Deep Learning based Vehicle Detection in Aerial Imagery

Lars Sommer

Vision and Fusion Laboratory
Institute for Anthropomatics
Karlsruhe Institute of Technology (KIT), Germany
lars.sommer@kit.edu

Technical Report IES-2017-06

Abstract: Detecting vehicles in aerial images is an important task for many applications like traffic monitoring or search and rescue work. In recent years, several deep learning based frameworks have been proposed for object detection. However, these detection frameworks were developed and optimized for datasets that exhibit considerably differing characteristics compared to aerial images, e.g. size of objects to detect. In this report, we demonstrate the potential of Faster R-CNN, which is one of the state-of-the-art detection frameworks, for vehicle detection in aerial images. Therefore, we systematically investigate the impact of adapting relevant parameters. Due to the small size of vehicles in aerial images, the most improvement in performance is achieved by using features of shallower layers to localize vehicles. However, these features offer less semantic and contextual information compared to features of deeper layers. This results in more false alarms due to objects with similar shapes as vehicles. To account for that, we further propose a deconvolutional module that up-samples features of deeper layers and combines these features with features of shallower layers.

1 Introduction

Vehicle detection in aerial images is an important task for many applications like traffic monitoring or search and rescue work. Conventional approaches applied to detect vehicles in aerial images are generally comprised of hand-crafted features and a classifier within a sliding window approach [LM15, CH16, MM14]. In recent years, several authors applied convolutional neural networks (CNNs) to extract features at each sliding window position [CXLP14, KPF16]. In [CXLP14], improved results are achieved for vehicle detection in satellite images by applying convolutional features instead of hand-crafted features. However, the computation of convolutional features for each candidate window separately is computational expensive [Gir15].

In recent years, deep learning based detection frameworks like Faster R-CNN [RHGS15], which achieves top performing results on common detection benchmark datasets, have been proposed to reduce the computational effort. Therefore, a convolutional feature map is computed for the entire image at once and shared for all candidate windows [Gir15, RHGS15]. However, such detection frameworks are developed and optimized for common detection benchmark datasets that exhibit considerably differing characteristics compared to aerial images, e.g. size of objects to detect.

In the context of this report, we demonstrate the applicability of Faster R-CNN for vehicle detection in aerial images. Therefore, several adaptions are performed to account for the characteristics of the aerial images and the impact on the detection performance is evaluated. The DLR 3K Munich Vehicle Aerial Image Dataset [LM15] that comprises objects in the range of 15×30 pixels is used for all experiments.

The main improvement is achieved by adapting the resolution of the output of the last convolutional layer, which is used as feature map to localize and classify objects. The resolution of the standard feature map is only 1/16 of the input image and consequently insufficient for object sizes between 15 and 30 pixels. To provide a sufficient feature map resolution, the output of shallower convolutional layers is used as feature map. However, these features offer less semantic and contextual information compared to features of deeper layers. This results in more false alarms due to objects with similar shapes as vehicles. To account for that, we further propose a deconvolutional module that up-samples features of deeper layers and combines these features with features of shallower layers.

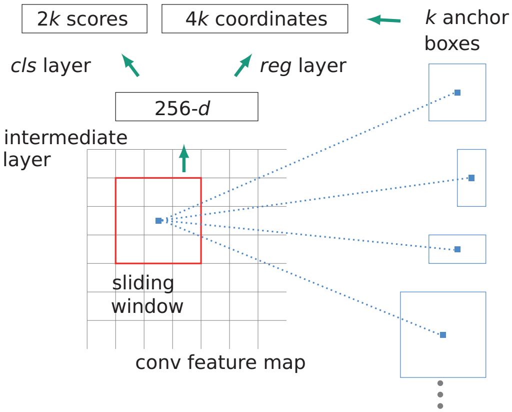

Figure 2.1: Schematic illustration of the Region Proposals Network (RPN) used to generate a set of candidate regions that are likely to contain an object.

2 Faster R-CNN

In the following, the functional principle of the Faster R-CNN detection framework as proposed by Ren et al. [RHGS15] is introduced. Faster R-CNN is comprised of two modules: an initial deep learning based object proposals method called Region Proposals Network (RPN) and the subsequent Fast R-CNN module [Gir15] used to classify the generated proposals. Both the RPN and the Fast R-CNN module share the convolutional layers to reduce the computational effort.

Figure 2.1 shows schematically the RPN. The RPN uses the output of the last convolutional layers as feature map. Then, a small network is shifted over the feature map to generate a set of candidate regions. The small network comprises a 3×3 convolutional layer followed by a classification layer (*cls* layer) and a bounding box regression layer (*reg* layer). The classification layer outputs a confidence score at each position, which is used to rank the proposals. The bounding box regression layer is used to compute the corresponding coordinates. For this, a set of fixed scaled anchor boxes k are used as bounding box reference.

The top 300 region proposals (highest confidence score) are forwarded to the Fast R-CNN module. The Fast R-CNN module classifies each region proposal into various object classes or background. Therefore, each region proposal is projected onto the feature map. Then, the corresponding features are extracted by the so called Region of Interest (RoI) pooling layer to generate a vector of fixed length as required for the subsequent fully connected layers. After a sequence of fully connected layers, a classification layer and a bounding box regression layer are used for classification and to refine the coordinates of the corresponding candidate region, respectively.

3 Adaption to Aerial Images

The detection performance is mainly affected by adapting the resolution of the feature map used to compute proposals and for classification and by adapting the parameters of the RPN.

The original Faster R-CNN utilizes VGG-16 [SZ14] as base architecture. The VGG-16 comprises 13 convolutional layers with a kernel size of 3×3 followed by 3 fully-connected layers. To reduce the amount of parameters and to make the network invariant to small translations of the input, max-pooling layers are inserted after the 2nd (conv1_2), 4th (conv2_2), 7th (conv3_3), 10th (conv4_3), and 13th (conv5_3) convolutional layer. In case of Faster R-CNN, the output of the last convolutional layer is used as feature map. As illustrated in Figure 3.1, the dimensions of the feature map are only 1/16 of the dimensions of the input image. Thus, the feature map resolution is insufficient to accurately localize objects in the range of 15 to 30 pixels or even smaller. To account for that, we replace the initially used VGG-16 architecture by a network architecture optimized for handling small instances. The network is inspired by the network proposed in [HWB16] and comprises 4 convolutional layers followed by 3 fully connected layers. Max-pooling layers are inserted after the 1st, 2nd, and 4th convolutional layer. We performed optimization of all relevant network parameters including number of layers, number of filters per layer, kernel size and dropout. Analogous to the original Faster R-CNN, the output of the last convolutional layer is used as feature map. As depicted in Figure 3.2, the dimensions of the feature map are 1/4 of the dimensions of the input image. Thus, a finer localization of small objects is feasible due to the higher resolution of the feature map.

In addition to increasing the feature map resolution, adapting the parameters of the RPN mainly affects the detection performance. The benchmark datasets used

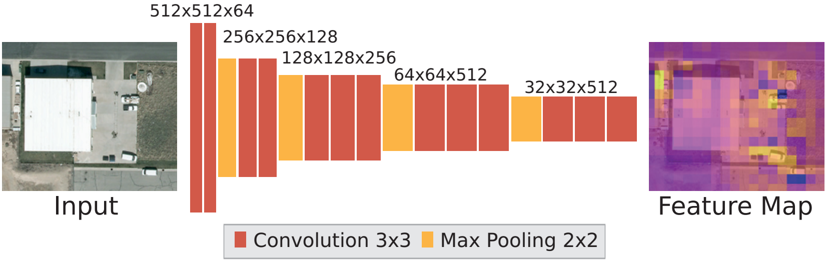

Figure 3.1: Schematic illustration of the convolutional part of VGG-16 and the resulting feature map.

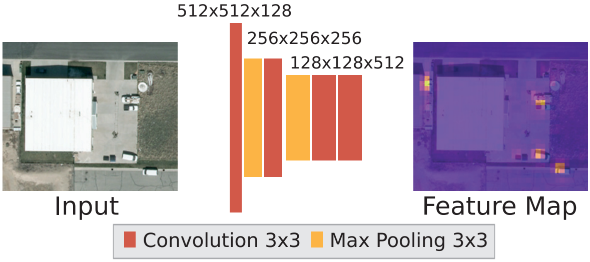

Figure 3.2: Schematic illustration of the convolutional part of the proposed network architecture optimized for handling small objects and the resulting feature map.

for developing Faster R-CNN contain objects that are generally in the range between 50 and 200 pixels. Thus, the parameters of the RPN are adjusted for these object dimensions. First, we reduce the minimal height and width of considered proposals (RPN_MI_SIZE) from 16 to 4, in order to account for the smaller object sizes in case of aerial images. Initially, the top 300 region proposals are considered for classification. This is enough to localize objects in the benchmark datasets, which generally contain only one or a few objects per image. Multiple proposals are typically located around the same object. Aerial images can contain clearly more objects per image and furthermore can contain more potentially disturbing objects, e.g. trailers or solar cells on buildings. Therefore, we set the number of proposals considered for classification to 2,000. As

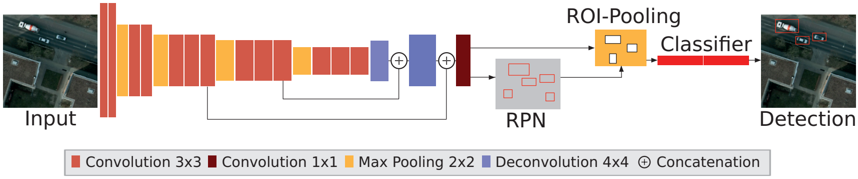

Figure 4.1: Schematic illustration of the Faster R-CNN extended by the deconvolutional module (DFRCNN).

described in Section 2, anchor boxes are used as reference for bounding box regression. The initially used anchor scales are chosen to account for the size of objects in the benchmark datasets. The ANCHOR_BASE_SIZE is set to 16 and the ANCHOR_SCALE factors are set to 8, 16 and 32, which results in anchor boxes with dimensions in the range between 128 and 512 pixels. We set the ANCHOR_BASE_SIZE to 2 while the ANCHOR_SCALE factors are kept unchanged.

4 Deconvolutional Module

To achieve a higher feature map resolution that is sufficient to localize small objects as in case of aerial images, small networks as described in Section 3 or shallow layers of standard architectures like VGG-16 are applicable. However, high-resolution feature maps offer less semantic and contextual information compared to features of deeper layers. The less semantic and contextual information make the detection framework more prone to false alarms due to objects with shapes similar to vehicles.

In order to achieve a high-resolution feature map and semantic and contextual informative features, we extend the Faster R-CNN by a deconvolutional module. The deconvolutional module up-samples low-dimensional feature maps of deep layers and combines the up-sampled features with the features of shallow layers while the feature map resolution is kept sufficiently high to localize small objects. The network architecture of the Faster R-CNN extended by the deconvolutional module (DFRCNN) is schematically illustrated in Figure 4.1. We use VGG-16 as base network architecture. First, the features of conv5_3 are up-sampled by a factor of 2 and then concatenated with the features of conv4_3. Then, the combined features are up-sampled by a factor of 2 and then concatenated with the features of conv3_3. Thus, the features of conv4_3 and conv5_3 are up-sampled by a factor of 2 and 4, respectively. We use deconvolutional layers with a kernel size of 4×4 and a stride of 2 to up-sample the features. The combined features of conv3_3, conv4_4 and conv5_3 are used as feature map. The feature map dimensions are 1/4 of the dimensions of the input image. To adapt the number of output channels of the feature map required as input for the fully connected layers, we insert an additional convolutional layer with kernel size 1×1.

5 Evaluation

In the following section, we evaluate the impact of the adaptions described in Section 3 and of the deconvolutional module proposed in Section 4. We use Average Precision (AP) computed as defined in [EVGW$^+$10], precision and recall as evaluation metrics. Ground truth (GT) objects are considered as recalled, if the Pascal-overlap criterion [EVGW$^+$10] is satisfied. For all experiments, we use the publicly available DLR 3K Munich Vehicle Aerial Image Dataset. The dataset comprises 20 aerial images with a resolution of 5616×3744 pixels and a ground sampling distance (GSD) of approximately 13 cm. Due to the limited memory capacity of the used GPUs, each image is divided into tiles of 936×624 pixels. Image sections are exemplarily depicted in Figure 5.1. We further align the provided GT annotations at image edges as required for the Faster R-CNN detection framework.

5.1 Adaption to Aerial Images

The impact of increasing the feature map resolution on the detection performance is shown in Figure 5.2. The blue line corresponds to the precision-recall curve for an IoU threshold value of 0.5 used to accept GT objects as recalled (PASCAL-criterion). In case of using the VGG-16 architecture (feature map 1/16), both precision and recall are considerably worse compared to using the optimized network architecture (feature map 1/4). Precision values close to 1 and recall values

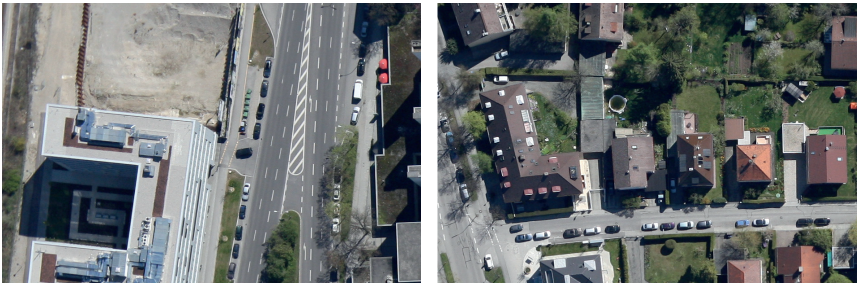

Figure 5.1: Image sections of the DLR 3K Munich Vehicle Aerial Image Dataset [LM15]

above 0.95 are achieved for the optimized network architecture. Reason for the improved performance is the higher feature map resolution as the detections are better localized around the GT elements. The better localization of the detections is illustrated in Figure 5.2 by plotting precision-recall curves for various IoU threshold values used to accept GT objects as recalled. For a resolution of 1/4 of the input image, the performance is only slightly decreasing with increasing IoU thresholds up to 0.5, which indicates a good localization of the detections. In contrast, the performance for lower resolutions decreases stronger with increasing IoU threshold values. The worse localization results in worse classification into object or background though the features comprise more semantic and contextual information. To highlight the difference in localization quality, qualitative detection examples are given for both feature map resolutions (see Figure 5.3 and Figure 5.4, respectively). For a resolution of 1/16 of the input image, the bounding box positions of the detections (red boxes) clearly differ from the GT annotations (green boxes). Furthermore, multiple detections are often generated due to the poor localization. In contrast, the detections for a feature map resolution of 1/4 of the input image overlap very well with the GT annotations.

The impact of adapting the RPN is illustrated in Figure 5.5 and Figure 5.6. Figure 5.5 depicts the proposals' quality for various anchor box sizes. Therefore, we plot the recall achieved for the proposals with respect to the IoU threshold value used to accept the GT objects as recalled. The mean anchor box dimensions are given in the legend. Reducing the anchor box sizes clearly improves the proposals' quality. For anchor box dimensions in the range between 14 and 28 pixels, which is roughly the size of present objects, the best recall values are

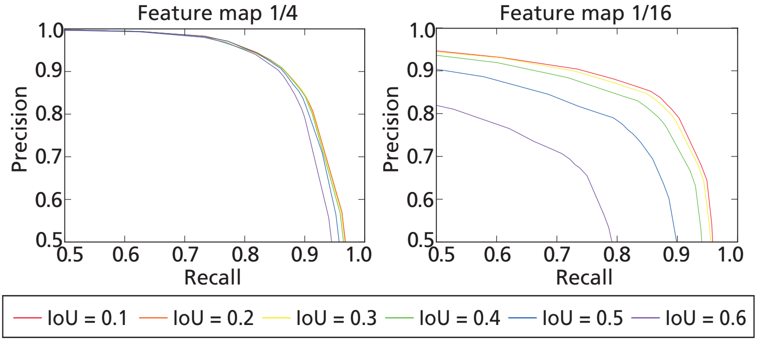

Figure 5.2: Precision-recall curves for various IoU threshold values used to accept GT objects as recalled. Higher feature map resolutions result in better localization quality as the performance decreases clearly less with increasing threshold values.

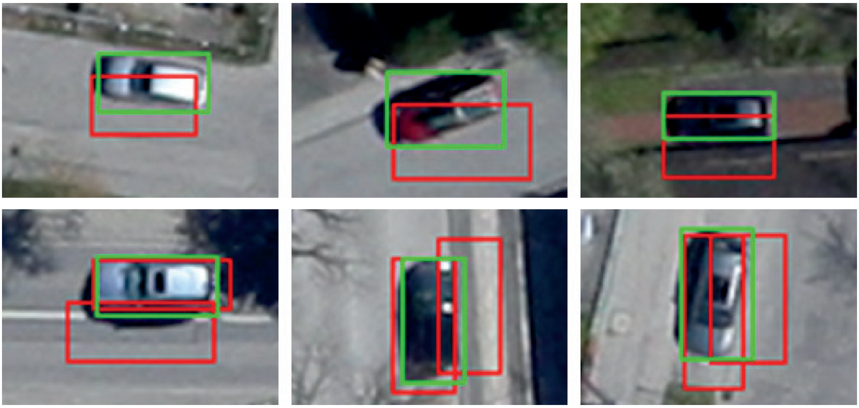

Figure 5.3: Qualitative detections (red boxes) and corresponding GT annotations (green boxes) for a feature map resolution of 1/16 of the input image. The detections show a relatively poor overlap with the GT annotations and multiple detections are often generated for one GT object.

achieved. The relation between proposals' quality and detection performance is shown in Figure 5.6. Therefore, we plot AP with respect to Average Best Overlap

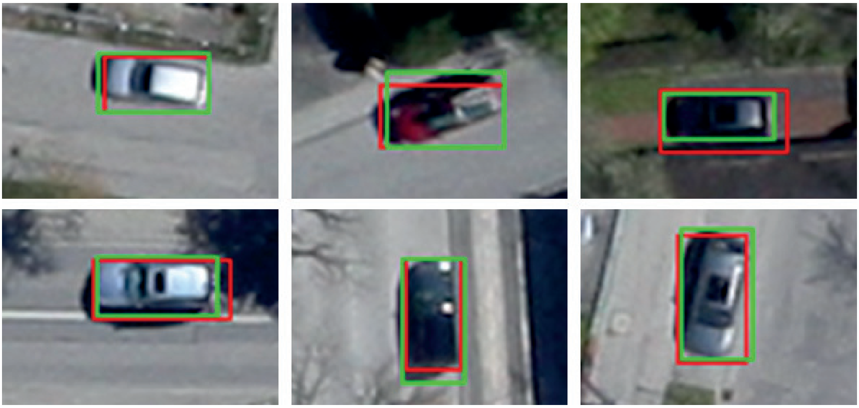

Figure 5.4: Qualitative detections (red boxes) and corresponding GT annotations (green boxes) for a feature map resolution of 1/4 of the input image. The detections overlap very well with the GT annotations.

(ABO), which is an evaluation metric for the localization quality. ABO is calculated by averaging the best overlap between each GT annotation $g_i \in G$ and the corresponding set of object proposals L:

$$ABO = \frac{1}{|G|} \sum_{g_i \in G} \max_{l_j \in L} \boldsymbol{IoU}(g_i, l_j)\,.$$

The best ABO is achieved for anchor box sizes in the range of present objects. The best AP is achieved for anchor boxes in the range of present objects as well. Thus, we assume that better proposals result in better detection performance.

To sum up the impact of the adaptions, the detection performance for both adaptions and the original Faster R-CNN is given in Figure 5.7. The performance of the original Faster R-CNN is poor. Both precision and recall are clearly less than 1. Applying the adapted RPN results in clearly improved precision and recall (VGG-16 adapted). However, the detection performance is still poor. Replacing the VGG-16 architecture with the optimized network architecture and consequently increasing the feature map resolution results in a significantly improved detection performance. It is to mention, that both adaptions are necessary to achieve the best detection results.

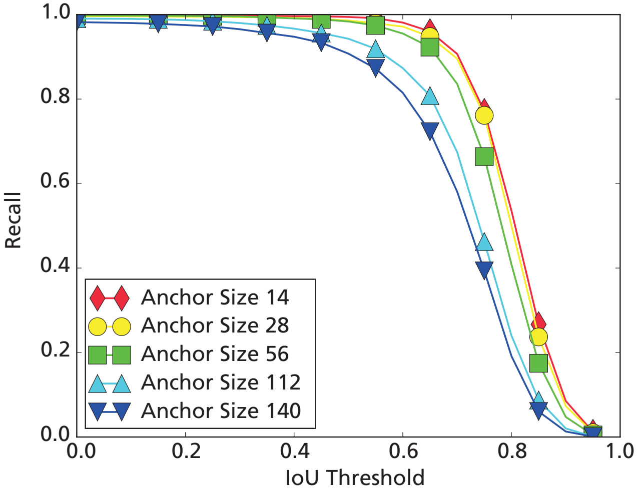

Figure 5.5: Recall-IoU curves for various anchor box sizes used as bounding box reference. The mean anchor box dimensions are given in the legend.

5.2 Deconvolutional Module

The impact of our proposed deconvolutional module on the detection performance is given in Table 5.1. We compare the detection results of our proposed Deconvolutional Faster R-CNN (DFRCNN) to baselines on the DLR 3K dataset for various GSDs. For this, we re-scaled the input images for training and testing by factors 1, 0.75, and 0.5. As baseline, we consider Faster R-CNN with different convolutional layers of VGG-16 used as feature map. For each GSD, the anchor box sizes are adapted for all Faster R-CNNs to the size of present objects. As discussed above increasing the feature map resolution from 1/16 of the input image (VGG-16 – conv5_3) to 1/4 of the input image (VGG-16 – conv3_3) clearly improves the detection performance especially for tiny objects as for a GSD of 26 cm. The performance is improved though the used features are less semantically and contextually informative. To account for the smaller receptive fields and less semantic information, we use our DFRCNN which combines features of conv3_3, conv4_3, and conv5_3 as described in Section 4. The performance is improved for all GSD especially for a GSD of 26 cm and consequently smaller

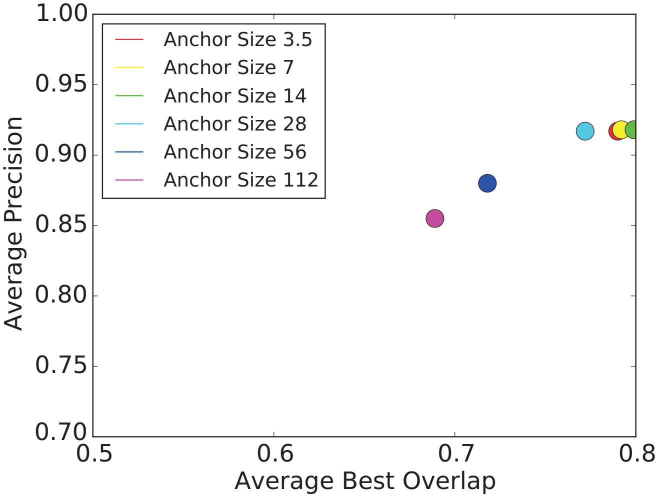

Figure 5.6: Average Precision (AP) w.r.t. Average Best Overlap (ABO) for various anchor box sizes used as bounding box reference. Applying region proposals with better ABO results in higher AP.

objects. In case of GSD 26, the number of false positive detections is reduced by a factor of 33.4% compared to VGG-16 — conv3_3, while the number of false negative detections remains almost unchanged.

To illustrate the impact of adding more semantic information, qualitative detection examples for Faster R-CNN using conv3_3 as feature map (left column) and

Table 5.1: Average Precision of our proposed DFRCNN compared to baselines on the DLR 3K dataset for various GSDs (in cm).

Method	GSD 13	GSD 19.5	GSD 26
VGG-16 — conv5_3	0.770	0.558	0.207
VGG-16 — conv4_3	0.967	0.896	0.601
VGG-16 — conv3_3	0.979	0.944	0.836
DFRCNN	**0.980**	**0.957**	**0.864**

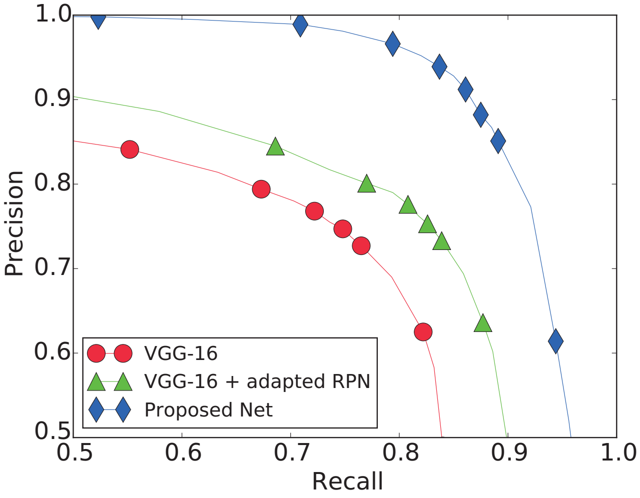

Figure 5.7: Precision-recall curves of the original Faster R-CNN and for both adaptions.

our proposed DFRCNN (right column) are given in Figure 5.8. Therefore, we use a classification threshold value of 0.5. For Faster R-CNN using conv3_3, several false positive detections are caused by objects with shapes similar to vehicles such as solar cells or chimneys on buildings. Integrating more semantic information clearly reduces the number of false positive detections caused by such objects.

6 Summary

In this report, the applicability of Faster R-CNN for vehicle detection in aerial images was demonstrated. Therefore, we have systematically evaluated the impact of adapting relevant parameters of Faster R-CNN to the characteristics of aerial images. The most improvement in detection performance was achieved by adapting the size of the anchor boxes used for bounding box regression and by increasing the feature map resolution as the initial resolution is insufficient to localize small objects. To achieve high feature map resolutions that are sufficient to localize small objects, small networks or shallow layers of standard architectures

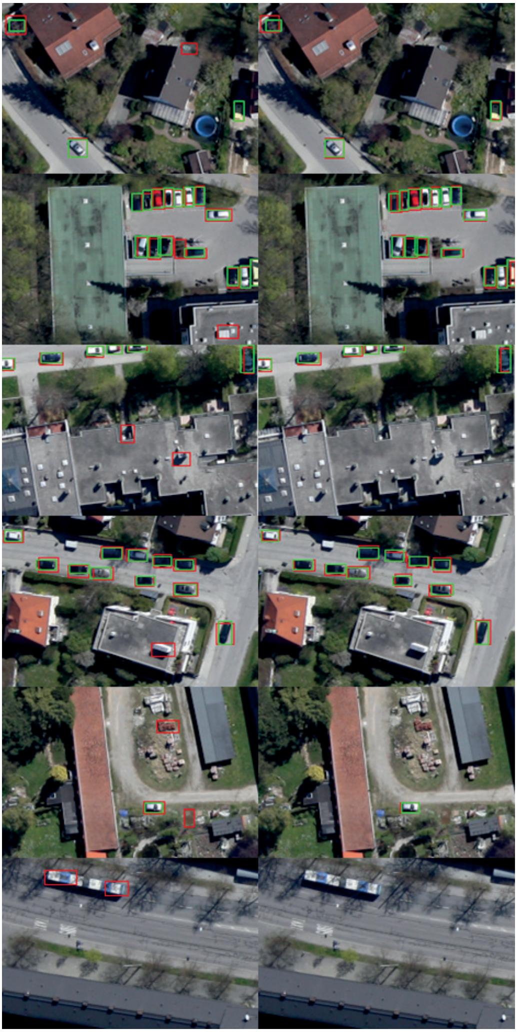

Figure 5.8: Qualitative detections (red boxes) and corresponding GT (green boxes) for Faster R-CNN using *conv3_3* (left column) and our proposed DFRCNN (right column) on DLR 3K indicate that false alarms due to objects with shapes similar to vehicles are reduced by integrating more semantic information.

like VGG-16 are applicable, which offer less semantic and contextual information compared to features of deeper layers. In order to overcome this drawback, we extended the original Faster R-CNN by a deconvolutional module. Therefore, features of deeper layers are up-sampled and combined with features of shallower layers. The detection performance is improved by integrating features with more semantic information especially for tiny objects as the number of false positive detections due to objects with shapes similar to vehicles is reduced.

Bibliography

[CH16] Gong Cheng and Junwei Han. A survey on object detection in optical remote sensing images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 117:11–28, 2016.

[CXLP14] Xueyun Chen, Shiming Xiang, Cheng-Lin Liu, and Chun-Hong Pan. Vehicle detection in satellite images by hybrid deep convolutional neural networks. *IEEE GRSL*, 11(10):1797–1801, 2014.

[EVGW$^+$10] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

[Gir15] Ross Girshick. Fast R-CNN. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1440–1448, 2015.

[HWB16] Christian Herrmann, Dieter Willersinn, and Jürgen Beyerer. Low-resolution convolutional neural networks for video face recognition. In *Advanced Video and Signal Based Surveillance*. IEEE, 2016.

[KPF16] Georgy V Konoplich, Evgeniy O Putin, and Andrey A Filchenkov. Application of deep learning to the problem of vehicle detection in UAV images. In *Soft Computing and Measurements (SCM), 2016 XIX IEEE International Conference on*, pages 4–6. IEEE, 2016.

[LM15] K. Liu and G. Mattyus. Fast multiclass vehicle detection on aerial images. *GRSL, IEEE*, PP(99):1–5, 2015.

[MM14] Thomas Moranduzzo and Farid Melgani. Detecting cars in UAV images with a catalog-based approach. *IEEE Transactions on Geoscience and Remote Sensing*, 52(10):6356–6367, 2014.

[RHGS15] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[SZ14] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.