ON THE USABILITY OF DEEP NETWORKS FOR OBJECT-BASED IMAGE ANALYSIS

Nicolas Audeberta,b, Bertrand Le Sauxc, Sébastien Lefèvreb

a ONERA, The French Aerospace Lab, F-91761 Palaiseau, France (nicolas.audebert,bertrand.le_saux)@onera.fr
b Univ. Bretagne-Sud, UMR 6074, IRISA, F-56000 Vannes, France - sebastien.lefevre@irisa.fr

KEY WORDS: deep learning, vehicle detection, semantic segmentation, object classification

ABSTRACT:

As computer vision before, remote sensing has been radically changed by the introduction of Convolution Neural Networks. Land cover use, object detection and scene understanding in aerial images rely more and more on deep learning to achieve new state-of-the-art results. Recent architectures such as Fully Convolutional Networks (Long et al., 2015) can even produce pixel level annotations for semantic mapping. In this work, we show how to use such deep networks to detect, segment and classify different varieties of wheeled vehicles in aerial images from the ISPRS Potsdam dataset. This allows us to tackle object detection and classification on a complex dataset made up of visually similar classes, and to demonstrate the relevance of such a subclass modeling approach. Especially, we want to show that deep learning is also suitable for object-oriented analysis of Earth Observation data. First, we train a FCN variant on the ISPRS Potsdam dataset and show how the learnt semantic maps can be used to extract precise segmentation of vehicles, which allow us studying the repartition of vehicles in the city. Second, we train a CNN to perform vehicle classification on the VEDAI (Razakarivony and Jurie, 2016) dataset, and transfer its knowledge to classify candidate segmented vehicles on the Potsdam dataset.

1. INTRODUCTION

Deep learning for computer vision grows more popular every year, especially thanks to Convolutional Neural Networks (CNN) that are able to learn powerful and expressive descriptors from images for a large range of tasks: classification, segmentation, detection … This ubiquity of CNN in computer vision is now starting to affect remote sensing as well, as they can tackle many tasks such as land use classification or object detection in aerial images. Moreover, new architectures have appeared, derived from Fully Convolutional Networks (Long et al., 2015), able to output dense pixel-wise annotations and thus able to achieve fine-grained classification. Such architectures have quickly become state-of-the-art for popular datasets such as PASCAL VOC2012 (Everingham et al., 2014) and Microsoft COCO (Lin et al., 2014). In an Earth Observation context, these FCN models are now especially appealing, as dense prediction allows us performing semantic mapping without requiring any pre-processing tricks. Therefore, using FCN for Earth Observation means we can shift from superpixel segmentation and region-based classification (Lagrange et al., 2015; Audebert et al., 2016; Nogueira et al., 2016) to fully supervised semantic segmentation (Marmanis et al., 2016).

FCN models have been successfully applied for remote sensing data analysis, notably land cover mapping on urban areas (Marmanis et al., 2016; Paisitkriangkrai et al., 2015). For example, FCN-based models are now the state-of-the-art on the ISPRS Vaihingen Semantic Labeling dataset (Rottensteiner et al., 2012; Cramer, 2010). Therefore, even though remote sensing images do not share the same structure as natural images, traditional computer vision deep networks are able to successfully extract semantics from them, which was already known for deep CNN-based classifiers (Penatti et al., 2015). This encourages us to investigate further: can we use deep networks to tackle an especially hard remote sensing task, namely object segmentation? Therefore, this work focuses on using deep convolutional models for segmentation and classification of vehicles using optical remote sensing data.

To tackle this problem, we design a two-step pipeline for segmentation and classification of vehicles in aerial images. First, we use the SegNet architecture (Badrinarayanan et al., 2015) for semantic segmentation on the ISPRS Potsdam dataset. This allows us generating a pixel-level mask on which we can extract connected components to detect the vehicle instances. Then, using a CNN trained on vehicle classification using the VEDAI dataset (Razakarivony and Jurie, 2016), we classify each instance to infer the vehicle type and to eliminate false positives. We then show how to exploit this information to provide new pieces of data about vehicle types and vehicle repartition in the scene.

Our work is closely related to (Marmanis et al., 2016; Paisitkriangkrai et al., 2015) who also use Fully Convolutional Network for urban area mapping. However we focus only on small objects, namely vehicles, although we use the full semantic mapping as an intermediate product in our pipeline. Moreover, regarding (Paisitkriangkrai et al., 2015), our work rely solely on the deeply learnt representation of the optical data, as we do not include any expert features in the process. On the vehicle detection task, (Chen et al., 2014) tried to detect vehicle in satellite images using deep CNN, but only regressed the bounding boxes. On the contrary, our pipeline is able to infer the precise object segmentation, which could be regressed into a bounding box if needed. In addition, we also classify the vehicles into several subcategories using a classifier trained on a more generic dataset, thus performing transfer learning.

2. PROPOSED METHOD

2.1 Semantic segmentation

Many deep network architectures are available for semantic segmentation, including the original FCN (Long et al., 2015) or many variants such as DeepLab (Liang-Chieh et al., 2015). We choose to use the SegNet architecture (Badrinarayanan et al., 2015), since it is well-balanced between accuracy and computational cost. SegNet’s symmetrical structure and its use of pooling/unpooling layers is very effective for precise relocation of features according to its authors. Indeed, preliminary tests underlined that SegNet performs very well, including for small object localization. Results with other architectures such as FCN reported either no
improvement or an unimportant improvement. However, note
that any deep network trained for semantic segmentation can be
used in our pipeline.

SegNet is based on the convolutional layers of the VGG-16 model
(Simonyan and Zisserman, 2014). VGG-16 was designed for the
ILSRVC competition and trained on the ImageNet dataset (Russ-
sakovskyy et al., 2015). Each block of the encoder is comprised of
2 or 3 convolutional layers followed by batch normalization (Ioffe
and Szegedy, 2015) and rectified linear units. A maxpooling layer
reduces the dimensions between each block. In the decoder, sym-
metrical operations are applied in the reverse order. The unpool-
ing operation replaces the pooling in the decoder: it relocates the
value of the activations into the mask of the maximum values
"(argmax") computed at the pooling stage. Such an upsampling
results in a sparse activation map that is densified by the con-
secutive decoding convolutions. This allows us upsampling the
feature activations up to the original input size, so that after the
decoder, the final output feature maps have the same dimensions
as the input, which is necessary for pixel-level inference.

We initialize the weights of the encoder using VGG-16’s weights
trained on ImageNet. By doing so, we leverage the expressiv-
ness of the filters learned by VGG-16 when training on this very
diverse dataset. This initialization allows us improving the final
segmentation accuracy compared to a random initialization and
also makes the model converge faster.

We train SegNet on semantic segmentation on the ISPRS Pots-
dam dataset. This translates to pixel-wise classification, i.e each
pixel is classified as belonging to one class from the ISPRS bench-
mark: “imperious surface”, “building”, “tree”, “low vegetation”,
“car” or “clutter”. From this semantic map, we can then extract
the vehicle mask which labels all the pixels inferred to belong to
the “car” class.

2.2 CNN classification

After generating the full semantic map from one tile, we can ex-
tract the vehicle mask. We perform connected components ex-
traction to find all candidate vehicles in the predicted map. To
separate vehicles that are very close to each other in the RGB im-
age and that might belong to the same connected component, we
use a morphological opening to open the gap and separate the ob-
jects. In addition, to smooth the predictions and eliminate small
artifacts and imperfections from SegNet’s predicted map, we re-
move connected components that are too small to be a vehicle
(surface < 32 px). For each vehicle (any remaining connected
component), we extract a rectangular patch around its bounding
box, including 16 px of spatial context on all sides. This patch is
then fed to a CNN classifier trained to infer the vehicle’s precise
class. The full pipeline is illustrated in Figure 1.

We train these models, so that the Potsdam vehicles will be entirely
unseen data. This is to show how knowledge learnt by deep net-
works on remote sensing data can be transferred efficiently from
one dataset to another, under the assumption that provided resolu-
tions (12.5cm) and data sources (RGB) are the same. The clas-
sifiers learn to discriminate between the following 11 classes of
vehicles from VEDAI, with a very high variability: “car”, “swept-
car”, “tractor”, “truck”, “bike”, “van”, “bus”, “ship”, “plane”,
“pick up” and “other vehicles”.

LeNet is a model introduced by (Lecun et al., 1998) designed for
color image. VGG-16 is the biggest of our three models with
few parameters (≃ 461M parameters) that can be trained
very quickly.

AlexNet is a very popular model introduced in (Krizhevsky et
al., 2012) that won the ILSVRC challenge in 2012. It takes in
input a 224 × 224 color image. As there are three full finally
connected layers, AlexNet is a relatively big network with 61M
parameters. We fine-tune the last layer of the reference ImageNet
trained implementation of AlexNet.

Finally, VGG-16 is another popular model designed by (Simonyar
and Zisserman, 2014), on which is based the SegNet segmenta-
tion network introduced in Section 2.1. It outperformed AlexNet
on the ImageNet benchmark in 2014. It takes in input a 224 × 224
color image. VGG-16 is the biggest of our three models with
138M parameters. Once again, we fine-tune only the last layer of
the reference weights of VGG-16 trained on ImageNet. It is the
biggest and slowest of the three models tested in this work.

3. EXPERIMENTS

3.1 Experimental setup

3.1.1 VEDAI This VEDAI dataset (Razakarivony and Jurie,
2016) is comprised of 1268 RGB tiles (1024 × 1024 px) and the
associated IR image at 12.5 cm spatial resolution. For each tile,
annotations are provided detailing the position of the vehicles and
the associated bounding box (defined by its 4 corners). We cross-
validated the results on three of the suggested splits of the dataset
for the training and testing sets. The training set for the CNN is
built by extracting square patches around each vehicle bounding
box, padded with 16 px of spatial context on each side. We use
data augmentation to increase the number of samples by including
rotated (π, 2π, and 4π) and mirrored versions of the patches
containing vehicles.

We train each model during 50 epochs (≃ 1000000 iterations)
with a batch size of respectively 128 for LeNet and AlexNet and
32 for VGG-16, and a learning rate of 0.001, divided by 10 after
30 epochs. Training takes about 25 minutes for LeNet, 60 min-
utes for AlexNet and 10 hours for VGG-16 on NVIDIA K20c
GPU.
We build our segmentation training set by sliding a window of object segmentation on the high resolution tile. We downsample the resolution from 5 cm/pixel to 12.5 cm/pixel. For a fair comparison of the two datasets (ISPRS Potsdam and VEDAI) in the same framework, we only use the RGB data and downsample the resolution from 5 cm/pixel to 12.5 cm/pixel. However, note that SegNet would perform even better for small object segmentation on the high resolution tile.

We build our segmentation training set by sliding a window of 128×128 px over each high resolution tile, including an overlap of 75% (32 px stride). For this experiment, we use all the classes from the ground truth. This means that we not only train the model to predict the vehicle mask, but also to assign a label to each pixel according to the ISPRS classes, except “clutter”: “impervious surface”, “building”, “tree”, “low vegetation” and “car”. The model is trained by Stochastic Gradient Descent for 10 epochs (≈100000 iterations) with a batch size of 10 and a learning rate of 0.1, divided by 10 after 3 and 8 epochs. This takes around 24 hours with a NVIDIA K20c GPU.
Counting cars We extract the vehicle mask from the semantic map or ground truth of each car present in the scene. We trained several deep CNN on the ISPRS Potsdam dataset showing very promising results on this specific task.

Table 2. Vehicle counts in testing tiles of ISPRS Potsdam

<table>
<thead>
<tr>
<th>Tile #</th>
<th>2_11</th>
<th>7_12</th>
<th>3_11</th>
<th>5_12</th>
<th>7_10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cars in ground truth</td>
<td>110</td>
<td>351</td>
<td>168</td>
<td>428</td>
<td>253</td>
</tr>
<tr>
<td>Predicted cars</td>
<td>115</td>
<td>342</td>
<td>182</td>
<td>435</td>
<td>257</td>
</tr>
</tbody>
</table>

This is supported by qualitative inspection of the semantic maps on car-heavy areas, such as the one presented in Figure 5.

The mean Intersection over Union (IoU) reaches 75.6%. However, there is a high variance in the IoU depending on the specific surroundings of each car. Indeed, because of occlusions due to trees, the ground truth sometimes is labelled as tree even if the car is visible in the RGB image. This causes SegNet to misclassify (according to the ground truth) most of the pixels, resulting in a low IoU despite the visual quality of the segmentation. This phenomenon is illustrated by Figure 6. However, as SegNet manages to still infer the car’s contours, this should not impede too severely later vehicle extraction and classification.

Classification Using our AlexNet-based CNN classifier, we are able to classify the candidate vehicle instances generated by SegNet on the Potsdam dataset. We compare the predicted results to our enhanced vehicle ground truth. On our testing set, we classify correctly 75.0% of the instances. We are able to improve even further this result by switching to the VGG-16 based classifier which brings accuracy to 77.3% (cf. Table 3 for detailed results). It should be noted that the dataset is predominantly comprised of cars, and that trucks and pick ups are less represented, which decreases their influence on the global accuracy score. Figure 5 illustrates some vehicle instances where our deep network-based segmentation and classification pipeline was successful, while Figure 6 shows some examples of correct segmentation but subsequent misclassification.

The fact that the average accuracy of the models on Potsdam are lower than results reported on VEDAI suggests that our networks suffered from overfitting. One hypothesis that we make is that VEDAI and Potsdam are images taken from two similar but subtly different environments. Indeed, Potsdam is a urban European city whereas VEDAI images have been shot over Utah, on a more rural American environment. Therefore, vehicle brands and different land covers around the cars might influence the classifiers.

We also show that we can use this information to generate heat maps of the vehicle repartition in the tile (cf. Figure 7). Such a visualization provides an indicator of the vehicle density in the urban area, and exacerbates high traffic roads and “hot spots” corresponding to parking lots around points of interest (e.g. hospitals or shopping malls). This can be useful for many urban planning applications, such as parking lot sizing, traffic estimation, etc. In temporal data, this could be used to monitor traffic flows, but also study the impact of the road blocks, jams, etc.

In this work, we presented a two-step framework to detect, segment and classify vehicles from aerial RGB images using deep learning. More precisely, we showed that deep network designed for semantic segmentation such as SegNet are useful for scene understanding of remote sensing data and can be used to segment even small objects, such as cars and trucks. We reported results on the ISPRS Potsdam dataset showing very promising results on wheeled vehicles, achieving a F1 score of 77.3% on this specific class.

In addition, we presented a simple deep learning based method to improve further this analysis by classifying the different types of vehicle present in the scene. We trained several deep CNN on the VEDAI dataset and transferred their knowledge to the Potsdam dataset. Our best model, fine-tuned from VGG-16, was able to classify successfully more than 77% of the vehicles in our testing set.

Finally, this work meant to provide useful pointers for applying deep learning to scene understanding in Earth Observation with an object-oriented approach. We showed that not only deep networks are the state-of-the-art for semantic mapping, but that they can also be used to extract useful information at an object level, with a direct application on vehicle detection and classification.

4. CONCLUSION

Table 3. Classification results on the enhanced Potsdam vehicle ground truth

<table>
<thead>
<tr>
<th>Class</th>
<th>Car</th>
<th>Van</th>
<th>Truck</th>
<th>Pick up</th>
<th>Global</th>
</tr>
</thead>
<tbody>
<tr>
<td>AlexNet</td>
<td>81%</td>
<td>42%</td>
<td>50%</td>
<td>50%</td>
<td>75%</td>
</tr>
<tr>
<td>VGG</td>
<td>83%</td>
<td>55%</td>
<td>50%</td>
<td>33%</td>
<td>77%</td>
</tr>
</tbody>
</table>

In this study, we showed that it is possible to enumerate and extract individual object instances from the semantic map in order to analyze the vehicle distribution in the images, leading to the localization of points of interest such as high traffic roads and parking lots. This has many applications in traffic monitoring and urban planning, such as analyzing parking lots occupancy, finding pollutants vehicles in unauthorized zones (combined with a classifier), etc.
ACKNOWLEDGEMENTS

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

The Vehicle Detection in Aerial Imagery (VEDAI) dataset was provided by S. Razakarivony and F. Jurie.

N. Audebert’s work is funded by ONERA-TOTAL research project Naomi.

The authors would like to thank A. Boulch and A. Chan Hon Tong for fruitful discussions on object detection and classification.

REFERENCES


Figure 6. Visualizing vehicles in the ISPRS Potsdam dataset (best viewed in color)

