gal, 2017). Previous works have already tackled the questions issued in this work, namely how to measure the confidence of the CNN output and, as a consequence, how to select reliable training data for deep-learning-based methods. Some works combine the CNN classifier with additional measures which are supposed to make the classification output more interpretable. For example, (Papernot and McDaniel, 2018) use the nearest neighbor classifier right within the convolutional network in order to estimate the nonconformity of a prediction in the training data. On the contrary, (Dong et al., 2019) apply random forests at the end of the deep learning-based pipeline arguing that the importance of these features is more easily tractable (e.g. using boot-strapping techniques) in comparison to the black-box-like CNNs. Generally, the a-posteriori estimates (i.e., tree votes in the random forest (RF) or probabilities in neural networks) can function as indicators of classification uncertainty (Shadman Roodposhti et

* Corresponding author

al., 2019). In this work, conclusions were drawn that while correctly classified pixels belong to the low uncertainty areas, most of the incorrectly predicted class labels are located inside high-uncertainty areas with very few exceptions within low-uncertainty regions. The authors of (Shadman Roodposhti et al., 2019) reported a better correlation between accuracy and entropy in deep learning techniques than for Random Forest. Another quite frequently used technique to study the effect of uncertainty in source data intrinsically is the Monte Carlo modeling (Gal and Ghahramani, 2016), which allows determining the uncertainty with additional effort only in inference. Various measures of uncertainty from standard deviation to entropy (Kampffmeyer et al., 2016; Kendall et al., 2015; Gal, 2016) were used already. The studies show that the largest uncertainties occur at the class boundaries. An increasing overall accuracy due to out-masked uncertain pixels leads to the assumption that the misclassifications are also located in these areas (Budde et al., 2020). To improve the quality of semantic segmentation, especially at boundaries, several attempts have been made to combine the advantages of neural networks and Markov Random Fields (MRFs) (Chen et al., 2018; Liu et al., 2018b; Liu et al., 2017; Zhang et al., 2018). The focus of their investigation was to extend the training by a Conditional or Markov Random Field. Hence the neural network handles feature extraction and Conditional Random Fields handle the use of context. However, this results in a significant slowdown in both training and inference (Teichmann and Cipolla, 2018). Analogously, (Paisitkriangkrai et al., 2015; Kampffmeyer et al., 2016) used Conditional Random Fields for smoothing of deep learning results.

Contribution: Encouraged by the positive findings of (Shadman Roodposhti et al., 2019) (see above) and other related works on the correlation of accuracy and uncertainty, we investigate the smoothing effect of discrete optimization in the form of MRF on the identification of misclassifications. Due to the consideration of neighboring pixels, the smoothing property should reduce misclassifications of individual pixels and support classification decisions using surrounding pixels. Lower uncertainty is expected in homogeneous areas so that the identification of possible misclassifications should be facilitated. The required uncertainties are to be determined by Monte-Carlo dropout without additional training effort. By comparing two different uncertainty measures, entropy and confidence, a suitable measure to identify misclassifications shall be found.

Notation: In the course of this article, we work with image pixels (x, y, etc.), which are, of course, two-dimensional vectors. The class labels are always denoted by s. For the landcover classification task, $s_x = 1$ means that x is a road pixel, $s = 2$ is for buildings, whereby we occasionally drop the subscripts, and so on. Hence, the probabilistic outputs of the presented classifiers will be three-dimensional ones. In the case of deep learning, one speaks about *probability cubus* $P = P(s_x)$, whereby *softmax* is usually applied to the network outputs for max-margin estimation. Contrarily, in the case of MRF-based method, it is convenient to perform cost or *energy minimization* and thus, refer to a *cost cubus* $C = C(s_x)$. As will be explained later, the negative logarithm is the common way to switch between the probability and cost cubus. The total number of classes is denoted by S.

Organization: The following section 2 explains the methods. The experimental setup is described in section 3. Section 4 contains the results. The discussion and the conclusion are in section 5 together with ideas for future research.

2. METHODS

2.1 Semantic segmentation with Monte-Carlo dropout

In this work, we evaluate the quality of a U-Net-based semantic segmentation. U-Net was first introduced by (Ronneberger et al., 2015). The original focused data sources were biomedical images. Meanwhile, this network became a very popular network architecture for semantic segmentation. Additional components, such as batch normalization and dropout, can be incorporated into the original U-Net architecture. Dropout was developed as a regularization method to avoid overfitting (Hinton et al., 2012). The idea is to simulate sub-networks by randomly turning off individual neurons in the network, helping to incorporate the principle of ensemble learning. In the default case, dropout is disabled for inference after training. In the case of using Monte-Carlo dropout (Gal and Ghahramani, 2016), dropout is still active during inference. Each time, the output from the softmax function provides predicted pseudo-probabilities for each pixel and class. The predicted probability $\bar{P}(s)$ is approximated by averaging of these pseudo-probabilities and thus from the softmax outputs (Gal, 2016):

$$\bar{P}(s) = \frac{1}{T} \sum_{t=1}^{T} P^t(s), \tag{1}$$

where T is the number of Monte-Carlo samples and $P^t(s)$ denotes pseudo-probability of t-th sample voting for the class s. The Monte-Carlo dropout has the advantage that no additional parameters have to be determined during the training.

2.2 Optimization on Markov Random Field

Discrete optimization techniques are often used for depth maps estimation or dense image matching tasks (Hirschmuller, 2007; Bulatov et al., 2011). In this work, we use the smoothness assumption that neighboring pixels in the landcover map mostly have the same classes. This assumption as soft constraint (Schindler, 2012; Bulatov et al., 2019) yields the cost function

$$C(s) = \sum_{x} \left[C_{\text{dt}}(s_x) + \sum_{x,y \in \mathcal{N}} C_{\text{sm}}(s_x, s_y) \right] \rightarrow \min, \tag{2}$$

whose efficient minimization (Szeliski et al., 2008) is required (see next paragraph). In (2), C_{dt} denotes the cost cubus while C_{sm} is the system of smoothness values of neighbors, which are defined by $\mathcal{N}$. In the simplest case, $\mathcal{N}$ is the 4-neighborhood of a pixel and C_{sm} is the Potts model multiplied with a constant scalar λ: $C_{\text{sm}}(s_x, s_y) = \lambda \min(|s_x - s_y|, 1)$.

At first glance, any method described in (Szeliski et al., 2008) can be applied to minimize the cost function (2). However, the move-making methods are less suitable because they do not provide the posteriori probability distribution needed to label the reliably reconstructed pixels. Therefore, and also because of the very simple smoothness prior, we extended the message passing semi-global algorithm of (Hirschmuller, 2007) to output not only the min-marginals but also the accumulated costs. The minimum cost along the class dimension induce the predicted class for each pixel. Note that the scaling of the a-posteriori cost cubus varies strongly from pixel to pixel, however, only the pixel-wise costs matter for successive computations.

2.3 Uncertainty evaluation

The use of Monte-Carlo dropout allows different evaluations to determine the uncertainty of the model (Gal, 2016). In this case, two different measures are determined. One is the Shannon entropy (3) from the averaged pseudo-probabilities of multiple Monte-Carlo samples. The second one is the confidence measure (4) from the cost cubus. The lowest confidence shows maximum uncertainty and thus, possibly incorrectly classified pixels:

$$H_1 = -\frac{1}{\log(S)} \sum_{s=1}^{S} \bar{P}(s) \, \log_2 \left(\bar{P}(s) \right), \tag{3}$$

where $\bar{P}(s)$ is from (1). The confidence is computed according to (Pollefeys et al., 2008). The reliable labels have a confidence near to one while values near zero indicate high uncertainty. On the contrary, for entropy, the highest uncertainty corresponds to the values close to 1. Therefore, for a better comparison, the complementary measure is computed as follows:

$$H_2 = 1 - \left(\sum_{s=1}^{S} \exp \left(- \frac{(C(s) - C(s^*))^2}{\sigma^2} \right) \right)^{-1}, \tag{4}$$

where s^* is the predicted by the U-Net or MRF output label (for example, $s_x^* = \arg\max_s(P(s_x))$ and σ is empirically determined noise factor given by

$$\sigma = Q_{0.75} \left[C(s) - C(s^*) \right], \tag{5}$$

where $Q_{0.75}$ is the 75 % quantile reflecting the fact that a-priori values 75 % of all pixels have been obtained correctly and are *not* supposed to be oversmoothed (Bulatov et al., 2011). Note that C is the a-priori cost cubus. For each uncertainty measure, an uncertainty map is produced. From this, those areas with the largest uncertainties can be extracted and compared with the misclassifications. The extraction is performed using a threshold value.

3. EXPERIMENTAL SETUP

3.1 Dataset

For the experiments we used the 2D semantic labeling Potsdam dataset[1] (ISPRS WGII/4, n.d.). Eight of the 38 available tiles, generally distributed over the entire area, were selected as test data. In detail, these are the tiles with the denotation 2_12, 4_15, 5_11, 5_14, 6_11, 6_12, 6_14 and 7_9. The remaining 30 tiles were used in a ratio of 80 to 20 for training and validation. This subdivision of the dataset allows to examine test data with a wider varying – in comparison to the original contest – image structures, like water bodies and large park areas. As input features, all available channels are used. These are: RGB and NIR multispectral true orthophotos and a normalized surface model. The ground sampling distance is 5 cm. The ground truth data contains six different label classes: impervious surface, building, low and high vegetation (less formally, tree and grass), as well as vehicle and clutter.

3.2 Pipeline

Figure 1 shows the pipeline for the test data tiles. In the first step, the trained U-Net is used. Each test data tile is predicted

[1] http://www2.isprs.org/commissions/comm3/wg4/2d-sem-label-potsdam.html

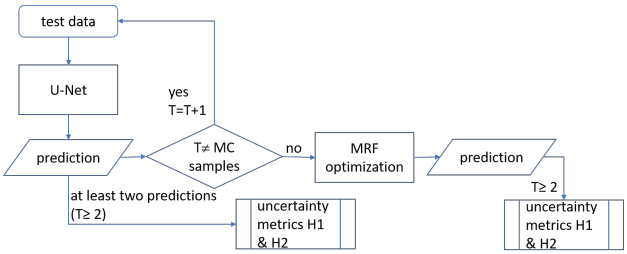

Figure 1. Pipeline for uncertainty evaluation from U-Net predictions and after Markov Random Field optimization

20 times ($T = 20$). Due to Monte-Carlo dropout, this results in 20 possibly different prediction values. In the second part, these predictions are used to calculate aggregated cost cubes by the MRF optimization. Uncertainties are determined both after the U-Net predictions and after optimization. We compare two different uncertainty metrics: entropy and confidence (section 2.3). To evaluate the performance, the F1-score is used.

3.2.1 U-Net The used U-Net architecture contains zero-padding and batch normalization. Dropout is used before the fully connected layer with 50 % drop probability. For the first convolution layer, 32 filters are used. The limited computing power required a resolution reduction by the factor 2 compared to the original data. Moreover, training took place on image patches of the size 300×300 pixels. To the image patches, data augmentation is added. The chosen data augmentation operations include flipping, multiple 90-degree rotations and normally distributed noise. A cross-entropy weighted with the class frequency is chosen to be our loss function. In the test phase, image patches of size 1500×1500 pixels were used.

3.2.2 MRF optimization The U-Net outputs pseudo-probabilities from a softmax layer for each Monte-Carlo sample. To process these with the MRF, a conversion to 16 bit integer cost is desirable in order to create data arrays tractable by the standard CPU even for relatively large images. The probabilistic output is usually processed by negative logarithm as follows

$$C = 2048 \min \left(-\frac{log_2(P)}{\min(log_2(P))}, 1 \right), \tag{6}$$

whereby square brackets denote rounding, and the constant $2^{11} = 2048$ reflects the possible cost accumulation from the characteristic paths of the semi-global optimization with the smoothness parameter λ varying between 400 and 800. The optimization from section 2.2 results in the a-posteriori distribution cube for each Monte-Carlo sample. From this, both the class prediction and the uncertainties can be derived.

3.2.3 Uncertainty evaluation For the uncertainty evaluation, we use both entropy and confidence (section 2.3). To evaluate entropy from aggregated costs, a re-conversion in probabilities is necessary (6). In each resulting uncertainty map, a specific percentage of the pixels with the highest uncertainty values are marked as uncertain. The removing percentages vary from zero to 50 %. This labeling creates a binary mask each time. Subsequently, the pixels marked as uncertain are checked for correctness in classification with the available ground truth data.

Setting	F1 value [%]						
	Impervious surface	Building	Low vegetation	Tree	Car	Clutter	OA
U-Net without Monte-Carlo dropout	87.7	90.3	79.7	72.1	65.7	23.8	82.6
U-Net with Monte-Carlo dropout	87.7	90.3	79.7	72.1	65.7	23.8	82.6
U-Net + MRF400	**87.8**	90.3	79.7	72.2	65.7	23.6	**82.7**
U-Net + MRF800	87.7	90.2	79.7	**72.3**	65.4	23.4	**82.7**

Table 1. F1 value [%] of semantic segmentation for the classes Impervious surface, building, low vegetation, tree, car and clutter. Additionally, the overall accuracy (OA) is specified. Improvements by MRF-based methods are bold.

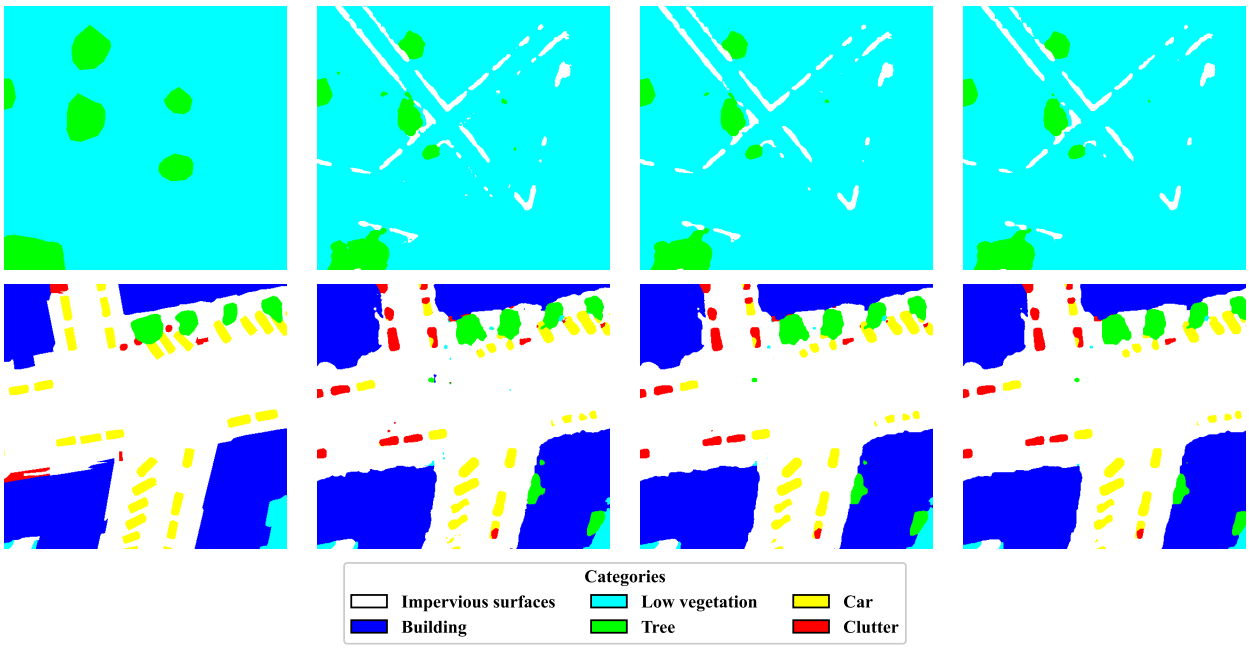

Figure 2. Image details of semantic segmentation. From left to right: ground truth, U-Net segmentation, MRF400 optimization, MRF800 optimization.

4. RESULTS

With the results presented below, the first step is to investigate the contribution of Monte Carlo sampling to semantic segmentation. In a second step, the correlation between uncertainty and misclassification is considered. Finally, we wish to explore which of the two uncertainty measures presented here has a higher correlation to the misclassifications. To quantify a possible improvement by MRF optimization, the results are presented both with and without MRF optimization.

4.1 Semantic Segmentation

The results of the semantic segmentation can be found in Table 1. The highest and the lowest values of accuracy are achieved by the building class and the clutter class, respectively. We see that the accuracy is comparable to the earlier results of the contest[2], such as those of (Volpi and Tuia, 2016), and stay slightly behind the newer methods, for example, (Liu et al., 2018a), who increasingly considers context information from different resolution levels via a self-cascaded network. However, it was not our purpose to optimize the classifier but to investigate the added value that the Monte-Carlo dropouts and Markov Random Fields can provide. Thus, it is possible to apply the presented methods to other networks as well. From the results in Table 1, compared to a single prediction with the U-Net without Monte Carlo dropout, there is no improvement in semantic segmentation from multiple predictions with Monte Carlo dropout. For the MRF optimization, two different smoothing parameters are tested. A more generous and a more rigorous smoothing are denoted by MRF400 and by MRF800, respectively. The quantitative results yielded only a small improvement for the tree class (Table 1). Instead, the accuracy for the clutter class decreases. This clutter class occurs, among other things, in the area of object edges. Due to smoothing, these pixels tend to be assigned to the object or impervious surface class. However, there are various effects shown in Figure 2. This example compares the ground truth with the predicted classes of the U-Net and the MRF optimization. The top of Figure 2 illustrates an image detail of a park area. This example shows the capability of the used U-Net. The U-Net is able to classify small paths (Figure 3) which are not included in the ground truth data (Figure 2 top left). This is one source of deviations between our prediction and the ground truth, with the former one corresponding to the reality. At the bottom of Figure 2, the image detail shows the effect of optimization. In the area of the road intersection, the U-Net generates several misclassifications. Fortunately, the optimization process was able to remove individual misclassifications. A larger accumulation of incorrectly classified pixels cannot be completely removed. However, increased smoothing can further reduce the amount of such incorrect pixel accumulations.

[2] https://www2.isprs.org/commissions/comm2/wg4/results/potsdam-2d-semantic-labeling/

Figure 3. RGB image fragment around the park paths. The image appears very dark because of the 16-bit representation of color values.

Setting	Percentage of misclassifications [%]
U-Net	17.37
MRF 400	17.34
MRF 800	17.34

Table 2. Percentage of misclassifications from test data semantic segmentation.

4.2 Uncertainty maps

Before and after the MRF optimization, uncertainty maps are calculated (section 3.2). With both uncertainty metrics H_1 from (3) and H_2 from (4), the calculations result in six uncertainty maps (Figure 5). The example in Figure 5 illustrates the positive effect of the MRF smoothing. With smoothing, the homogeneous areas within an object are less uncertain than without. Visually, this image detail shows no significant differences between MRF400 and MRF800. Also, in this image detail, the entropy appears to have a lower sensitivity in the homogeneous areas after smoothing than the confidence. However, the edges appear less clear in the entropy map. For example, road markings are omitted as conspicuous pixels. For the use of the uncertainty maps for the detection of misclassifications, these maps are used as masks (section 3.2.3). The following section 4.3 contains the results of the evaluation based on these uncertainty masks.

4.3 Identification of misclassifications

The semantic segmentation from section 4.1 results in a percentage of misclassifications in Table 2. The percentage of misclassifications in the whole test data is thus about 17 %. Thereby, the percentage decreases by about 0.03 % when using the optimization. A change of 0.01 % corresponds to 7200 pixels. In practice, images differ from each other, and the percentage of misclassifications and the corresponding uncertainty values vary strongly in the evaluation. Incorrect input data also leads to false positive and true negative results. To find out which percentage of pixels should be removed to increase the accuracy, precision-recall curves are created (Figure 4). Removing pixels due to their uncertainty increases correctness, but at the expense of completeness. A compromise should be found between the number of removed pixels and remaining misclassifications. The left part of Figure 4 shows the evaluation for all classes. The right part of Figure 4 shows an example of the results of a reduction to a binary classification of the class building. In both cases, a significantly better performance of H_2 compared to H_1 can be observed. Only at a threshold of about 20 %, a growing difference between the U-Net and the MRF optimized results can be detected. For classes that are already very well classified, such as buildings, the increase in accuracy converges at a significantly lower removing fraction than for all classes. Since the threshold 20 % is a good compromise between precision and recall, the results with this threshold are considered in more detail below.

Setting	F1 [%]	
	incorrect	correct
CNN Entropy	46.74	87.76
MRF400 Entropy	41.81	86.64
MRF800 Entropy	41.24	86.51
CNN Confidence	47.62	87.96
MRF400 Confidence	47.78	88.01
MRF800 Confidence	**47.93**	**88.05**

Table 3. F1 values for detection of misclassifications for 20% threshold. As many pixels as possible should be ideally classified correctly and certain at the same time (correct). All uncertain pixels should be ideally misclassifications (incorrect). Results corresponding to best configurations are marked in bold.

Setting	Uncertainty metric [%]	
	Entropy	Confidence
detection rate CNN	**50.28**	51.22
detection rate MRF400	45.01	51.44
detection rate MRF800	44.41	**51.61**

Table 4. Detection rate of misclassifications with H_1 and H_2 while 20 % of the pixels with highest uncertainty are removed. Results corresponding to best configurations are marked in bold.

The F1-values from Table 3 show the correlation between the uncertainty and the misclassifications with 20 % threshold. This means: 20 % of the pixels are classified as uncertain and thus are candidates for misclassification. In this case most of the correct classified pixels are also labeled as certain (88.1 % with H_2). The ability to detect misclassifications is at most about 47.9 % with the MRF optimization and the complementary confidence measure H_2. There are still many uncertain pixels that have been correctly classified. The relation between the identified misclassifications and the included misclassifications (Table 2) are shown in Table 4. At least 51.2 % of misclassifications are detected using confidence. Stronger smoothing increases this percentage to up to 51.6 %. Among the pixel with high uncertainty, there are more misclassifications. From both Table 3 and Table 4, it can be seen that less misclassifications are detected using entropy. Using entropy, the best case is achieved with 50.3 % correct identifications of misclassifications with the U-Net without MRF (Table 4). However, when looking at Figure 5, it is noticeable that the correctly classified but as uncertain labeled pixels vary strongly for both uncertainty measures. Some correctly classified areas that are considered certain by entropy are considered uncertain by confidence and vice versa.

5. CONCLUSIONS

Uncertainty assessment techniques can provide an uncertainty map as a spatial approximator of classification accuracy, which can be used to locate and segregate unreliable pixel-level class

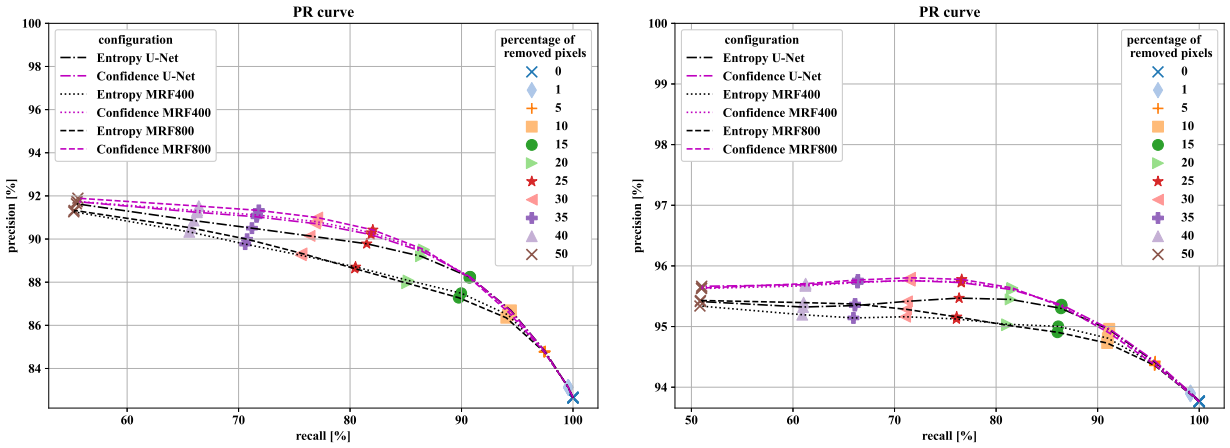

Figure 4. Precision-recall curve for each of the configurations with different percentage of removed pixels [%]. Here, a percentage of zero means that no pixels were removed due to uncertainty. Left: all classes are considered, right: only the binary classification building - no building is considered.

allocations from reliable ones. This is the first important conclusion we can draw from our results, because especially in the case of non-optimal configuration of the neural network, misclassifications can be detected by determining model uncertainties. Secondly, smoothing by the MRF optimization reduces the number of misclassifications. At the same time, the proportion of misclassifications in the marked uncertain areas is increasing. This is primarily due to lower uncertainty values of correct classifications. Over the whole test data, the identification by means of the confidence seems to be more successful than entropy. Additionally, this has the advantage because a reconversion into probabilities is not necessary. One direction for future work could include finding a reasonable combination of multiple uncertainty measures. The partly complementary signatures on the second and fourth columns of Figure 5 offer the possibility to reduce the number of pixels classified as correct by U-Net but uncertain by either single measure. This can also reduce the strong dependence on the choice of the threshold value. The threshold affects the trade-off of detecting as many misclassifications as possible and removing as few correct classifications as possible.

The Monte Carlo samples can be used for semantic segmentation as well as for determination of the uncertainty. However, an increased extra time due to the multiple prediction of the data during the inference phase has to be considered. A positive effect of MRF optimization occurs mainly for single misclassifications. In addition, a trade-off must be made between the improvement from MRF optimization and the additional processing effort. In particular, the MRF increases the processing time by one minute for each image patch.

Nevertheless, only errors caused by the model can be found with the Monte Carlo dropout method. Systematic gross errors in the input data caused, for example, by incorrect height values like in the Potsdam dataset, on contrary, remain undetected. Current semantic segmentation methods are also increasingly outperforming the quality of the sometimes erroneous or incomplete ground truth data. By comparing synthetically reconstructed data with real input (Xia et al., 2020), these ground truth differences could be deciphered in future studies.

References

Badrinarayanan, V., Handa, A., Cipolla, R., 2015. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Robust Semantic Pixel-Wise Labelling. *arXiv preprint arXiv:1505.07293*.

Budde, L. E., Schmohl, S., Soergel, U., 2020. Unsicherheitsauswertung von semantischer Segmentierung mittels Neuronaler Netze. *40. Wissenschaftlich-Technische Jahrestagung der DGPF* , 29, Publikationen der DGPF, 280–289.

Bulatov, D., Häufel, G., Lucks, L., Pohl, M., 2019. Land cover classification in combined elevation and optical images supported by OSM data, mixed-level features, and non-local optimization algorithms. *Photogrammetric Engineering & Remote Sensing*, 85(3), 179–195.

Bulatov, D., Wernerus, P., Heipke, C., 2011. Multi-view dense matching supported by triangular meshes. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(6), 907–918.

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A. L., 2018. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4), 834–848. dx.doi.org/10.1109/TPAMI.2017.2699184.

Dong, L., Du, H., Mao, F., Han, N., Li, X., Zhou, G., Zheng, J., Zhang, M., Xing, L., Liu, T. et al., 2019. Very High Resolution Remote Sensing Imagery Classification Using a Fusion of Random Forest and Deep Learning Technique—Subtropical Area for Example. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13, 113–128.

Gal, Y., 2016. Uncertainty in Deep Learning. PhD thesis, University of Cambridge.

Gal, Y., Ghahramani, Z., 2016. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. *ArXiv*, abs/1506.02142.

Goodfellow, I., Bengio, Y., Courville, A., 2016. *Deep Learning*. MIT Press. `http://www.deeplearningbook.org`.

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2012. Improving neural networks by preventing co-adaptation of feature detectors. *CoRR*, abs/1207.0580. http://arxiv.org/abs/1207.0580.

Hirschmuller, H., 2007. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on pattern analysis and machine intelligence*, 30(2), 328–341.

ISPRS WGII/4, n.d. 2D Semantic Labeling Contest. `http://www2.isprs.org/commissions/comm3/wg4/2d-sem-label-potsdam.html`.

Kampffmeyer, M., Salberg, A., Jenssen, R., 2016. Semantic segmentation of small objects and modeling of uncertainty in urban remote sensing images using deep convolutional neural networks. *2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 680–688.

Kendall, A., Badrinarayanan, V., Cipolla, R., 2015. Bayesian SegNet: Model Uncertainty in Deep Convolutional Encoder-Decoder Architectures for Scene Understanding. *CoRR*, abs/1511.02680. http://arxiv.org/abs/1511.02680.

Kendall, A., Gal, Y., 2017. What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? *CoRR*, abs/1703.04977. http://arxiv.org/abs/1703.04977.

Liu, Y., Fan, B., Wang, L., Bai, J., Xiang, S., Pan, C., 2018a. Semantic labeling in very high resolution images via a self-cascaded convolutional neural network. *ISPRS Journal of Photogrammetry and Remote Sensing*, 145, 78–95. dx.doi.org/10.1016/j.isprsjprs.2017.12.007.

Liu, Y., Piramanayagam, S., Monteiro, S. T., Saber, E., 2017. Dense Semantic Labeling of Very-High-Resolution Aerial Imagery and LiDAR with Fully-Convolutional Neural Networks and Higher-Order CRFs. *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, IEEE, 1561–1570.

Liu, Z., Li, X., Luo, P., Loy, C. C., Tang, X., 2018b. Deep Learning Markov Random Field for Semantic Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(8), 1814-1828.

Long, J., Shelhamer, E., Darrell, T., 2015. Fully convolutional networks for semantic segmentation. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3431–3440.

Marmanis, D., Wegner, J. D., Galliani, S., Schindler, K., Datcu, M., Stilla, U., 2016. Semantic segmentation of aerial images with an ensemble of CNSS. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 2016*, 3, 473–480.

Paisitkriangkrai, S., Sherrah, J., Janney, P., Van-Den Hengel, A., 2015. Effective semantic pixel labelling with convolutional networks and conditional random fields. *2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 36–43.

Papernot, N., McDaniel, P., 2018. Deep k-nearest neighbors: Towards confident, interpretable and robust deep learning. *arXiv preprint arXiv:1803.04765*.

Pollefeys, M., Nistér, D., Frahm, J.-M., Akbarzadeh, A., Mordohai, P., Clipp, B., Engels, C., Gallup, D., Kim, S.-J., Merrell, P., Salmi, C., Sinha, S., Talton, B., Wang, L., Yang, Q., Stewénius, H., Yang, R., Welch, G., Towles, H., 2008. Detailed Real-Time Urban 3D Reconstruction from Video. *International Journal of Computer Vision*, 78(2-3), 143–167. dx.doi.org/10.1007/s11263-007-0086-4.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. *CoRR*, abs/1505.04597. http://arxiv.org/abs/1505.04597.

Schindler, K., 2012. An overview and comparison of smooth labeling methods for land-cover classification. *IEEE transactions on geoscience and remote sensing*, 50(11), 4534–4545.

Shadman Roodposhti, M., Aryal, J., Lucieer, A., Bryan, B. A., 2019. Uncertainty assessment of hyperspectral image classification: Deep learning vs. random forest. *Entropy*, 21(1), 78.

Szeliski, R., Zabih, R., Scharstein, D., Veksler, O., Kolmogorov, V., Agarwala, A., Tappen, M., Rother, C., 2008. A comparative study of energy minimization methods for Markov Random Fields with smoothness-based priors. *IEEE transactions on pattern analysis and machine intelligence*, 30(6), 1068–1080.

Teichmann, M. T. T., Cipolla, R., 2018. Convolutional CRFs for Semantic Segmentation. *CoRR*, abs/1805.04777. http://arxiv.org/abs/1805.04777.

Tong, X.-Y., Xia, G.-S., Lu, Q., Shen, H., Li, S., You, S., Zhang, L., 2020. Land-cover classification with high-resolution remote sensing images using transferable deep models. *Remote Sensing of Environment*, 237, 111322.

Volpi, M., Tuia, D., 2016. Dense semantic labeling of sub-decimeter resolution images with convolutional neural networks. *CoRR*, abs/1608.00775. http://arxiv.org/abs/1608.00775.

Wang, K., Zhang, D., Li, Y., Zhang, R., Lin, L., 2017. Cost-Effective Active Learning for Deep Image Classification. *IEEE Transactions on Circuits and Systems for Video Technology*, 27(12), 2591-2600. dx.doi.org/10.1109/TCSVT.2016.2589879.

Xia, Y., Zhang, Y., Liu, F., Shen, W., Yuille, A. L., 2020. Synthesize then compare: Detecting failures and anomalies for semantic segmentation. A. Vedaldi, H. Bischof, T. Brox, J.-M. Frahm (eds), *Computer Vision – ECCV 2020*, Springer International Publishing, Cham, 145–161.

Zhang, C., Sargent, I., Pan, X., Gardiner, A., Hare, J., Atkinson, P. M., 2018. VPRS-Based Regional Decision Fusion of CNN and MRF Classifications for Very Fine Resolution Remotely Sensed Images. *IEEE Transactions on Geoscience and Remote Sensing*, 56(8), 4507–4521. dx.doi.org/10.1109/TGRS.2018.2822783.

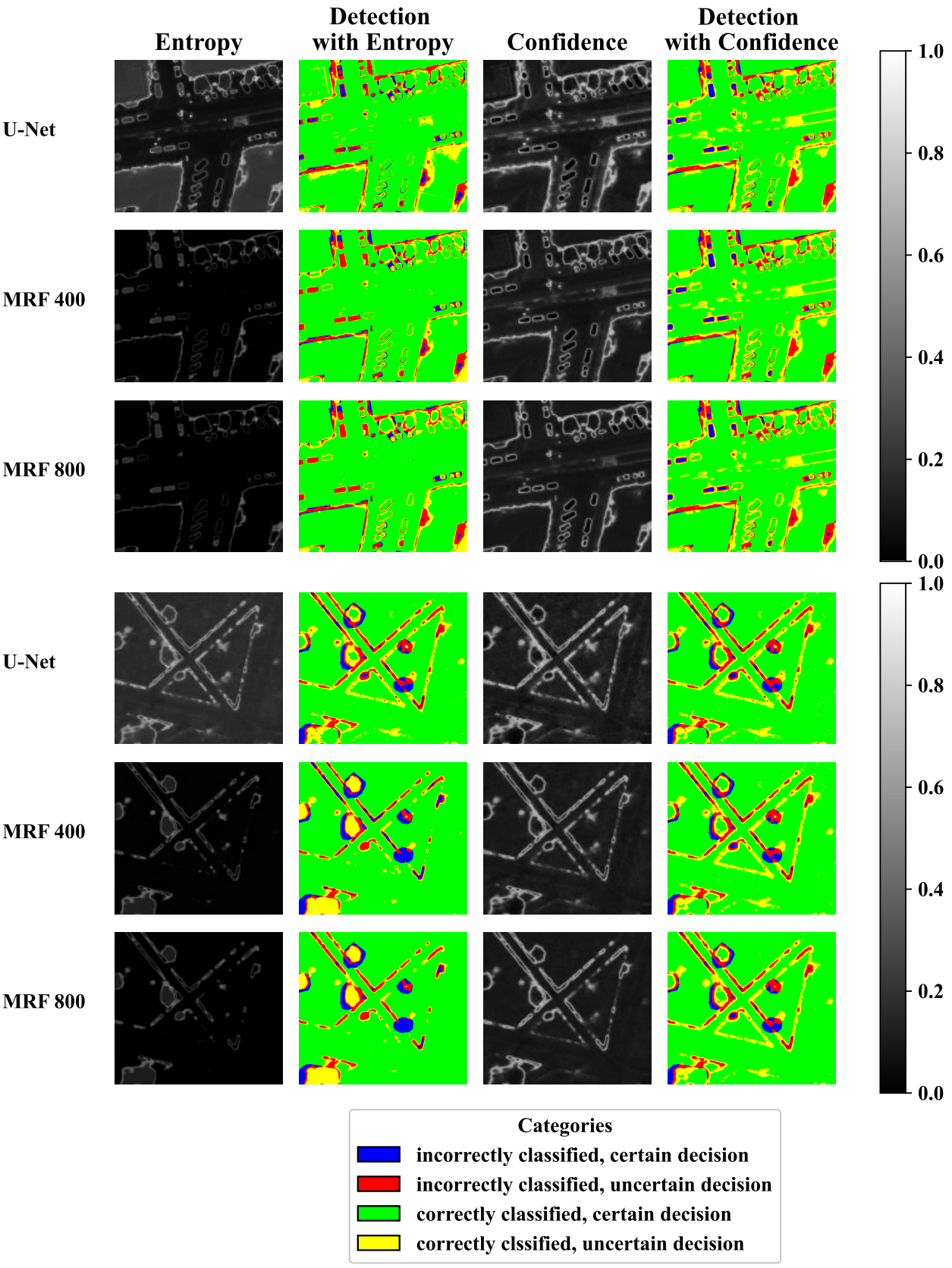

Figure 5. Uncertainty and detection maps for both image fragments of Figure 2. The entropy H_1 from (3) and H_2 from (4) for the choices U-Net, MRF400 and MRF800 are shown in first and third columns, respectively. For detection maps (second and fourth columns) uncertainty masks are compared with ground truth. A distinction is made between uncertain and certain pixels in combination with correct and incorrect classification.