SEGMENT-AND-COUNT: VEHICLE COUNTING IN AERIAL IMAGERY USING ATROUS CONVOLUTIONAL NEURAL NETWORKS

S. Azimi*, E. Vig, F. Kurz, P. Reinartz

German Aerospace Center (DLR), Remote Sensing Technology Institute, D-82234 Weßling, Germany -
(seyedmajid.azimi, eleonora.vig, franz.kurz, peter.reinartz)@dlr.de

Commission I, WG I/2

KEY WORDS: Vehicle Segmentation, Vehicle Counting, Aerial Imagery, Convolutional Neural Networks, Atrous Convolution

ABSTRACT:

High-resolution aerial imagery can provide detailed and in some cases even real-time information about traffic related objects. Vehicle localization and counting using aerial imagery play an important role in a broad range of applications. Recently, convolutional neural networks (CNNs) with atrous convolution layers have shown better performance for semantic segmentation compared to conventional convolutional aproaches. In this work, we propose a joint vehicle segmentation and counting method based on atrous convolutional layers. This method uses a multi-task loss function to simultaneously reduce pixel-wise segmentation and vehicle counting errors. In addition, the rectangular shapes of vehicle segmentations are refined using morphological operations. In order to evaluate the proposed methodology, we apply it to the public "DLR 3K" benchmark dataset which contains aerial images with a ground sampling distance of 13 cm. Results show that our proposed method reaches 81.58% mean intersection over union in vehicle segmentation and shows an accuracy of 91.12% in vehicle counting, outperforming the baselines.

1. INTRODUCTION

Vehicle segmentation and counting in aerial imagery is of significant importance, as aerial imagery can provide valuable information over a large area in a short period of time. The automatic analysis of such images to segment and count vehicles can yield valuable information for multiple applications such as traffic monitoring, parking lot detection and utilization, and urban management. In recent years, the advances in camera sensor technology have improved the resolution of remote sensing images, particularly airborne images. Therefore, thanks to the higher resolution, it is feasible to distinguish vehicles from each other. This is crucial in applications as the number of vehicles provides valuable insights over the captured area. Nevertheless, the small size of vehicles as well as different scales, shadow and complex background introduce considerable challenges in the success of current vehicle counting methods. Moving objects in airborne images, especially vehicles, appear to be of small size (*e.g.*, 10×20 px) depending on the ground sampling distance.

In the last few years, deep learning methods have shown impressive performance in different terrestrial imagery tasks, such as object detection (Redmon and Farhadi, 2017; Ren et al., 2015) and segmentation (Chen et al., 2018; Long et al., 2015). Inspired by this success, multiple remote sensing works have used *convolutional neural networks (CNNs)* (Máttyus et al., 2017; Tang et al., 2017; Audebert et al., 2017). Despite significant improvements made by CNNs, the majority of previous works apply minor modifications to the recent CNN architectures and use the main part of the architectures directly. However, these methods do not take into account the specific challenges encountered in aerial imagery. To address such challenges, a specialized architecture is required in aerial imagery instead of the direct employment of standard CNNs. One of these challenges is the small size of objects especially for densely populated areas (*e.g.*, vehicles in parking lots). For segmenting such dense groups of vehicles, the context plays an important role. Hu and Ramanan (2017) showed how important the context element is to detect small objects by CNNs. To integrate context data, CNNs employ stacked sub-sampling layers. Despite its usefulness in decreasing computation costs and in increasing the receptive field of the network, the resolution as one of the key elements is practically ignored. However, resolution is crucial to distinguish vehicles in a dense place. Moreover, feature resolution is lost gradually by applying sub-sampling (pooling) layers through consecutive network layers. Therefore, the resulting coarse feature maps do not carry details of object boundaries specifically for small objects which makes it hard for the network to recover them during the decoding step of the feature maps. Even deploying skip connections (Long et al., 2015) or hypercolumns (Hariharan et al., 2015) does not resolve this issue completely. Thus a dedicated method which expands the receptive field with no resolution loss is needed.

Recently, Yu and Koltun (2015) proposed an interesting variant of the convolution operation called atrous convolution (also called dilated convolution). In atrous convolutions, the atrous rate increases the arrangement of kernel parameters (weights). The larger the atrous rate is, the more sparse the kernel weights are arranged (i.e. the parameters point to input information in larger gaps). Hence, it is possible to increase the receptive field steadily by just stacking atrous convolutions on top of each other. This comes with the loss of resolution from the input data which is essential to recover small size vehicles. Atrous convolution has been used recently in different tasks in ground imagery showing promising performance.

In this work, we investigate the effect of atrous convolutions in improving current CNNs for semantic segmentation of vehicles in aerial images. We show that one should choose atrous rates in each layer separately and avoid steadily increasing atrous rate. The naive monotonous increase of atrous rates does not achieve

*Corresponding author

better performance in comparison to not using atrous convolutions. The aggregation of context for small objects is damaged by a strong increase of atrous rates. This was observed even though increasing atrous rates steadily preserves the resolution and yields large receptive fields to aggregate the context; however, it deteriorates the performance for small vehicles leading to a failure to capture the essential context. This effect indicates that even though atrous convolutions are more and more common in state-of-the-art computer-vision methods, they should be used differently when it comes to segmenting small vehicles in airborne images.

The proposed method incorporates a new idea of joint vehicle segmentation and counting. To solve this issue, we perform segmentation with atrous convolutions, however, we do not increase atrous rates monotonously, but keep the rate steady and decrease it only at the end. This approach shows better performance compared with the naive monotonous atrous rate expansion. On the one hand, increasing atrous convolutions keep the resolution and context preserved, but on the other hand, the steady and decreasing atrous convolutions recover dense features for small vehicles. To further improve our proposed network for semantic segmentation of vehicles, we design a secondary branch in the network architecture, which is responsible for counting vehicles in the input image. During the training phase, we deploy a multi-task loss function including a conventional cross entropy loss for semantic segmentation and the Euclidean loss to decrease the error between the number of predicted vehicles and the actual number of vehicles in the input image. We train the network using these two loss functions in an end-to-end fashion. Results show better performance in the separation of nearby vehicles leading to an improvement in the semantic segmentation task. The output of the architecture is a binary image indicating vehicle positions with a different value as background as well as the number of vehicles yielding valuable information on the application side.

We evaluate our method on the "DLR 3K" dataset (Liu and Máttyus, 2015) which contains high-resolution aerial images with a *ground sampling distances (GSDs)* of 13 cm taken over the city of Munich, Germany. Results show that our proposed method performs well on this dataset and it outperforms the baseline network which does not include atrous convolutions.

2. RELATED WORK

The semantic segmentation methods have progressed considerably since Long et al. (2015) proposed *fully convolutional neural networks (FCNNs)*. The task of vehicle semantic segmentation is to assign each pixel a semantic label, which – when compared with vehicle detection – can provide a dense pixel-wise prediction of the vehicle location. In recent years, thanks to the advances in deep learning, FCNNs have achieved impressive results in semantic segmentation tasks. Hence, recently these methods have been investigated in the remote sensing domain with or without modifications. The majority of these works are based on the proposed architecture in computer vision: Long et al. (2015) and Badrinarayanan et al. (2015) use FCNN with skip-connections, Chen et al. (2018) is based on encoder-decoder, and Chen et al. (2014) relies on FCNN with *conditional random fields (CRFs)*. In order to segment and classify vehicles, Audebert et al. (2017) proposed a two-stage algorithm. First, vehicles are segmented via a binary FCNN, and then segmented vehicles are cropped and classified through a classification neural network. Kampffmeyer et al. (2017) also proposed a network based on FCNNs, but unlike previous methods, multi-class objects such as buildings, trees, low-vegetation and vehicles are classified directly. They also augment the segmentation of small objects by incorporating a balanced loss function. Liu et al. (2017) proposed a method on the basis of FCNN combined with CRFs to have pixel-wise semantic segmentation in high-resolution images in remote sensing. Sherrah (2016) also utilizes atrous convolutions, however, they employ max-pooling layers (stride 1) leading to decreased resolution of output feature maps. In the work of Yuan (2016), pixels are classified considering their distance to object instance boundary leading to a considerable improvement on boundary regions.

3. METHODOLOGY

In this section, we provide a detailed description of our proposed method in which we preserve output resolution and yet increase the size of the receptive field to aggregate contextual information in order to simultaneously segment and count vehicle instances in the input image. Figure 1 illustrates the overview of our method in which we design a neural network architecture composed of two branches for pixel-wise semantic segmentation and object counting. We use FCNNs as the first branch composed of mainly atrous convolution layers for the segmentation task. As the second branch, we use a combination of fully connected layers and atrous convolution layers for the counting task. In this section, we explain the architecture in detail.

FCNNs have shown high performance in the task of pixel-wise semantic segmentation (Chen et al., 2018). Although we can increase the receptive field by stacking striding and pooling layers on top of each other which yields better accuracy with contextual data, the feature maps are of lower spatial resolution *e.g.*, reduced by a factor of 32 in FCNNs. In the object recognition task, this might not be an issue. However, in the dense pixel-wise segmentation of small aerial objects in which spatial resolution has a vital role, it deteriorates the performance. Often, transposed convolution layers (Long et al., 2015) are utilized to recover the lost spatial resolution, but the use of atrous convolutions (Yu and Koltun, 2015) has shown better performance for this task.

The term "atrous convolution" (or "dilated convolution", rooted from the French expression "convolutions à trous") was developed originally for the wavelet transform in the "algorithme à trous". It has been used in the context of *deep convolutional neural networks (DCNNs) e.g.*, in Chen et al. (2018) and Yu and Koltun (2015) showing better performance compared with DCNNs transposed convolutions with pooling and striding layers. Atrous convolutions enlarge the kernel by integrating holes between pixels in kernels. The atrous "rate" is determined by the hyper-parameter r. Usually the stride between kernel parameters is $r-1$, *i.e.*, $r=1$ yields a regular convolution calculation. Atrous convolutions are used to cheaply increase the receptive field of output units without increasing the kernel size, which is especially effective when multiple atrous convolutions are stacked consecutively. A kernel of size k with an atrous convolution rate r has an effective size of

$$\hat{k} = k + (k-1)(r-1)\,.$$

As mentioned in Section 1, context plays an important role because small objects in high resolution images can be easily overlooked even by human experts. Moreover, even with contextual information, spotting small objects is not trivial, in particular,

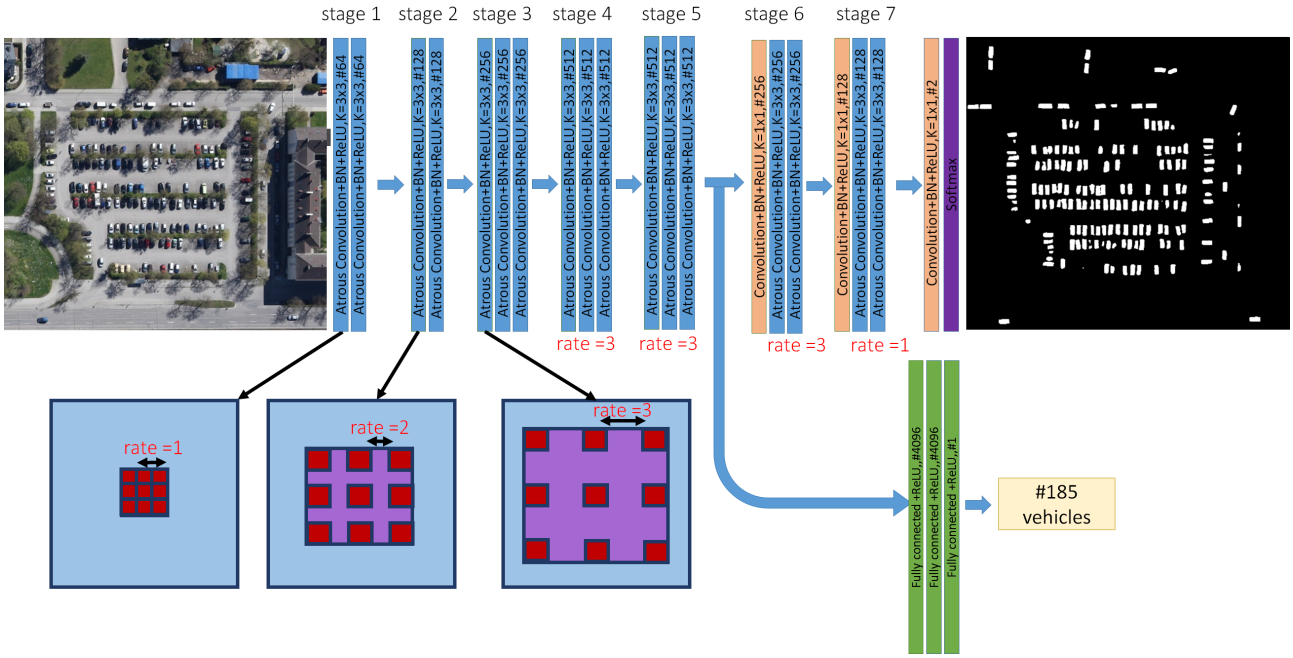

Figure 1. Overflow of the proposed method for pixel-wise dense semantic segmentation of vehicles.

if the resolution of the image is low. The components of the FCNN branch of our network in Figure 1 are designed to preserve the resolution by using atrous convolution layers. We use the VGG16 (Simonyan and Zisserman, 2014) architecture and adapt it to the new task of small vehicle segmentation in aerial imagery. Inspired by Yu and Koltun (2015) who had shown that pooling layers and striding convolutions damage the accuracy of dense pixel-wise prediction, we remove all pooling layers from the VGG16 architecture. In addition, we set the stride parameter to 1 in order to remove the side effects of striding convolution layers. We denote each block of the original VGG16 architecture in the new design as stages 1 to 5. We set the rate of atrous convolution to 1 for stage 1, 2 for stage 2 and 3 for stage 3. The reason for the increasing rate value is to capture a larger context in the extracted features. For stages 4 to 5 from the original VGG16, we keep the dilation rate unchanged to extract higher-level features. We preserve the number of channels compared to VGG16. Moreover, we perform batch-normalization in atrous convolutions for faster convergence during the training phase. The batch-normalization layer is also used to avoid over-fitting to some extent, as a replacement to dropout layers. As an extension to VGG16, we add two more stages: stages 6 and 7. Similar to the previous stages, in stage 6 and 7 we use atrous convolutions. However, as the first layer in each of these stages we apply a 1×1 convolutional layer to reduce the dimension. If we consider stages 1 to 5 as an expansion of the number of channels to extract a high amount of features, stages 6 and 7 can be seen as shrinking the number of channels to consider only informative features from those features. Another reason for shrinking the number of channels is to reduce computational cost.

After the 1×1 convolutional layer in stage 6 and 7, we use atrous convolution while keeping the number of channels unchanged. In stage 6, we still use atrous convolution layers with the rate of 3, however, in stage 7, we reduce the rate to 1. Although, increasing the atrous rate is beneficial to integrate more context by having a larger receptive field and also by preserving the resolution, the spatial information between neighboring features is decreased. This is detrimental for the dense pixel-wise segmentation of small objects, such as vehicles. Therefore, to solve this issue, we recover the inconsistency between features by reducing the atrous rate, which in our experiments proves to be useful to segment small objects. After stage 7, we apply a 1×1 convolutional layer to reduce the number of channels to 2, equal to the number of classes: vehicle and non-vehicle. The softmax function produces a probability map for each class in the end. To obtain the output map illustrated in Figure 1, for each pixel, we choose the class with the highest probability. As a post-processing step, we apply a dilation operation with the kernel size of 3×3.

Table 1. Ablation study on the proposed method based on atrous convolution layers. fc means fully connected layer. “last” is the last fc layer with one neuron for counting.

Stages	Counting	fc-layers	mIoU (%)	non-vehicle (%)	vehicle (%)
$1-5$	—	—	80.25	99.17	61.33
$1-6$	—	—	80.41	99.18	61.64
$1-7$	—	—	81.20	99.24	63.16
$1-7$	✓	last	81.25	99.25	63.25
$1-7$	✓	1-last	81.40	99.26	63.54
$1-7$	✓	1-2-last	81.58	99.26	63.89

In addition to pixel-wise information, the number of object instances also carries valuable insight, which can be used by the network to adapt its parameters for a better performance in the semantic segmentation task. Therefore, we apply two fully connected layers to the last layer of stage 5, as illustrated in Figure 1. This flattens the features, making it feasible to extract only one number as output. For this, we apply a fully connected layer with one neuron and then apply the ReLU activation function. We treat the output of the ReLU function as the predicted number of vehicles in the input image. Note that in this case, both tasks share weights from stage 1 to 5 as the main feature extraction step. We deploy a multi-task loss function penalizing errors both in dense pixel-wise prediction as well as in vehicle count prediction. The output of the dense pixel-wise prediction task is determined by the class posterior probability of the input image using the soft-

Table 2. Comparison of the proposed method with other approaches. Numbers represent percentages. In the first row, VGG19 was trained from scratch instead of using the pretrained model.

Method	backend network	mIoU	non-vehicle	vehicle	frequency weighted IoU	pixel acc.	mean acc.	counting acc.
FCN-8s	VGG19 (scratch)	74.15	98.78	49.53	98.00	98.80	86.91	83.76
FCN-8s	VGG19	79.96	99.15	60.77	98.54	99.16	90.86	89.61
FCN-8s	ResNet50	80.40	99.18	61.61	98.57	99.19	89.70	87.25
FCN-8s	ResNet101	80.67	99.20	62.15	98.59	99.21	89.78	86.46
DenseASPP	DenseNet121	78.29	99.03	57.55	98.36	99.04	**90.02**	86.15
PSPNet	ResNet50	78.71	99.09	58.32	98.43	99.11	88.36	77.79
ours	customized	**81.58**	**99.26**	**63.89**	**98.64**	**99.31**	90.01	91.12

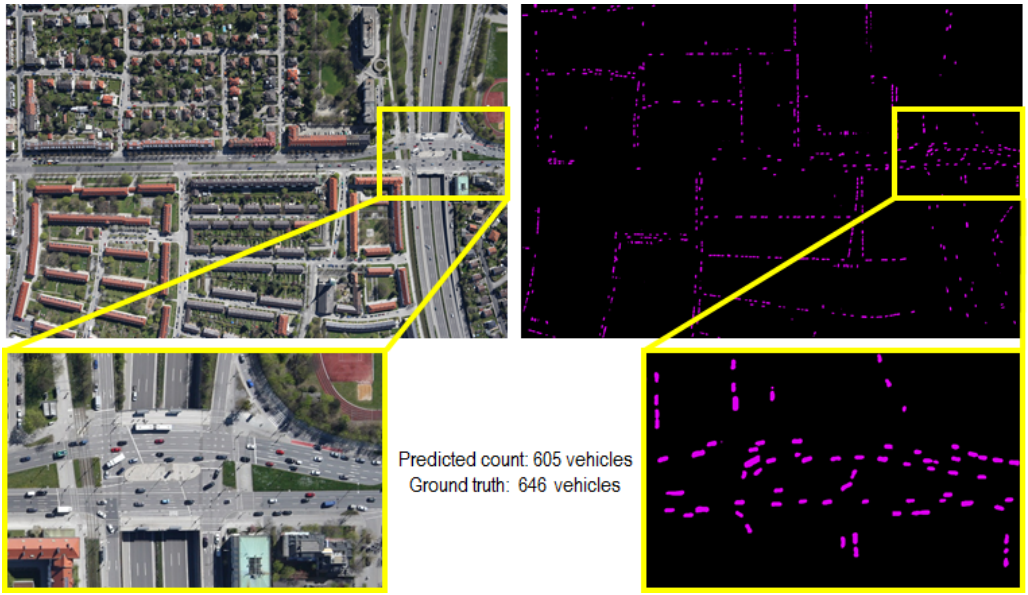

Figure 2. Sample performance of our proposed method on the DLR 3K dataset. The left image is the original input image. Segmented vehicles are shown in violet in the right image. We merged truck and car classes for single-class vehicle segmentation. Our algorithm is capable of segmenting vehicles with high accuracy in the areas with sparse vehicle distribution.

max function. The softmax function creates a vector of real values between 0 and 1 to indicate the distribution of the probability of a class j and an input vector X as

$$p(y=j|X;W) = \frac{e^{X^T W_j}}{\sum_{k=1}^{K} e^{X^T W_k}}\,, \tag{1}$$

where W stands for the network weights, y is the predicted class, and K is the number of classes. For dense prediction of the feature map output shown in Figure 1, we use the cross-entropy loss function expressed as

$$L_{\text{segmentation}} = -\sum_{x} p'(x)\log p(x)\,, \tag{2}$$

where x is the input pixel and $p(x)$ the softmax function. We aim to reduce the cross-entropy between the “true” one-hot encoded data distribution denoted as $p'(x)$ and the predicted probabilities $p(x)$. To minimize counting errors, we use the Euclidean distance as loss function defined as

$$L_{\text{count}} = \|H(X,W) - C\|^2\,, \tag{3}$$

where H is the network function mapping the input data X using the network weights W to the predicted number of vehicles and C stands for the ground truth vehicle count. H is the output of the ReLU layer: the predicted vehicle number. We count an object as a vehicle when 70% of its pixels are inside the annotated object.

4. EXPERIMENTS AND DISCUSSION

We carried out the experiments using the “DLR 3K” dataset, captured over the city of Munich in Germany. This dataset contains 20 images with the GSD of 13 cm and resolution of 5616×3744 px split in a training and a test set, each set with 10 images. During the training phase, we use *Stochastic Gradient Descent (SGD)* as optimizer with a learning rate of 0.0001, a momentum of 0.9 and a weight decay of 0.004 for 100 epochs. We utilize Tensorflow (Abadi et al., 2016) as the implementation framework. As the images are large, to avoid memory issues, we crop each image to a patch of size 1024×1024 with a patch overlap of 100 px. For the final output, we stitch patches together to form an output map with the same resolution as the input image. Similar to Long et al. (2015), we use *mean intersection over union (mIoU)* as the main criterion to evaluate our proposed algorithm as well as pixel accuracy, mean accuracy, and frequency weighted IoU. Table 1 shows the ablation experiments on the proposed network for the semantic segmentation task. First, we trained the network without stages 6 and 7 and without the counting branch. The results show that even though the network is not very deep, it can achieve a decent mIoU of 80.25%, outperforming FCN-8s (Long et al., 2015) with VGG19 feature-extraction (backend) network by a small margin of 0.29%

mIoU. This shows the effectiveness of employing atrous convolution layers in contrast to the approach of utilizing pooling and striding layers. Although adding stage 6 improves the performance by 0.16% mIoU, this boost is not significant and could be due to the increased depth. However, adding stage 7 with atrout rate 1 shows a significant improvement of 1.80% mIoU which indicates that using lower atrous rates to recover the connectivity between feature map elements in the last part of the network can lead to a better performance in dense pixel-wise prediction of small objects. Moreover, adding the counting task to the network improves performance by 0.25% mIoU which is larger than the effect of stage 6. This indicates that the additional task of vehicle counting plays a more important role than solely increasing the depth of the network.

In Table 2, we compared our proposed network with other network architectures. The quantitative comparisons show that our proposed method outperforms all baselines. Interestingly, the recently proposed PSPNet (Zhao et al., 2017) which uses a ResNet50 backend performs worse than FCN-8s (Long et al., 2015) with the same backend. This shows that, in order to segment small objects in aerial images, FCNNs should be modified to be adapted to this new domain. Figure 2 shows sample vehicle segmentation and counting performance of the proposed method in which vehicles are indicated with violet. The high performance suggests that atrous convolutional networks should be further studied in this domain. Albeit recent FCNNs, *e.g.*, PSPNet or DenseASPP (Yang et al., 2018) outperform older methods, *e.g.*, FCN-8s, their direct application for small object segmentation leads to worse performance.

5. CONCLUSION

In this work, we presented a joint vehicle segmentation and counting method based on FCNNs with atrous convolutions. We showed that atrous convolutions are an effective operation compared to pooling and striding ones, but for segmenting vehicles as small objects a combination of increasing and decreasing atrous rates should be used, instead of monotonously increasing the atrous rate. Moreover, we showed that integrating the vehicle counting task, in addition to pixel-wise segmentation, leads to a better performance. In the future, more investigations of different combinations of atrous rates could result in an even better performance.

REFERENCES

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M. et al., 2016. Tensorflow: a system for large-scale machine learning. In: *OSDI*, Vol. 16, pp. 265–283.

Audebert, N., Le Saux, B. and Lefèvre, S., 2017. Segment-before-detect: Vehicle detection and classification through semantic segmentation of aerial images. In: *Remote Sensing*, Vol. 9number 4, Multidisciplinary Digital Publishing Institute, p. 368.

Badrinarayanan, V., Kendall, A. and Cipolla, R., 2015. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *arXiv preprint arXiv:1511.00561*.

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K. and Yuille, A. L., 2014. Semantic image segmentation with deep convolutional nets and fully connected CRFs. *arXiv preprint arXiv:1412.7062*.

Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F. and Adam, H., 2018. Encoder-decoder with atrous separable convolution for semantic image segmentation. *arXiv preprint arXiv:1802.02611*.

Hariharan, B., Arbeláez, P., Girshick, R. and Malik, J., 2015. Hypercolumns for object segmentation and fine-grained localization. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 447–456.

Hu, P. and Ramanan, D., 2017. Finding tiny faces. In: *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, IEEE, pp. 1522–1530.

Kampffmeyer, M., Jenssen, R. et al., 2017. Urban land cover classification with missing data using deep convolutional neural networks. In: *Geoscience and Remote Sensing Symposium (IGARSS), 2017 IEEE International*, IEEE, pp. 5161–5164.

Liu, K. and Máttyus, G., 2015. Fast multiclass vehicle detection on aerial images. In: *IEEE Geosci. Remote Sensing Lett.*, Vol. 12number 9, pp. 1938–1942.

Liu, Y., Piramanayagam, S., Monteiro, S. T. and Saber, E., 2017. Dense semantic labeling of very-high-resolution aerial imagery and lidar with fully-convolutional neural networks and higher-order CRFs. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 1561–1570.

Long, J., Shelhamer, E. and Darrell, T., 2015. Fully convolutional networks for semantic segmentation. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440.

Máttyus, G., Luo, W. and Urtasun, R., 2017. Deeproadmapper: Extracting road topology from aerial images. In: *The IEEE International Conference on Computer Vision (ICCV)*, pp. 3458–3466.

Redmon, J. and Farhadi, A., 2017. Yolo9000: better, faster, stronger. *arXiv preprint arXiv:1612.08242*.

Ren, S., He, K., Girshick, R. and Sun, J., 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In: *Advances in neural information processing systems*, pp. 91–99.

Sherrah, J., 2016. Fully convolutional networks for dense semantic labelling of high-resolution aerial imagery. *arXiv preprint arXiv:1606.02585*.

Simonyan, K. and Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Tang, T., Zhou, S., Deng, Z., Zou, H. and Lei, L., 2017. Vehicle detection in aerial images based on region convolutional neural networks and hard negative example mining. In: *Sensors*, Vol. 17number 2, Multidisciplinary Digital Publishing Institute, p. 336.

Yang, M., Yu, K., Zhang, C., Li, Z. and Yang, K., 2018. DenseASPP for semantic segmentation in street scenes. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3684–3692.

Yu, F. and Koltun, V., 2015. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*.

Yuan, J., 2016. Automatic building extraction in aerial scenes using convolutional networks. *arXiv preprint arXiv:1602.06564*.

Zhao, H., Shi, J., Qi, X., Wang, X. and Jia, J., 2017. Pyramid scene parsing network. In: *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 2881–2890.