Deep Learning for Remote Sensing

Nicolas Audebert[1,2], Alexandre Boulch[1], Adrien Lagrange[1,3], Bertrand Le Saux[1,*], and Sébastien Lefèvre[2]

[1]ONERA The French Aerospace Lab, DTIM, F-91761 Palaiseau, France
[2]Univ. Bretagne-Sud, UMR 6074, IRISA, F-56000 Vannes, France
[3]ENSTA ParisTech, F-91120 Palaiseau, France
[*]corresponding author: bertrand.le_saux@onera.fr

ABSTRACT

This work shows how various, recent statistical techniques can benefit to remote sensing. We focus on three tasks which are recurrent in Earth-observation data analysis: multimodal classification, orthophoto rectification and aerial image segmentation. For each of them we present a novel approach based on recent developments of deep learning and discrete optimization. We assess our approaches on challenging urban, multi-sensor data-sets and establish new state-of-the-art performances. It shows that deep learning allows re-thinking the remote sensing of areas with abundant information and offers promising paths for urban monitoring and modeling.

1 Introduction

Deep learning is a new way to solve old problems in remote sensing. Various changes in the technical ecosystem made it possible. First, data become abundant, thanks to more and more automated sensing and processing. Second, the theory behind machine learning was better understood and led to the development of algorithms (such as neural networks with several hidden layers) which obtained practical successes in related fields, such as speech recognition or computer vision. Third, computational capacities (e.g. highly-parallel processors) became widely available, allowing training these algorithms in tractable times.

It comes out that we can now use such powerful statistical models for various remote sensing tasks: detection, classification or data fusion. Back in 2010, Mnih and Hinton[21] started to train large deep networks for road detection by combining aerial imagery and open data (road network). More recently, convolutional networks were used for unsupervised feature extraction[25] or hyperspectral data classification[8]. In the meantime new data-sets with multi-sensor data and corresponding ground-truth of various land-use classes were released, such as the Zeebrugge IEEE-GRSS DFC data-set[12] or the Vaihingen ISPRS Semantic Labeling data-set[26]. They allowed to reach new state-of-the-art performances in image classification and showed the re-use of cross-domain databases is possible to gain and transfer knowledge[16,22,23]. New challenges will soon be addressed, such as image registration or 3D data analysis. Serendipity plays a role here: while meta-data for standard decision-making are not always available, the co-existence of various correlated, continuous data allows the training of regression models which give the same output as analytic processes, but faster and with more robustness.

In the following, we propose several unsupervised optimization and deep learning approaches to address challenging issues of remote sensing for urban monitoring and assessment: multimodal classification (Section 2), geometric correction of orthophotos (Section 3) and semantic segmentation (Section 4). Two European towns are chosen to evaluate the results: Vaihingen (ISPRS data set) and Zeebruge (IEEE-GRSS data-set). Data-sets contain several Infrared-Red-Green (ISPRS) or Red-Green-Blue (IEEE-GRSS) tiles, with the corresponding Digital Surface Model (DSM) and Lidar-captured point-cloud.

2 Multimodal semantic classification

Semantic labeling consists in automatically building maps of geo-localized semantic classes (e.g. land use: buildings, roads, vegetation; or objects: vehicles) upon Earth-Observation data[16]. We train convolutional neural networks (CNNs) designed for image classification, such as LeNet[17] or NiN[18], on multi-sensor data associated with urban classes. Input data are optical image patches at various scales (to give both high-resolution precision and contextual information) and height information from Lidar data.

2.1 Approach

Superpixel segmentation We first segment orthophotos using the SLIC (Simple Linear Iterative Clustering[1]) method.This allows to generate coherent regions at sub-object level. Patches used to feed the CNNs will then be extracted around the superpixel centroïd, and the class estimated by the algorithm will be assigned to the whole superpixel.

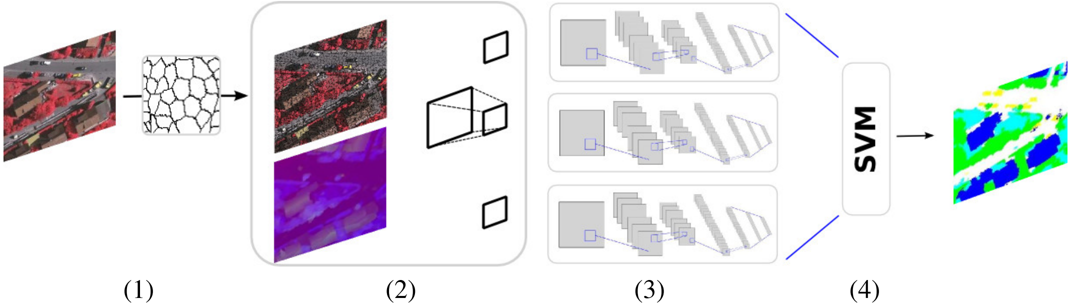


Figure 1. Semantic labeling work-flow: (1) superpixel segmentation; (2) multi-scale and multi-sensor patch extraction; (3) classification with CNNs; (4) fusion with multi-class SVM.

Multiple scale and multi-sensor data Our CNNs use 32×32-sized 3-channel patches as input. For each superpixel, we generate a first 32×32 patch at full resolution (which is roughly the size of a car) and a 124×124 patch (which is roughly the size of a house or a car in context) that we downsize to 32×32. We also build a composite image using the Digital Surface Model (DSM) from the original benchmark data, the normalized DSM (nDSM) provided with one of the baselines of the benchmark[10] and a Normalized Difference Vegetation Index (NDVI) computed using the Infrared (I) and Red (R) channels of the orthophoto according to the formula: $NDVI = (I - R)/(I + R)$. From this composite image we extract 32×32 patches. As a result, for each location defined by a superpixel, we get 3 patches at multiple scale / multiple data (ms/md).

Convolutional Neural Networks We used two different network architectures:

- The LeNet network[17] is made of three convolutional layers each followed by a rectified linear unit (ReLU) and pooling layers, then one more convolutional layer followed only by a ReLU, and finally two fully-connected layers and a softmax.
- The NiN network[18] implements a more complex structure which include three convolutional layers each followed by two fully-connected layers and then a final fully-connected layer and a softmax.

Although we could have used these networks as is, we chose to use them as feature extractors generated by the layer before the softmax one. For the NiN architecture, we had to add a second fully-connected layer at this stage to be able to generate usable vector outputs. The network parameters are learned using patches extracted from the training set, along with their respective class. We used mean subtraction, contrast augmentation and data whitening for preparing the network inputs. For each type of network, we train three CNNs in parallel: one for each scale and one for patches from the composite image.

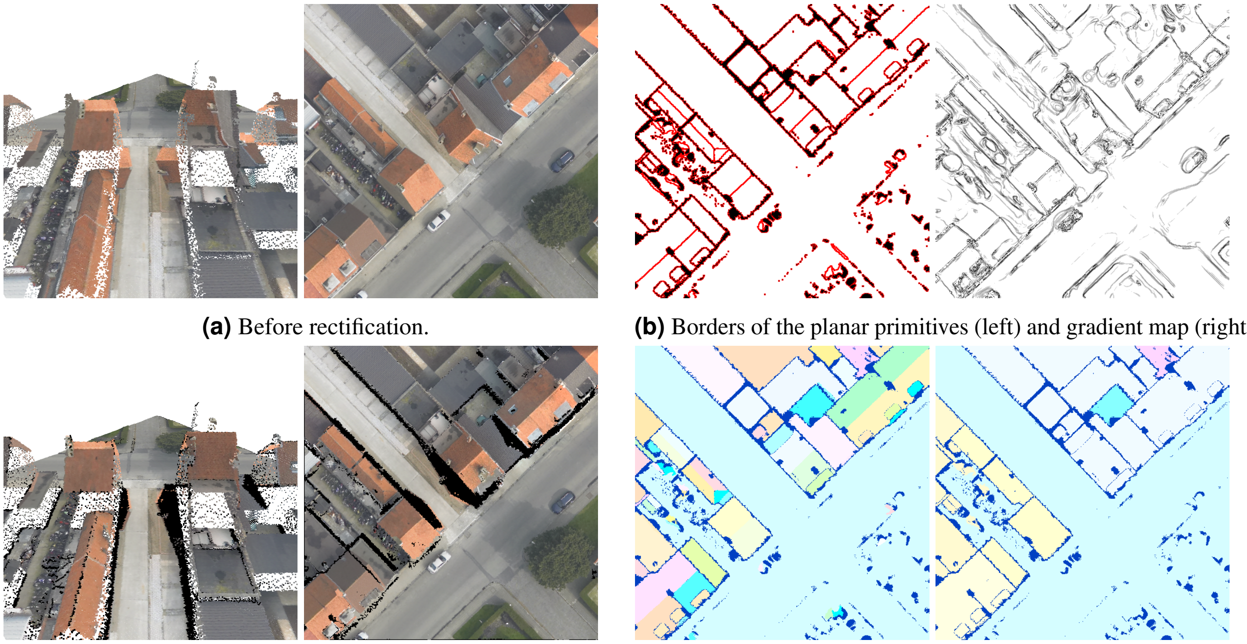
Support-Vector Machine The final classifier is a linear SVM trained after performing a grid search to tune the SVM parameters. More precisely, we train six SVMs corresponding to our six classes to proceed as a one-vs-all manner. Each SVM generates a soft-score map and from these six maps, we apply a simple max operation to select the predicted class. We form the inputs of the SVM classifier by concatenating the intermediate-layer features generated by the CNNs. Thus, the SVM performs both the data fusion of various networks (i.e. various data) and the classification.

2.2 Results

Table 1. F1 measures, overall accuracy and Cohen's Kappa coefficient of NiN-based or LeNet-based work-flows for semantic labeling.

Network	Imp. surf	Building	Low veg.	Tree	Car	Overall acc.	kappa
LeNet (val)	91.41	94.61	83.04	91.25	58.94	90.07	86.69
NiN (val)	90.84	93.12	83.51	91.35	71.80	89.82	86.43
LeNet (test ONE-2)	86.9	90.7	78.9	86.4	43.8	85.0	∼
NiN (test ONE-3)	86.7	89.3	79.0	86.4	56.3	85.0	∼

Table 1 shows performances of our work-flows for semantic labeling for both our validation set and the unknown test set provided by ISPRS (cf. http://www2.isprs.org/vaihingen-2d-semantic-labeling-contest.html).



(a) Before rectification.

(b) Borders of the planar primitives (left) and gradient map (right).

(c) After rectification.

(d) Projection vectors after optimization, with and without E_{regul}.

Figure 2. Geometric correction for orthophoto generation. IEEE-GRSS DFC tile example, orthopho and DSM point cloud.

Though we slightly over-fitted at training, the 85% test accuracy on unseen data shows the ability of CNNs for extracting meaningful semantic features and classification. In particular, non-linear NiN networks proved to be particularly efficient for small objects like vehicles: the fully-convolutional layers are less prone to overlook local details. Segmentation results can be seen on Fig. 4-d.

3 Geometric correction for orthophoto generation

Orthophotos are used in various domains, from cartography to simulation. They are subject to projection errors due to occlusions from elevated objects. We present a method for rectification of orthophotos given a digital surface model. The original image may be composed of a mosaic of orthoimages with unknown viewpoints. The goal is to associate to each point of the surface model, a projection direction for the point to be given the right color. This is an unsupervised learning task, under the regularization constraint that neighbor points should have the same projection direction. The problem is formulated as a discrete optimization problem over a graph and solved using state-of-the-art primal-dual techniques (cf. Fig. 2a, before correction and Fig. 2c, after correction).

3.1 Problem formulation

In order to associate to each point p of the DSM P its true color and then generate the true orthophoto, we compute a projection vector $\mathbf{u}_p \in U$ from P to the image I (the image is at the ground level of the DSM). We also impose the projection vector of neighboring points to be similar (consistency of the view point).

This problem can be very complex and costly: the size of P can be huge and the projections directions lie on the unit sphere, a continuous space. In order to reduce the potential complexity of the problem, we group neighboring DSM points into coherent planar cluster c and consider a unique projection direction for each cluster, the set of clusters is C. Secondly, we discretize the unit sphere by iterative refinement of an icosahedron mesh. As aerial images view direction are usually almost vertical, we also restrict the set of vectors to the directions with angle lower than 25° to the vertical. Let S bet this set.

Our problem is now a discrete optimization problem. The objective function is expressed in equation (1).

$$E(\mathbf{U}) = E_{\mathrm{data}} + E_{\mathrm{regul}} = \sum_{c \in C} \sum_{p \in c} \phi(p, \pi_{\mathbf{u}_c}(p)) + \lambda \sum_{c_1 \in C} \sum_{c_2 \in \mathcal{N}(c_1)} w_{c_1 c_2}\, \psi(\mathbf{u}_{c_1}, \mathbf{u}_{c_2}) \tag{1}$$

E_{data} takes high value when edges on the DSM do not correspond to edges in the image. The regularization term, E_{regul}, imposes neighboring points to have the same projection vector.

In $E(\mathbf{U})$, $\pi_{\mathbf{u}_c}(p)$ is the projection of the point p on the image according to $\mathbf{u}_c$ and

$$\phi(p,q) = \mathbb{1}(p \notin E_P)\,\|\vec{\nabla}_I(q)\|_2 \qquad (2) \qquad\qquad w_{c_1c_2} = \min(\bar{z}_{c_1}, \bar{z}_{c_2})\exp(-\frac{d_{c_1c_2}^2}{2\sigma^2}) \qquad (3)$$

where E_P is the set of edge points in the DSM and $\vec{\nabla}_I$ is the gradient in the image, $\bar{z}$ is the mean elevation of the point cluster, $d_{c_1c_2}$ is the minimum distance between c_1 and c_2 and σ is the attenuation parameter.

ϕ penalizes interior points of the clusters that project on the high gradients area in the image.

$\psi(\mathbf{u}_{c_1}, \mathbf{u}_{c_2})$ is the angle between the two vectors $\mathbf{u}_{c_1}$ and $\mathbf{u}_{c_2}$. The weight value $w_{c_1c_2}$ expresses the affinity of two segments, i.e., how the direction associated to c_1 influences that of c_2 (and conversely). We use the minimum of $\bar{z}$ values to decrease the influence of edges involving a primitive near to the ground. Projection error decreases with the closeness to the ground. Moreover as the ground may be a very large primitive, it may expand in the whole image, we do not want it to influence the other segments.

3.2 Preprocessing

The Preprocessing steps are the extraction of planar primitives in the DSM and the detection of the edges in the DSM and the image. The figure 2b shows the borders of the extracted primitives and the gradient map on a DFC 2015 tile detail.

Planar primitive extraction There are several methods to extract planes in point clouds, structured or not. Among them, the most common are RANSAC[27] and region growing[3]. As there may be variation in the projection direction (wide image, mosaic ...), we want the primitives to be connected components and we choose to use a region growing algorithm. As region growing depends on the normal quality at each point, we use the method from[4] as it estimates well normal around sharp edges.

Edge map generation In the DSM, the points are on edges if they belong to an extracted primitive and are adjacent to a point not in the same primitive. To detect the edges in the image, we first regularize it using the Rolling Guidance Filter[28]. Then we simply compute the gradient. The figure 2b (right) shows the result of the edge extraction process in the DSM and the image.

3.3 Optimization

The main challenge of the proposed method is to be able to minimize the energy E. If ψ in equation (1) is a metric, we have a Metric Labeling problem. Metric Labeling problem is directly linked to Markov Random Field theory[14]. Optimizing energy of MRFs has been a very active field of research over the past years[5,7]. In this paper we chose to use the method from Komodakis and Tziritas[15].

The figure 2d shows the influence of the optimization on the direction projection. The Lab color map is used for representation. The more two colors are different, the more the angle between the projection directions is wide. The left picture is the result considering only E_{data}, there is no uniformity. On the right is the result of the whole optimization scheme, the directions are, as expected, piece-wise uniform.

3.4 Results

Figure 2c shows the results obtained by the proposed method. The occluded pixels are shown in black. As expected occlusions are well detected and the DSM pixels are given a more coherent color, even in the mosaic case. Projection errors still can be observed on some parts of the images. They are mainly due to the simplifications of the problem without simplification. The discretization of the direction space while making the problem easier to solve, may leave the true projection direction unreachable. The second bias introduced is the unique direction given to segments. The camera viewpoint is not at infinite distance and the direction differ at each pixel. This approximation will be valid on patches of the original image, but it may suffer on complete images.

4 Dense prediction for semantic segmentation

Standard CNN classifiers are powerful tools but are not designed for pixel-wise labeling. By altering standard CNN models, we can build new networks able to produce a dense classification map rather than a flat classification[19]. Those networks have a fully-convolutional structure instead of the standard succession of convolutional and fully-connected layers (cf. Fig. 3). We can train in a fully-supervised manner an end-to-end network able to segment the data into semantic regions. These fully-convolutional networks (FCN) have been proven to be highly effective on several computer vision data-sets and we show they are now also the state-of-the-art on remote sensing data.

4.1 Deep network architecture

We use the SegNet[2] architecture to illustrate how FCN architectures can improve semantic segmentation of Earth Observation data. SegNet uses an encoder-decoder architecture (cf. Fig. 3) based on VGG-16 from Oxford's Visual Geometry Group[6]. As

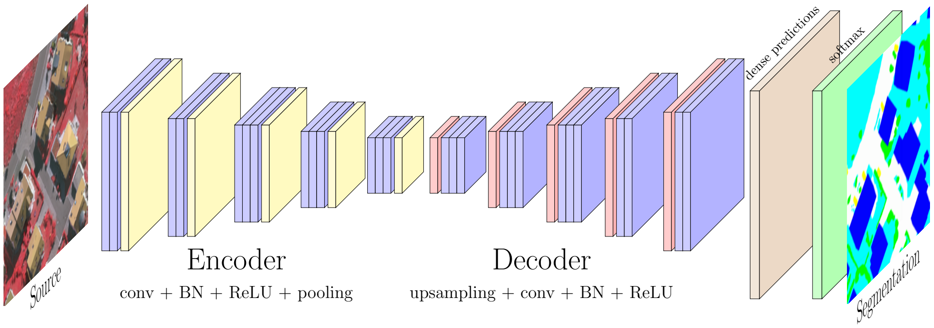


Figure 3. Fully convolutional architecture for semantic segmentation (SegNet[2]) of remote sensing data extracted from the ISPRS Vaihingen dataset.

in the VGG topology, convolutions are followed by a batch normalization[13] and a ReLU ($max(0,x)$). Each block of 2 or 3 convolutions ends with a max-pooling layer of stride 2. Similarly, the decoder has a symmetrical topology where pooling layers are replaced by unpooling operations. The unpooling layer takes as input the activations from the previous layer and a mask of indices from its associated pooling layer. These indices are the positions in the feature maps of the maximum activations in input of the pooling layer. The unpooling then upsamples its input feature maps by relocating the activations in the maximum positions and padding with zeroes everywhere else. This allows to relocate highly abstracted activations at the saliency points detected by the low level filters, thus increasing the spatial accuracy of the segmentation.

SegNet weights are initialized using VGG-16 trained on ImageNet and the decoder weights are randomly initialized using the MSRA strategy[11]. This supports the idea from[16,20,23] that generic visual filters learned on ImageNet can be used successfully for processing remote sensing data, even though the tiles from ISPRS data-set over Vaihingen are IR/R/G images and not regular RGB pictures. We train the network using Stochastic Gradient Descent (SGD) with a learning rate of 0.1 and a momentum of 0.9, and we divide the learning rate by 10 every 5 epochs.

4.2 Results

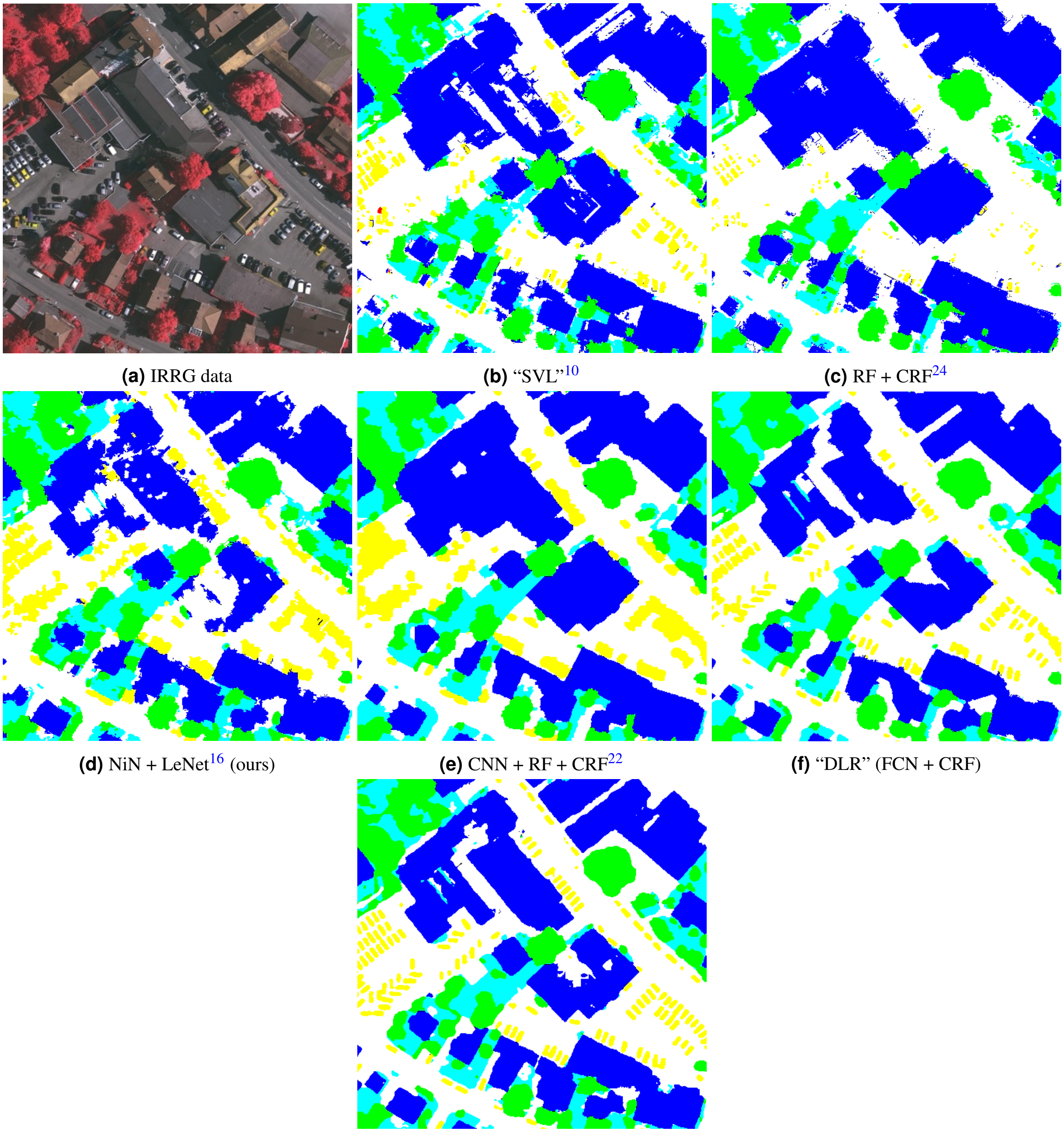
In order to process a full tile (approximately $2500 * 2500$) from the ISPRS data-set, we move a sliding window across the image to extract 128×128 patches with a stride of 32px. This overlap allows us to regularize the segmentation along the borders by averaging several predictions over one pixel. Processing one tile takes only a few minutes with a GPU.

Qualitative comparison of several methods is illustrated by the Fig.4. The deep fully convolutional network generates a much more visually appealing semantic map. Compared to traditional frameworks such as hand-crafted features and random forests[24] (RF), transitions between two classes are more precise and a lot smoother.

On a validation set, the overall accuracy reaches 92.5% and a F1 score of 0.90 on cars (to compare with the results from Tab.1). Compared to our previous work using superpixels segmentation and an SVM classifier trained on deep features[16] (cf. Sec.2), SegNet predictions are more detailed, especially on cars where each instance is clearly segmented. In comparison to recent work also using a FCN, we find that SegNet helps refining the prediction on cars but is also more precise on buildings, confusing less often roads and buildings than other methods. Moreover, previous state-of-the-art frameworks used computationally expensive structured models such as Conditional Random Fields (CRF) to regularize their predictions during post-processing. Our SegNet-based model outperforms those methods without requiring such structured models or the use of hand-crafted features.

5 Conclusion

We presented three new approaches for producing better Earth-observation products. First, we use deep convolutional neural networks and support vector machines for multi-sensor and multi-scale classification, which allow to produce accurate thematic maps. Second, we estimate the projection directions of each pixel of orthophotos using discrete optimization, in order to remove geometric aberrations in the new, resulting orthophotos. Third, we perform dense prediction over aerial images with fully-convolutional neural networks, which yields in precise segmentation maps. Our results show that deep learning allows re-thinking the remote sensing of areas with abundant information and offers promising paths for urban monitoring and modeling.



(a) IRRG data

(b) “SVL”[10]

(c) RF + CRF[24]

(d) NiN + LeNet[16] (ours)

(e) CNN + RF + CRF[22]

(f) “DLR” (FCN + CRF)

(g) SegNet (ours)

Figure 4. Segmentations from several methods on an extract of the ISPRS testing set of Vaihingen

6 Acknowledgement

The Vaihingen data set was provided by the German Society for Photogrammetry, Remote Sensing and Geoinformation (DGPF)[9] : http://www.ifp.uni-stuttgart.de/dgpf/DKEP-Allg.html. N. Audebert's work is funded by ONERA-TOTAL research project Naomi.

References

1. Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurélien Lucchi, Pascal Fua, and Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 34(11):2274–2282, 2012.

2. Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *arXiv preprint arXiv:1511.00561*, 2015.

3. P. J. Besl and R. C. Jain. Segmentation through variable-order surface fitting. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 10(2):167–192, 1988.

4. A. Boulch and R. Marlet. Fast and robust normal estimation for point clouds with sharp features. *Computer graphics forum*, 31(5):1765–1774, 2012.

5. Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, 2001.

6. Ken Chatfield, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Return of the Devil in the Details: Delving Deep into Convolutional Nets. In *Proceedings of the British Machine Vision Conference*, pages 6.1–6.12. British Machine Vision Association, 2014.

7. C. Chekuri, S. Khanna, J. S. Naor, and L. Zosin. Approximation algorithms for the metric labeling problem via a new linear programming formulation. In *ACM-SIAM symposium on Discrete algorithms*, pages 109–118, 2001.

8. Yushi Chen, Zhouhan Lin, Xing Zhao, Gang Wang, and Yanfeng Gu. Deep learning-based classification of hyperspectral data. *Selected Topics in Applied Earth Observations and Remote Sensing, IEEE Journal of*, 7(6):2094–2107, 2014.

9. M. Cramer. The dgpf test on digital aerial camera evaluation – overview and test design. *Photogrammetrie – Fernerkundung – Geoinformation*, 2:73–82, 2010.

10. Markus Gerke. Use of the Stair Vision Library within the ISPRS 2d Semantic Labeling Benchmark (Vaihingen). Technical report, International Institute for Geo-Information Science and Earth Observation, 2015.

11. Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1026–1034, 2015.

12. IEEE GRSS DFTC. 2015 IEEE GRSS data fusion contest. http://www.grss-ieee.org/community/technical-committees/data-fusion, 2015. Accessed: 2015-02-02.

13. Sergey Ioffe and Christian Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *Proceedings of The 32nd International Conference on Machine Learning*, pages 448–456, 2015.

14. J. Kleinberg and E. Tardos. Approximation algorithms for classification problems with pairwise relationships: Metric labeling and markov random fields. *Journal of the ACM*, 49(5):616–639, 2002.

15. N. Komodakis and G. Tziritas. Approximate labeling via graph cuts based on linear programming. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 29(8):1436–1453, 2007.

16. Adrien Lagrange, Bertrand Le Saux, Anne Beaupère, Alexandre Boulch, Adrien Chan-Hon-Tong, Stéphane Herbin, Hicham Randrianarivo, and Marin Ferecatu. Benchmarking classification of earth-observation data: from learning explicit features to convolutional networks. In *Proc. of IGARSS'2015*, Milano, Italy, 2015.

17. Y. LeCun, L. Bottou, Y. Bengio, , and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

18. Min Lin, Qiang Chen, and Shuicheng Yan. Network in network. *CoRR*, abs/1312.4400, 2013.

19. J. Long, E. Shelhamer, and T. Darell. Fully convolutional networks for semantic segmentation. In *Proc. of Computer Vision and Pattern Recognition Conf.*, 2015.

20. D. Marmanis, M. Datcu, T. Esch, and U. Stilla. Deep Learning Earth Observation Classification Using ImageNet Pretrained Networks. *IEEE Geoscience and Remote Sensing Letters*, 13(1):105–109, January 2016.

21. Volodymyr Mnih and Geoffrey Hinton. Learning to detect roads in high-resolution aerial images. In *Proc. of European Conf. on Computer Vision (ECCV)*, Crete, Greece, 2010.

22. S. Paisitkriangkrai, J. Sherrah, P. Janney, and A. Van-Den Hengel. Effective semantic pixel labelling with convolutional networks and conditional random fields. In *Proc. of CVPRw/Earth-Vision*, Boston, MA, 2015.

23. O. Penatti, K. Nogueira, and J. Dos Santos. Do deep features generalize from everyday objects to remote sensing and aerial scenes domains? In *Proc. of CVPRw/Earth-Vision*, Boston, MA, 2015.

24. Nguyen Tien Quang, Nguyen Thi Thuy, Dinh Viet Sang, and Huynh Thi Thanh Binh. An Efficient Framework for Pixel-wise Building Segmentation from Aerial Images. In *Proceedings of the Sixth International Symposium on Information and Communication Technology*, page 43. ACM, 2015.

25. Adriana Romero, Carlo Gatta, and Gustavo Camps-Valls. Unsupervised deep feature extraction of hyperspectral images. In *Proc. of WHISPERS*, Lausanne, Switzerland, 2014.

26. Franz Rottensteiner, Gunho Sohn, Markus Gerke, and Jan Dirk Wegner. Journal of Photogrammetry and Remote Sensing*: Special issue on Urban object detection and 3D building reconstruction*, volume 93. Elsevier, July 2014.

27. R. Schnabel, R. Wahl, and R. Klein. Efficient RANSAC for Point-Cloud Shape Detection. *Computer graphics forum*, 26(2):214–226, 2007.

28. Qi Zhang, Xiaoyong Shen, Li Xu, and Jiaya Jia. Rolling guidance filter. In *Computer Vision–ECCV 2014*, pages 815–830. 2014.