OVeNet: Offset Vector Network for Semantic Segmentation

Stamatis Alexandropoulos
Princeton University *

Christos Sakaridis
ETH Zürich

Petros Maragos
National Technical University of Athens

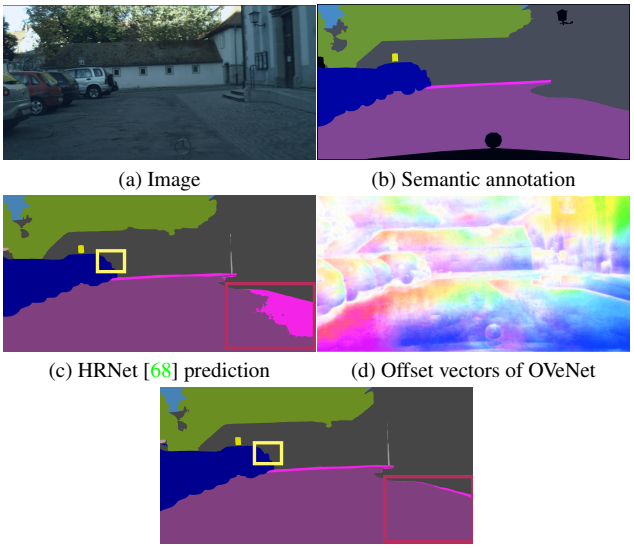
Abstract

Semantic segmentation is a fundamental task in visual scene understanding. We focus on the supervised setting, where ground-truth semantic annotations are available. Based on knowledge about the high regularity of real-world scenes, we propose a method for improving class predictions by learning to selectively exploit information from neighboring pixels. In particular, our method is based on the prior that for each pixel, there is a seed pixel in its close neighborhood sharing the same prediction with the former. Motivated by this prior, we design a novel two-head network, named Offset Vector Network (OVeNet), which generates both standard semantic predictions and a dense 2D offset vector field indicating the offset from each pixel to the respective seed pixel, which is used to compute an alternative, seed-based semantic prediction. The two predictions are adaptively fused at each pixel using a learnt dense confidence map for the predicted offset vector field. We supervise offset vectors indirectly via optimizing the seed-based prediction and via a novel loss on the confidence map. Compared to the baseline state-of-the-art architectures HRNet and HRNet+OCR on which OVeNet is built, the latter achieves significant performance gains on three prominent benchmarks for semantic segmentation, namely Cityscapes, ACDC and ADE20K. Code is available at https://github.com/stamatisalex/OVeNet.

1. Introduction

Semantic segmentation is one of the most central tasks in computer vision. In particular, it is the task of assigning a class to every pixel in a given image. It has lots of applications in a variety of fields, such as autonomous driving [9, 30], robotics [3, 65], and medical image processing [1, 57], where pixel-level labeling is critical.

The adoption of convolutional neural networks (CNNs) [47] for semantic image segmentation has led to a tremendous improvement in performance on challenging, large-scale datasets, such as Cityscapes [21], MS COCO [42] and ACDC [58]. Most of the related works [14, 17, 51, 68, 76] focus primarily on architectural modifications of the employed networks in order to better combine global context aggregation and local detail preservation, and use a simple loss that is computed on individual pixels. The design of more sophisticated losses [32, 49, 77] that take into account the structure which is present in semantic labelings has received significantly less attention. Many supervised techniques utilize a pixel-level loss function that handles predictions for individual pixels independently of each other. By doing so, they ignore the high regularity of real-world scenes, which can eventually profit the final model's performance by leveraging information from adjacent pixels.



(a) Image (b) Semantic annotation (c) HRNet [68] prediction (d) Offset vectors of OVeNet (e) OVeNet prediction (ours)

Figure 1. The semantic content of real-world scenes has a high degree of regularity. We propose a semantic segmentation method which can exploit this regularity, by learning to selectively leverage information from neighboring *seed* pixels. Our proposed Offset Vector Network (OVeNet) can improve upon state-of-the-art architectures [68] by estimating the offset vectors to such seed pixels and using them to refine semantic predictions.

*This work was done when Stamatis Alexandropoulos was in the National Technical University of Athens.

Thus, these methods misclassify several pixels, primarily near semantic boundaries, which leads to major losses in performance.

Based on knowledge about the high regularity of real scenes, we propose a method for improving class predictions by learning to selectively exploit information from neighboring pixels. In particular, the general nature of this idea is applicable on SOTA models like HRNet [68] or HRNet + OCR [77] and can extend these models by adding a second head to them capable of achieving this goal.

The architecture of our Offset Vector Network (OVeNet) is shown in Fig. 2. In particular, following the base architecture of the backbone model (e.g HRNet or HRNet + OCR), the first head of our network outputs the initial pixel-level semantic predictions. In general, two pixels $\mathbf{p}$ and $\mathbf{q}$ that belong to the same category share the same semantic outcome. If the pixels belong to the same class, using the label of $\mathbf{q}$ for estimating class at the position of $\mathbf{p}$ results in a correct prediction.

We leverage this property by learning to identify *seed pixels* which belong to the same class as the examined pixel, whenever such pixels exist, in order to selectively use the prediction at the former for improving the prediction at the latter. This idea is motivated by a prior which states for every pixel $\mathbf{p}$ associated with a 2D semantically segmented image, there exists a seed pixel $\mathbf{q}$ in the neighborhood of $\mathbf{p}$ which shares the same prediction with the former. In order to predict classes with this scheme, we need to find the regions where the prior is valid. Furthermore, in order to point out the seed pixels in these regions, we must predict the offset vector $\mathbf{o}(\mathbf{p}) = \mathbf{q} - \mathbf{p}$ for each pixel $\mathbf{p}$.

As a result, we design a second head that generates a dense offset vector field and a confidence map. The predicted offsets are used to resample the class predictions from the first head and generate a second class prediction. The outcomes from the two heads are fused adaptively using the learnt confidence map as fusion weights, in order to down-weigh the offset-based prediction and rely primarily on the basic class prediction in regions where the prior is not valid. Thanks to using seed pixels for prediction, our network classifies several pixels with incorrect initial predictions, e.g., boundary pixels, to the correct classes. Thus, it improves the shape as well as the form of the corresponding segments, leading to more realistic results. Last but not least, we propose a confidence loss which supervises the confidence map explicitly and further improves performance. An illustrative example of this concept is depicted in Fig. 1, where OVeNet outperforms the baseline HRNet [68] model, since it enlarges correctly the road and the car segment (red and yellow frame correspondingly) and reduces the total number of misclassified pixels.

We evaluate our method extensively on 3 primary datasets, each serving a specific purpose. For semantic segmentation in driving scenes, we focus on two benchmarks: Cityscapes [21] and ACDC [58]. Additionally, we broaden our evaluation by incorporating the ADE20K [86, 87] dataset, which covers a diverse range of images spanning various indoor and outdoor scenes. We implement our offset vector branch both on HRNet [68] and HRNet+OCR [78]. Our approach significantly improves the initial models' output predictions by achieving better mean and per-class results. We conduct a thorough qualitative and quantitative experimental comparison to show the clear advantages of our method over previous SOTA techniques.

2. Related Work

Semantic segmentation architectures. Fully convolutional networks [59, 60] were the first models that re-architected and fine-tuned classification networks to direct dense prediction of semantic segmentation. They generated low-resolution representations by eliminating the fully-connected layers from a classification network (e.g AlexNet [34], VGGNet [62] or GoogleNet [63]) and then estimating coarse segmentation maps from those representations. To create medium-resolution representations [13–15, 36, 76], fully convolutional networks were expanded using dilated/atrous convolutions, which replaced a few strided convolutions and their associated ones. Following, in order to restore high-resolution representations from low-resolution representations an upsample process was used. This process involved a subnetwork that was symmetric to the downsample process (e.g VGGNet [62]), and included skipping connections between mirrored layers to transform the pooling indices (e.g. DeconvNet [51]). Other methods include duplicating feature maps, which is used in architectures like U-Net [57] and Hourglass [8, 20, 22, 31, 50, 64, 73, 75] or encoder-decoder architectures [2, 55]. Lastly, the process of asymmetric upsampling [7, 18, 29, 41, 54, 66, 71, 81] has also been extensively researched.

The models' representations were then enhanced to include multi-scale contextual information [10, 11, 82]. PSPNet [83] utilized regular convolutions on pyramid pooling representations to capture context at multiple scales, while the DeepLab series [14, 15] used parallel dilated convolutions with different dilation rates to capture context from different scales. Recent research [26, 35, 74] proposed extensions, such as DenseASPP [74], which increased the density of dilated rates to cover larger scale ranges, or HS3 [5], which supervised intermediate layers in a segmentation network to learn meaningful representations by varying task complexity. Other studies [17, 25, 38] used encoder-decoder structures to exploit the multi-resolution features as the multi-scale context. Here belongs the HRNet [68], the baseline model of our method. HRNet connects high-to-low convolution streams in parallel. It ensures that

high-resolution representations are maintained throughout the entire process and creates dependable high-resolution representations with accurate positional information by repeatedly merging the representations from various resolution streams. Applying additionally the OCR [77] method, HRNet + OCR is one of the leading models in the task of semantic segmentation.

Lately, transformers have been successful in computer vision tasks demonstrating their effectiveness. ViT [23] was the first attempt to use the vanilla transformer architecture [67] for image classification without extensive modification. Unlike later methods, such as PVT [69] and Swin [46], that incorporated vision-specific inductive biases into their architectures, the plain ViT suffers inferior performance on dense predictions due to weak prior assumptions. To tackle this problem, the ViT-Adapter [19] was introduced, which allowed plain ViT to achieve comparable performance to vision-specific transformers and achieves the SOTA performance on this task.

Semantic segmentation loss functions. Image segmentation has highly correlated outputs among the pixels. Converting pixel labeling problem into an independent problem can lead to problems such as producing results that are spatially inconsistent and have unwanted artifacts, making pixel-level classification unnecessarily challenging. To solve this problem, several techniques [16, 33, 45, 85] have been developed, such as integrating structural information into segmentation. For instance, Chen et al. [14] utilized denseCRF [33] for refining the final segmentation result. Following, Zheng et al. [85] and Liu et al. [45] made the CRF module differentiable within the deep neural network. Other methods that have been used to encode structures include pairwise low-level image cues like grouping affinity (e.g. SPNs [44], Affinity CNNs [48]) and contour cues [4, 12]. InverseForm [6] is another boundary-aware loss term using an inverse-transformation network, which efficiently learns the degree of parametric transformations between estimated and target boundaries. GANs [56] are an alternative for imposing structural regularity in the neural network output. However, these methods may not work well in cases where there are changes in visual appearance or may require expensive iterative inference procedures. Thus, Ke et. al. [32] introduced AAFs, which are easier to train than GANs and more efficient than CRF without run-time inference. There has been also proposed another loss function [49] suitable for in real time applications that pulls the spatial embeddings of pixels belonging to the same instance together.

Offset Vector-Based methods are essential for image analysis tasks that involve adjacent pixels. In particular, they can effectively exploit the information contained in neighbouring pixels, handle image distortions and noise, and improve the accuracy of various image analysis tasks. They can be used in applications such as depth estimation [52,53] or semantic segmentation [49,79]. Non-local SPNs [52] enhance depth completion by iteratively refining initial depth predictions using non-local neighbors. Based on knowledge about the high regularity of real 3D scenes, P3Depth [53] is another method used for 3D depth estimation that learns to selectively leverage information from coplanar pixels to improve the predicted depth. In semantic segmentation, SegFix [79] is a model-agnostic post-processing scheme that improved the boundary quality for the segmentation result. Motivated by the empirical observation that the label predictions of interior pixels are more reliable, SegFix replaced the originally unreliable predictions of boundary pixels by the predictions of interior pixels. OVeNet provides another perspective to use offset vectors and structure modeling by matching the relations between neighbouring pixels in the label space. Although our approach is inspired by the P3Depth idea, we focus on a different task in the 2D world which is semantic segmentation. Moreover, the key difference between our method and SegFix is the timing of when they are applied. Essentially, OVeNet integrates the offset vector learning process into the model training, while SegFix applies the offset correction as a separate post-processing step.

3. Method

In this section, we will analyze our method shown in Fig. 2. Firstly, in Sec. 3.1 we give some basic notation and terminology of semantic segmentation. As we mentioned before in Sec. 1, our network estimates semantic labels by selectively combining information from each pixel and its corresponding seed pixel. The intuition and the advantages of using seed pixels to improve the initial prediction of a model are described analytically in Sec. 3.2. Lastly, in Sec. 3.3 we introduce an additional confidence loss, which further enhances our method.

3.1. Terminology

Semantic segmentation requires learning a dense mapping $f_\theta : I(u, v) \rightarrow S(u, v)$ where I is the input image with spatial dimensions $H \times W$, S is the corresponding output prediction map of the same resolution, (u, v) are pixel coordinates in the image space and θ are the parameters of the mapping f. In supervised semantic segmentation, a ground-truth semantically segmented map H is available for each image during training. The aim is to optimize the function parameters θ such that the predicted output map is as close as possible to the ground-truth map across the entire training set T. This can be achieved by minimizing the difference between the predicted and ground-truth images:

$$\min_{\theta} \sum_{(I,H)\epsilon T} \mathcal{L}(f_\theta(I), H) \tag{1}$$

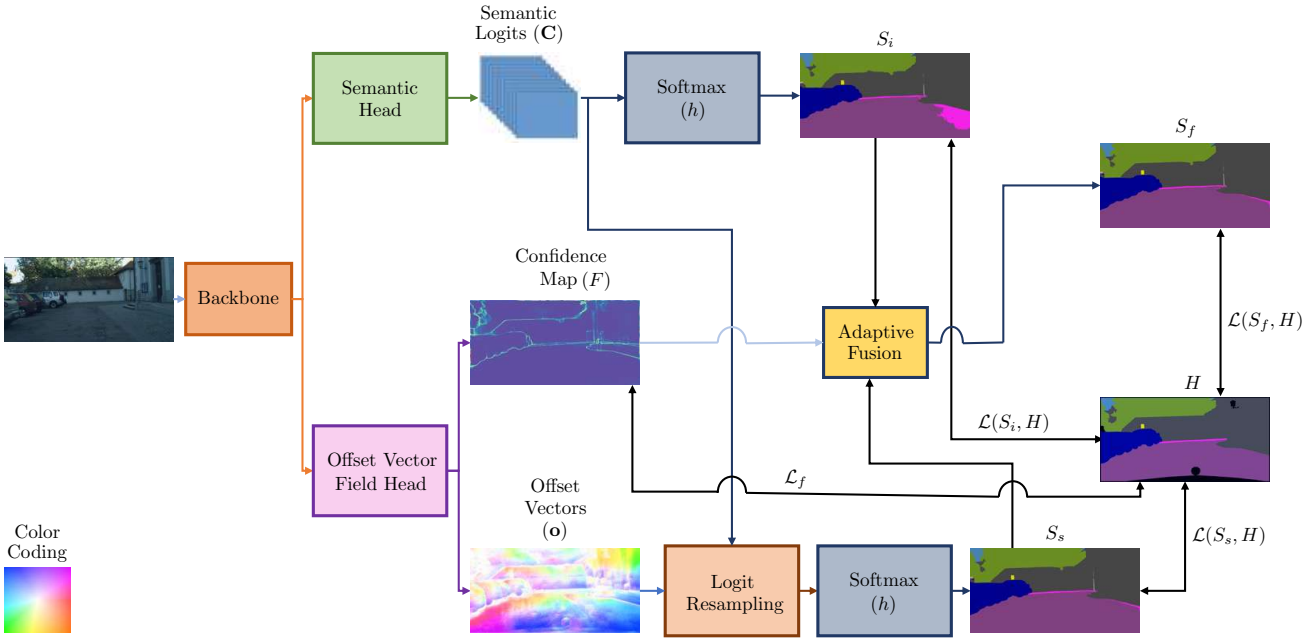


Figure 2. **Overview of OVeNet.** OVeNet is a two-headed network. The first head outputs semantic logits ($\mathbf{C}$), while the second head outputs a dense offset vector field ($\mathbf{o}$) identifying positions of seed pixels along with a confidence map (F). The logits are passed through a softmax function and output the initial class prediction (S_i) of the model. Then, the offset vectors are used to resample the logits from the first head and generate a second class prediction (S_s). The two predictions are adaptively fused using the confidence map resulting in the final prediction S_f. For the visualization of the offset vectors we use the optical flow color coding from [28]. Smaller vectors are lighter and color represents the direction.

where $\mathcal{L}$ is a loss function that penalizes variations between the prediction and the ground truth.

3.2. Seed Pixel Identification

Let us assume we have one pixel $\mathbf{p}$ which belongs to segment of a semantically segmented image. By definition, every other pixel on this segment has the same class value. Thus, ideally, in order to get all of the class values accurate, the network only has to predict the class at one of these pixels, $\mathbf{q}$. This pixel can be interpreted as the seed pixel that describes the segment-class. Finally, we let the network find this seed pixel and the corresponding region.

Let us define a prior which is a relaxed version of the previous idea.

Definition 1 *For every pixel* $\mathbf{p}$ *associated with a 2D semantically segmented image, there exists a seed pixel* $\mathbf{q}$ *in the neighborhood of* $\mathbf{p}$ *which shares the same prediction with the former.*

In general, there may be numerous seed pixels for $\mathbf{p}$ or none at all. Given that the Definition 1 holds, semantic segmentation task for $\mathbf{p}$ can be solved by identifying $\mathbf{q}$. For this reason, we let our network predict the offset vector $\mathbf{o}(\mathbf{p}) = \mathbf{q} - \mathbf{p}$. Thus, we design our model so that it features a second, offset head and let this offset head predict a dense offset vector field $\mathbf{o}(u, v)$. The two heads of the network share a common main body and then they follow different paths. We resample the initial logits $\mathbf{C}$, being predicted by the first head, using the estimated offset vector field via:

$$\mathbf{C_s}(\mathbf{p}) = \mathbf{C}(\mathbf{p} + \mathbf{o}(\mathbf{p})) \tag{2}$$

To manage fractional offsets, bilinear interpolation is used. The resampled logits are then used to compute a second semantic segmentation prediction:

$$\begin{aligned} S_s(u, v) &= h(\mathbf{C_s}(u, v), u, v) \\ \implies S_s(\mathbf{p}) &= S_i(\mathbf{p} + \mathbf{o}(\mathbf{p})) \end{aligned} \tag{3}$$

based on the seed locations. In our experiment, $h = softmax$.

Due to the fact that the prior is not always correct, the initial semantic prediction S_i may be preferred to the seed-based prediction S_s. To account for such cases, the second head additionally predicts a confidence map $F(u, v) \in [0, 1]$, which represents the model's confidence in adopting the predicted seed pixels for semantic segmentation via S_s. By adaptively fusing S_i and S_s, the confidence map is used to compute the final prediction:

$$S_f(\mathbf{p}) = (1 - F(\mathbf{p}))S_i(\mathbf{p}) + F(\mathbf{p})S_s(\mathbf{p}) \tag{4}$$

We apply supervision to each of S_f, S_s, and S_i in our model, by optimizing the following loss:

$$\mathcal{L}_{\text{semantic}} = \mathcal{L}(S_f, H) + \kappa\mathcal{L}(S_s, H) + \lambda\mathcal{L}(S_i, H) \quad (5)$$

with κ and λ being hyperparameters and H denoting the GT train ids of each class for each pixel. In this way, we encourage the initial model's head to output an accurate representation across all pixels, even when they have a high confidence value, and the offset vector head to learn high confidence values for pixels for which Definition 1 holds and low confidence values for pixels for which the prior does not.

Nonetheless, there is a downside to this approach. Since the model is not supervised directly on the offsets, it has the potential to predict zero offsets across the board. This implies S_s and S_f predictions equivalent to S_i. Since the initial predictions S_i are erroneously smoothed around semantic boundaries due to the regularity of the mapping f_θ in the case of neural networks, this undesirable behavior is avoided in practice. We opt for predicting non-zero offsets that point away from the boundary. Such a non-zero offset utilizes a seed pixel for S_s located further from the border and diminishes inaccuracies stemming from smoothing. Furthermore, these non-zero offsets extend from the boundaries into the inner sections of regions with smooth segments, aiding the network in forecasting non-trivial offsets, thanks to the regularity of the mapping that forms the offset vector field. Thus, pixels on either side of the boundary have a lower $\mathcal{L}_{\text{semantic}}$ value.

3.3. Confidence-Based Loss

Our confidence loss is based on the concept that given a pixel coordinate, its surrounding pixels should be in the same segment. For each pixel $\mathbf{p}$, we define the confidence loss as follows:

$$\begin{aligned}\mathcal{L}_f(\mathbf{p}) = &-\mathbb{1}_{H(\mathbf{p})=H(\mathbf{p}+\mathbf{o}(\mathbf{p}))} \log F(\mathbf{p}) \\ &- \mathbb{1}_{H(\mathbf{p})\neq H(\mathbf{p}+\mathbf{o}(\mathbf{p}))} \log(1 - F(\mathbf{p}))\end{aligned} \quad (6)$$

This idea is motivated by the fact that the confidence value should be large for pixels whose offset vector points to seed pixels with the same class. Similarly, the confidence value should be small for pixels whose offset vector points to seed pixels with a different class.

To sum up, the complete loss is:

$$\mathcal{L}_{\text{final}} = \mathcal{L}_{\text{semantic}} + \mathcal{L}_f \quad (7)$$

4. Experiments

To evaluate the proposed method, we carry out comprehensive experiments on the Cityscapes, ACDC and ADE20K datasets. Experimental results demonstrate that our method, compared to the baseline state-of-the-art architectures HRNet [68] and HRNet + OCR [78] on which it is built, achieved higher performance, outperforming these baselines. In the following, we first introduce the datasets, evaluation metrics and implementation details in Sec. 4.1. We then compare our method to SOTA approaches in Sec. 4.2. Finally, we perform a series of ablation experiments on Cityscapes in Sec. 4.3.

4.1. Experimental Setup

In this section, we present the Cityscapes, ACDC and ADE20K datasets, which are used to evaluate our approach. Evaluation on these datasets is performed using standard semantic segmentation metrics explained below.

Cityscapes [21] is a challenging urban scene understanding dataset. There are 30 classes from which only 19 classes are used for parsing evaluation. Around 5000 high quality pixel-level finely annotated images and 20000 coarsely annotated images make up the collection. The finely annotated 5000 images are divided into 2975, 500, 1525 images for training, validation and testing respectively.

ACDC [58] is a demanding dataset, used for training and testing semantic segmentation methods on adverse visual conditions. There are 4006 images divided evenly between four frequent unfavorable conditions: fog, dark, rain, and snow and 19 semantic classes, coinciding exactly with the evaluation classes of the Cityscapes dataset. It includes 1600 training and 406 validation images with public annotations and 2000 test images with annotations withheld for benchmarking purposes.

ADE20K [86] is a challenging scene parsing dataset which covers a diverse range of images depicting various indoor and outdoor scenes. It consists of 20210 images as the training set and 2000 images as the validation set. There are totally 150 semantic classes, including categories like sky, road, grass and discrete objects like person, car, bed.

Evaluation Metrics. The mean of class-wise intersection over union (mIoU) is adopted as the evaluation metric. In addition to the mean of class-wise intersection over union (mIoU), we report other three scores on the test set: IoU category (cat.), iIoU class (cla.) and iIoU category (cat.)

Implementation Details. Our network consists of two heads. The first head outputs 19 channels, one for each class. The second head outputs three channels: one for each coordinate of the offset vectors and one for confidence. These two heads follow the structure of HRNet. Both OVeNet and the baseline HRNet are initialized with pre-trained ImageNet [34] weights. This initialization is important to achieve competitive results as in [68]. Following the same training protocol as in [68], the data are augmented by random cropping (from 1024 × 2048 to 512 × 1024 in Cityscapes and from 1080 x 1920 to 540 x 960 in ACDC), random scaling in the range of [0.5, 2], and ran-

dom horizontal flipping.We use the SGD optimizer with a base learning rate of 0.01, a momentum of 0.9 and a weight decay of 0.0005. The number of epochs used for training is 484. For lowering the learning rate, a poly learning rate policy with a power of 0.9 is applied. The offset vectors are restricted via a tanh layer to have a maximum length of τ in normalized image coordinates. After an ablation study shown in Sec. 4.3, we set τ, λ and μ to 0.5 by default and branching is applied at the last (4^{th}) stage of HRNet. The confidence map is predicted through a sigmoid layer. For S_i, S_s and S_f predictions, Ohem Cross Entropy Loss [61] is used. We first experimented with predicting S_i, S_s and S_f directly, but this led to inferior results. On the other hand, our final model for which we report performance in the paper outputs the S_i, S_s, S_f logits. In addition, the confidence based loss is applied using the final semantic prediction S_f.

4.2. Comparison with the State of the Art

Cityscapes. The results on Cityscapes are shown below. We achieve better results on Cityscapes than both the initial HRNet and HRNet + OCR model under similar training time, outperforming prior SOTA methods based on HRNet backbones. Table 1 compares our method with SOTA methods on the Cityscapes test set. All the results are with six scales and flipping. Two cases w/o using coarse data are evaluated: one is about the model learned on the train set, and the other on the train + val set. Our offset vector model excels in both cases with performance gains of 1.4% in mIoU, 2.4% in iIoU cat. 0.4% in IoU cat. and 1.4% in iIoU cat. over the HRNet model learned only on train set and a gain of 0.5% in mIoU over the HRNet + OCR model learned on both train + val set. The OVeNet model which is built on HRNet and trained on the training set outperforms the HRNet baseline model which is trained on the training + val set. Table 2 thoroughly compares our approach with HRNet's per-class results. Our method achieves better results in the majority of classes. Our offset vector-based model learns an implicit representation of different objects which can benefit the overall semantic segmentation estimation capability of the network. Regarding val set results, HRNet achieves 81.8% mIoU while OVeNet built on it surpasses it reaching 82.4% mIoU .

Qualitative results on Cityscapes support the above findings, as shown in Fig. 3. To be more specific, from left to right, we depict the RBG input image, GT image, CCNet's [27], DANet's [24], HRNet's [78] and our model's prediction. Specifically, our model demonstrates successful classification of incorrectly predicted pixels (identified by a red and a blue frame) in the first example. In the second example, OVeNet exhibits superior performance compared to previous models, as it accurately expands the sidewalk and the pole depicted in the blue and yellow frames, respectively. Furthermore, not only does it successfully eliminate discontinuities in the pink frame, but it also predicts a better representation of the bicycle's shape. As far as the last two examples are concerned, our model shrinks the false predictions of HRNet on sidewalk, pole and bus segments resulting in a superior prediction compared to the HRNet baseline. To sum up, OVeNet surpasses the performance of HRNet.

Table 1. **Semantic segmentation results on Cityscapes test set.**. We compare our method against SOTA methods as in [68].D-ResNet-101 = Dilated-ResNet-101. By default, OVeNet is built on HRNet, unless stated otherwise.

	backbone	mIoU	iIoU cla.	IoU cat.	iIoU cat.
Model learned on the `train` set					
PSPNet [82]	D-ResNet-101	78.4	56.7	90.6	78.6
PSANet [84]	D-ResNet-101	78.6	-	-	-
HRNet [68]	HRNetV2-W48	80.4	59.2	91.5	80.8
OVeNet	HRNetV2-W48	**81.8**	**61.6**	**91.9**	**82.2**
Model learned on the `train+val` set					
DeepLab [10]	D-ResNet-101	70.4	42.6	86.4	67.7
RefineNet [40]	ResNet-101	73.6	47.2	87.9	70.6
DSSPN [37]	D-ResNet-101	76.6	56.2	89.6	77.8
ResNet38 [70]	WResNet-38	78.4	59.1	90.9	78.1
PADNet [72]	D-ResNet-101	80.3	58.8	90.8	78.5
CFNet [80]	D-ResNet-101	79.6	-	-	-
Auto-DeepLab [43]	-	80.4	-	-	-
DenseASPP [82]	WDenseNet-161	80.6	59.1	90.9	78.1
CCNet [27]	D-ResNet-101	81.4	-	-	-
DANet [24]	D-ResNet-101	81.5	-	-	-
HRNet [68]	HRNetV2-W48	81.6	61.8	92.1	82.2
HRNet + OCR [78]	HRNetV2-W48	81.9	**62.0**	**92.0**	**82.5**
OVeNet (*HRNet + OCR*)	HRNetV2-W48	**82.4**	61.6	91.9	82.2

ACDC. We also outperform initial HRNet and prior SOTA methods under similar training time. Table 4 compares our approach with SOTA models not only on all methods but also on different conditions of the ACDC test set. OVeNet improves the initial HRNet model by 2.5% mIoU in "All" conditions. Additionally, we can observe from the per class results shown on Table 3 on different conditions, that our approach outperforms HRNet in the vast majority of classes as well as in the total mIoU score. Specifically, in foggy images, small-instance classes like person, rider and bicycle perform poorly because of contrast reduction and resolution issues due to distant instances. There is also a huge performance boost of approximately 10% and 15% on the "bus" and "truck" class respectively. Moreover, it is more difficult to separate classes at night that are often dark or poorly lit, such as buildings, vegetation, traffic signs, and the sky. This behaviour is observed also in offset vector performance as they have small values when the visibility is limited. Lastly, during night and snow conditions, road and sidewalk performance is at its lowest, which can be attributed to misunderstanding between the two classes as a result of their similar look. As for val set results, HRNet achieves 75.5% mIoU while our OVeNet surpasses it reaching 75.9% mIoU.

Qualitative results on ACDC support the above findings, as shown in Fig. 4. To be more specific, in the first column,

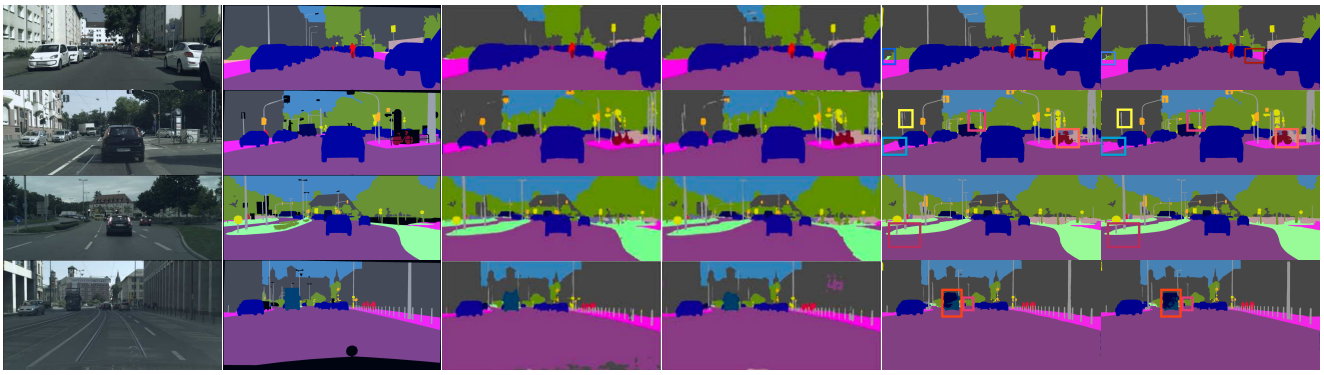


Figure 3. **Qualitative results of selected examples on Cityscapes**.From left to right: RGB Input Image, GT, CCNet [27], DANet [24], HRNet [68], OVeNet. Best viewed on a screen and zoomed in.

Table 2. **Per Class Results on Cityscapes test set**

Method	road	sidew.	build.	wall	fence	pole	light	sign	veget.	terrain	sky	person	rider	car	truck	bus	train	motorc.	bicycle	mIoU
HRNet [68]	98.73	**87.49**	93.65	56.48	61.57	**71.57**	78.76	81.81	93.99	74.11	95.68	87.95	73.72	96.35	69.94	82.52	76.93	70.88	**78.02**	80.40
OVeNet	**98.74**	87.41	**93.79**	**61.65**	**64.00**	71.35	**78.98**	**81.65**	**94.00**	**73.42**	**95.81**	**87.99**	**74.36**	**96.42**	**74.76**	**87.70**	**82.83**	**71.77**	77.86	**81.82**
HRNet + OCR	98.77	**87.85**	93.72	57.75	63.92	**71.74**	**78.56**	**81.77**	**94.06**	**73.69**	95.68	88.04	74.64	96.46	76.40	88.78	84.63	71.79	**78.63**	81.90
OVeNet (*HRNet + OCR*)	**98.79**	87.47	**93.86**	**62.97**	**64.41**	70.80	78.45	81.13	93.99	73.31	**95.72**	**88.08**	**74.90**	**96.47**	**76.95**	**89.95**	**88.48**	**71.87**	78.43	**82.42**

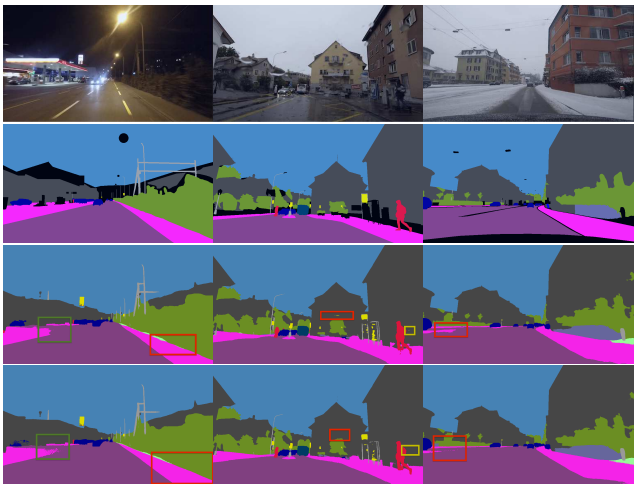


Figure 4. **Qualitative results of selected examples on ACDC**. From top to bottom: RGB, GT, HRNet [68], OVeNet. Best viewed on a screen and zoomed in.

we can underline that our model tries to enlarge correctly the sidewalk segments in both red and green frames and reduces the erroneous terrain segment predicted by HRNet. As far as the second example is concerned, HRNet classifies incorrectly the sign of the house as traffic sign (red frame). On the contrary, our model corrects not only this mistake, but also a discontinuity occurring in the yellow frame. Last but not least, regarding the last set of materials, our offset vector-based model eliminates correctly the sidewalk area (red frame), which does not exist in the ground truth.

ADE20K. We also achieve far better results than HRNet [68] under similar training time. HRNet achieves 44.6% mIoU, 80.7% PixelAcc and 58.2% MeanAcc while OVeNet surpasses it, reaching 45.3% mIoU, 81.3% PixelAcc and 58.7% MeanAcc.

Qualitative results on ADE20K support the above findings, as shown in Fig. 5. To be more specific, in the first column, we can underline that our model tries to enlarge correctly the wall and carpet segments (blue and yellow frames). Regarding the second example, our model corrects HRNet's false prediction on the sign but also a discontinuity occurring in the road. Regarding the last example, although our offset vector-based model has some false prediction in the bridge segment (yellow frame), it corrects many erroneously classified pixels leading to a better total prediction.

4.3. Ablation study

In order to experimentally confirm our design choices for the offset vector-based model, we performed an ablation study, as shown in Table 5. We trained and evaluated 7 different variants on Cityscapes. The performance of each model variation in relation to the ground-truth images was calculated by means of the mIoU. At first, we initialized both heads of the network with the pre-trained Imagenet weights and set the offset vector length equal to 0.5. Secondly, we froze both main body's and initial head's weights. The frozen part of our model was initialized with the corresponding Cityscapes final pre-trained weights. The only part trained was the second head, which was initialized with pre-trained ImageNet weights. As shown in Table 5, although the performance of our model is higher than the ini-

Table 3. **Per Class Results on ACDC test set.** OVeNet is built on HRNet.

Condition	Method	road	sidew.	build.	wall	fence	pole	light	sign	veget.	terrain	sky	person	rider	car	truck	bus	train	motorc.	bicycle	mIoU
Fog	HRNet	95.3	81.1	89.7	54.2	48.1	60.3	76.2	76.0	89.1	78.1	98.3	**58.8**	60.1	83.3	50.5	47.9	**86.4**	49.8	36.3	69.3
	OVeNet	**95.7**	**82.1**	**90.6**	**55.5**	**50.4**	**63.9**	**79.3**	**78.2**	**89.6**	**78.8**	**98.4**	56.5	**63.8**	**85.7**	**60.3**	**63.8**	85.5	**52.7**	**39.4**	**72.1**
Night	HRNet	95.4	78.1	86.8	46.5	41.6	59.1	65.8	65.2	77.2	42.0	86.2	66.2	40.1	80.5	**16.6**	31.0	86.2	37.1	49.2	60.6
	OVeNet	**95.6**	**79.4**	**87.4**	**48.5**	**42.3**	**61.1**	**68.2**	**69.9**	**78.1**	**43.8**	**86.4**	**68.6**	**44.4**	**82.0**	10.9	**44.3**	**87.3**	**43.3**	**51.3**	**62.8**
Rain	HRNet	95.7	83.9	93.6	60.2	62.7	70.1	80.9	79.7	94.4	51.9	98.7	72.4	18.7	92.5	67.0	85.3	87.3	50.9	66.4	74.5
	OVeNet	**96.4**	**86.5**	**94.3**	**65.2**	**64.6**	**72.6**	**83.6**	**82.2**	**94.6**	**54.3**	**98.7**	**76.4**	**21.8**	**93.4**	**75.7**	**88.9**	**89.1**	**50.9**	**66.4**	**76.6**
Snow	HRNet	95.0	79.6	91.4	49.6	58.2	69.7	86.6	78.6	92.9	59.4	97.9	77.2	24.5	91.7	53.4	56.2	90.1	39.8	66.8	71.5
	OVeNet	**95.7**	**81.9**	**92.4**	**55.3**	**59.8**	**71.9**	**88.1**	**80.5**	**93.4**	**60.7**	**98.0**	**79.8**	**25.7**	**92.7**	**59.6**	**65.4**	**91.6**	**45.2**	**70.3**	**74.1**
All	HRNet	95.3	80.3	90.5	52.0	53.1	65.1	78.2	74.2	89.2	68.4	96.7	70.6	36.1	88.2	55.9	54.3	88.0	43.8	58.9	70.5
	OVeNet	**95.8**	**82.1**	**91.3**	**55.8**	**54.6**	**67.6**	**80.5**	**77.3**	**89.7**	**69.5**	**96.8**	**73.4**	**39.1**	**89.5**	**61.9**	**65.0**	**89.4**	**47.2**	**60.6**	**73.0**

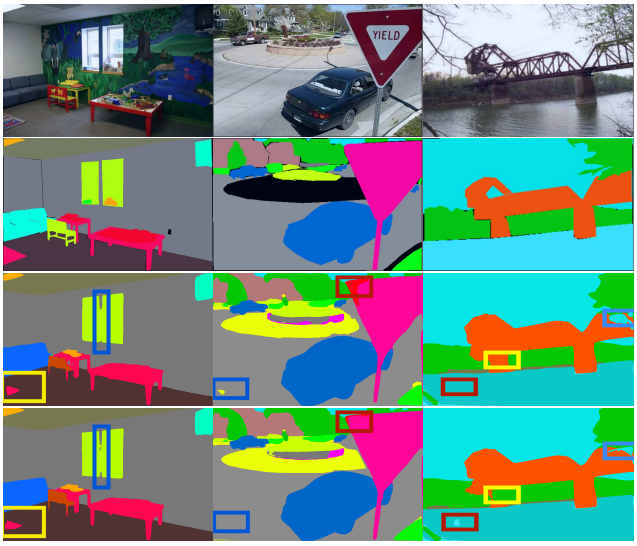


Figure 5. **Qualitative results of selected examples on ADE20K**. Up to down: RGB, GT, HRNet [68], OVeNet. Best viewed on a screen and zoomed in.

tial single-head model's one, it still remains lower than the case where both heads are trained simultaneously. Then, we deactivated the "Frozen" feature and changed the offset vector's length logarithmically, setting it either to 1 or 0.2. We observed that in both cases the performance is lower than that for 0.5. This is due to the fact that larger offset vectors point to more distant objects that may affect erroneously the final prediction, while smaller vectors do not exploit too much information from neighboring classes. Furthermore, we deactivated the OHEM Cross Entropy Loss and enabled the simple Cross Entropy Loss. As expected, the performance of the model was lower. OHEM penalizes high loss values more and leads to a better training of the model. Lastly, HRNet [68] consists of 4 stages. In all the previous cases, branch occurred in the last (4^{th}) stage so as not to overload the new network with many extra parameters. When branching in the 3^{rd} stage, the performance did not improve.

Table 4. **Comparison of the models on different conditions of ACDC.**

Method	**Fog**	**Night**	**Rain**	**Snow**	**All**
RefineNet [39]	65.7	55.5	68.7	65.9	65.3
DeepLabv2 [14]	54.5	45.3	59.3	57.1	55.3
DeepLabv3+ [17]	69.1	60.9	74.1	69.6	70.0
HRNet [68]	69.3	60.6	74.5	71.5	70.5
OVeNet	**72.1**	**62.8**	**76.6**	**74.1**	**73.0**

Table 5. **Ablation study of components of our method.** "**Fr**": Frozen main body's and initial head's weights initialized with HRNet's final Cityscapes weights, "**Br**": Branch, "τ": Offset Vector Length, "**OHEM**": OHEM Cross Entropy.

Fr	**Br**	τ	**OHEM**	**mIoU**
			✓	81.83
	4	0.5	✓	**82.40**
✓	4	0.5	✓	82.01
	4	1	✓	81.79
	4	0.2	✓	82.30
	4	0.2		81.96
	3	0.2	✓	82.20

5. Conclusion

All in all, we have presented OVeNet, a supervised model for semantic segmentation, which selectively exploits information from neighboring pixels to improve initial semantic predictions. OVeNet excels both in global and per-class performance across most classes on three widely used semantic segmentation benchmarks. By correcting misclassified pixels, it reduces discontinuities and improves the shapes of segments, leading to more realistic results. This is a highly relevant contribution for real-world applications that depend on semantic segmentation, such as autonomous cars or medical imaging and diagnostics.

References

[1] Matthew Amodio, Feng Gao, Arman Avesta, Sanjay Aneja, Lucian V Del Priore, Jay Wang, and Smita Krishnaswamy. Cuts: A fully unsupervised framework for medical image segmentation. *arXiv preprint arXiv:2209.11359*, 2022. 1

[2] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(12):2481–2495, 2017. 2

[3] Jonathan C Balloch, Varun Agrawal, Irfan Essa, and Sonia Chernova. Unbiasing semantic segmentation for robot perception using synthetic data feature transfer. *arXiv preprint arXiv:1809.03676*, 2018. 1

[4] Gedas Bertasius, Jianbo Shi, and Lorenzo Torresani. Semantic segmentation with boundary neural fields. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3602–3610, 2016. 3

[5] Shubhankar Borse, Hong Cai, Yizhe Zhang, and Fatih Porikli. Hs3: Learning with proper task complexity in hierarchically supervised semantic segmentation. *arXiv preprint arXiv:2111.02333*, 2021. 2

[6] Shubhankar Borse, Ying Wang, Yizhe Zhang, and Fatih Porikli. Inverseform: A loss function for structured boundary-aware segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5901–5911, 2021. 3

[7] Adrian Bulat and Georgios Tzimiropoulos. Human pose estimation via convolutional part heatmap regression. In *ECCV*, pages 717–732, 2016. 2

[8] Adrian Bulat and Georgios Tzimiropoulos. Binarized convolutional landmark localizers for human pose estimation and face alignment with limited resources. In *ICCV*, pages 3726–3734, 2017. 2

[9] Senay Cakir, Marcel Gauß, Kai Häppeler, Yassine Ounajjar, Fabian Heinle, and Reiner Marchthaler. Semantic segmentation for autonomous driving: Model evaluation, dataset generation, perspective comparison, and real-time capability. *arXiv preprint arXiv:2207.12939*, 2022. 1

[10] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans. Pattern Anal. Mach. Intell.*, 40(4):834–848, 2018. 2, 6

[11] Liang-Chieh Chen, Yi Yang, Jiang Wang, Wei Xu, and Alan L. Yuille. Attention to scale: Scale-aware semantic image segmentation. In *CVPR*, pages 3640–3649, 2016. 2

[12] Liang-Chieh Chen, Jonathan T Barron, George Papandreou, Kevin Murphy, and Alan L Yuille. Semantic image segmentation with task-specific edge detection using cnns and a discriminatively trained domain transform. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4545–4554, 2016. 3

[13] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv preprint arXiv:1412.7062*, 2014. 2

[14] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017. 1, 2, 3, 8

[15] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 2

[16] Liang-Chieh Chen, Alexander Schwing, Alan Yuille, and Raquel Urtasun. Learning deep structured models. In *International Conference on Machine Learning*, pages 1785–1794. PMLR, 2015. 3

[17] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018. 1, 2, 8

[18] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded pyramid network for multi-person pose estimation. *CoRR*, abs/1711.07319, 2017. 2

[19] Zhe Chen, Yuchen Duan, Wenhai Wang, Junjun He, Tong Lu, Jifeng Dai, and Yu Qiao. Vision transformer adapter for dense predictions. *arXiv preprint arXiv:2205.08534*, 2022. 3

[20] Xiao Chu, Wei Yang, Wanli Ouyang, Cheng Ma, Alan L. Yuille, and Xiaogang Wang. Multi-context attention for human pose estimation. In *CVPR*, pages 5669–5678, 2017. 2

[21] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 1, 2, 5

[22] Jiankang Deng, George Trigeorgis, Yuxiang Zhou, and Stefanos Zafeiriou. Joint multi-view face alignment in the wild. *CoRR*, abs/1708.06023, 2017. 2

[23] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 3

[24] Jun Fu, Jing Liu, Haijie Tian, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. *CoRR*, abs/1809.02983, 2018. 6, 7

[25] Jun Fu, Jing Liu, Yuhang Wang, Yong Li, Yongjun Bao, Jinhui Tang, and Hanqing Lu. Adaptive context network for scene parsing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6748–6757, 2019. 2

[26] Junjun He, Zhongying Deng, Lei Zhou, Yali Wang, and Yu Qiao. Adaptive pyramid context network for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7519–7528, 2019. 2

[27] Zilong Huang, Xinggang Wang, Lichao Huang, Chang Huang, Yunchao Wei, and Wenyu Liu. Ccnet: Criss-cross attention for semantic segmentation. *CoRR*, abs/1811.11721, 2018. 6, 7

[28] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2462–2470, 2017. 4

[29] Md. Amirul Islam, Mrigank Rochan, Neil D. B. Bruce, and Yang Wang. Gated feedback refinement network for dense image labeling. In *CVPR*, pages 4877–4885, 2017. 2

[30] Çağrı Kaymak and Ayşegül Uçar. Semantic image segmentation for autonomous driving using fully convolutional networks. In *2019 International Artificial Intelligence and Data Processing Symposium (IDAP)*, pages 1–8, 2019. 1

[31] Lipeng Ke, Ming-Ching Chang, Honggang Qi, and Siwei Lyu. Multi-scale structure-aware network for human pose estimation. *CoRR*, abs/1803.09894, 2018. 2

[32] Tsung-Wei Ke, Jyh-Jing Hwang, Ziwei Liu, and Stella X Yu. Adaptive affinity fields for semantic segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 587–602, 2018. 1, 3

[33] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. *Advances in neural information processing systems*, 24, 2011. 3

[34] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012. 2, 5

[35] Ke Li, Bharath Hariharan, and Jitendra Malik. Iterative instance segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3659–3667, 2016. 2

[36] Zeming Li, Chao Peng, Gang Yu, Xiangyu Zhang, Yangdong Deng, and Jian Sun. Detnet: Design backbone for object detection. In *Proceedings of the European conference on computer vision (ECCV)*, pages 334–350, 2018. 2

[37] Xiaodan Liang, Hongfei Zhou, and Eric Xing. Dynamic-structured semantic propagation network. In *CVPR*, pages 752–761, 2018. 6

[38] Di Lin, Dingguo Shen, Siting Shen, Yuanfeng Ji, Dani Lischinski, Daniel Cohen-Or, and Hui Huang. Zigzagnet: Fusing top-down and bottom-up context for object segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7490–7499, 2019. 2

[39] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian Reid. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1925–1934, 2017. 8

[40] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian D. Reid. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *CVPR*, pages 5168–5177, 2017. 6

[41] Tsung-Yi Lin, Piotr Dollár, Ross B. Girshick, Kaiming He, Bharath Hariharan, and Serge J. Belongie. Feature pyramid networks for object detection. In *CVPR*, pages 936–944, 2017. 2

[42] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 1

[43] Chenxi Liu, Liang-Chieh Chen, Florian Schroff, Hartwig Adam, Wei Hua, Alan Yuille, and Li Fei-Fei. Auto-deeplab: Hierarchical neural architecture search for semantic image segmentation. *arXiv preprint arXiv:1901.02985*, 2019. 6

[44] Sifei Liu, Shalini De Mello, Jinwei Gu, Guangyu Zhong, Ming-Hsuan Yang, and Jan Kautz. Learning affinity via spatial propagation networks. *Advances in Neural Information Processing Systems*, 30, 2017. 3

[45] Ziwei Liu, Xiaoxiao Li, Ping Luo, Chen-Change Loy, and Xiaoou Tang. Semantic image segmentation via deep parsing network. In *Proceedings of the IEEE international conference on computer vision*, pages 1377–1385, 2015. 3

[46] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 3

[47] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. 1

[48] Michael Maire, Takuya Narihira, and Stella X Yu. Affinity cnn: Learning pixel-centric pairwise relations for figure/ground embedding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 174–182, 2016. 3

[49] Davy Neven, Bert De Brabandere, Marc Proesmans, and Luc Van Gool. Instance segmentation by jointly optimizing spatial embeddings and clustering bandwidth. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8837–8845, 2019. 1, 3

[50] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *ECCV*, pages 483–499, 2016. 2

[51] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *Proceedings of the IEEE international conference on computer vision*, pages 1520–1528, 2015. 1, 2

[52] Jinsun Park, Kyungdon Joo, Zhe Hu, Chi-Kuei Liu, and In So Kweon. Non-local spatial propagation network for depth completion. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIII 16*, pages 120–136. Springer, 2020. 3

[53] Vaishakh Patil, Christos Sakaridis, Alexander Liniger, and Luc Van Gool. P3depth: Monocular depth estimation with a piecewise planarity prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1610–1621, June 2022. 3

[54] Chao Peng, Xiangyu Zhang, Gang Yu, Guiming Luo, and Jian Sun. Large kernel matters - improve semantic segmentation by global convolutional network. In *CVPR*, pages 1743–1751, 2017. 2

[55] Xi Peng, Rogério Schmidt Feris, Xiaoyu Wang, and Dimitris N. Metaxas. A recurrent encoder-decoder network for sequential face alignment. In *ECCV (1)*, volume 9905, pages 38–56, 2016. 2

[56] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015. 3

[57] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015. 1, 2

[58] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. ACDC: The adverse conditions dataset with correspondences for semantic driving scene understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10765–10775, 2021. 1, 2, 5

[59] Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, and Yann LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv:1312.6229*, 2013. 2

[60] Evan Shelhamer, Jonathan Long, and Trevor Darrell. Fully convolutional networks for semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(4):640–651, 2017. 2

[61] Abhinav Shrivastava, Abhinav Gupta, and Ross Girshick. Training region-based object detectors with online hard example mining. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 6

[62] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 2

[63] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015. 2

[64] Wei Tang, Pei Yu, and Ying Wu. Deeply learned compositional models for human pose estimation. In *ECCV*, September 2018. 2

[65] Maria Tzelepi and Anastasios Tefas. Semantic scene segmentation for robotics applications. In *2021 12th International Conference on Information, Intelligence, Systems & Applications (IISA)*, pages 1–4. IEEE, 2021. 1

[66] Roberto Valle, José Miguel Buenaposada, Antonio Valdés, and Luis Baumela. A deeply-initialized coarse-to-fine ensemble of regression trees for face alignment. In *ECCV*, pages 609–624, 2018. 2

[67] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 3

[68] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 43(10):3349–3364, 2020. 1, 2, 5, 6, 7, 8

[69] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 568–578, 2021. 3

[70] Zifeng Wu, Chunhua Shen, and Anton van den Hengel. Wider or deeper: Revisiting the resnet model for visual recognition. *CoRR*, abs/1611.10080, 2016. 6

[71] Bin Xiao, Haiping Wu, and Yichen Wei. Simple baselines for human pose estimation and tracking. In *ECCV*, pages 472–487, 2018. 2

[72] Dan Xu, Wanli Ouyang, Xiaogang Wang, and Nicu Sebe. Pad-net: Multi-tasks guided prediction-and-distillation network for simultaneous depth estimation and scene parsing. In *CVPR*, pages 675–684, 2018. 6

[73] Jing Yang, Qingshan Liu, and Kaihua Zhang. Stacked hourglass network for robust facial landmark localisation. In *CVPR*, pages 2025–2033, 2017. 2

[74] Maoke Yang, Kun Yu, Chi Zhang, Zhiwei Li, and Kuiyuan Yang. Denseaspp for semantic segmentation in street scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3684–3692, 2018. 2

[75] Wei Yang, Shuang Li, Wanli Ouyang, Hongsheng Li, and Xiaogang Wang. Learning feature pyramids for human pose estimation. In *ICCV*, pages 1290–1299, 2017. 2

[76] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. In Yoshua Bengio and Yann LeCun, editors, *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016. 1, 2

[77] Yuhui Yuan, Xiaokang Chen, Xilin Chen, and Jingdong Wang. Segmentation transformer: Object-contextual representations for semantic segmentation. *arXiv preprint arXiv:1909.11065*, 2019. 1, 2, 3

[78] Yuhui Yuan, Xilin Chen, and Jingdong Wang. Object-contextual representations for semantic segmentation. *CoRR*, abs/1909.11065, 2019. 2, 5, 6

[79] Yuhui Yuan, Jingyi Xie, Xilin Chen, and Jingdong Wang. Segfix: Model-agnostic boundary refinement for segmentation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XII 16*, pages 489–506. Springer, 2020. 3

[80] Hang Zhang, Han Zhang, Chenguang Wang, and Junyuan Xie. Co-occurrent features in semantic segmentation. In *CVPR*, June 2019. 6

[81] Zhenli Zhang, Xiangyu Zhang, Chao Peng, Xiangyang Xue, and Jian Sun. Exfuse: Enhancing feature fusion for semantic segmentation. In *ECCV*, pages 273–288, 2018. 2

[82] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *CVPR*, pages 6230–6239, 2017. 2, 6

[83] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017. 2

[84] Hengshuang Zhao, Yi Zhang, Shu Liu, Jianping Shi, Chen Change Loy, Dahua Lin, and Jiaya Jia. Psanet: Point-wise spatial attention network for scene parsing. In *ECCV*, pages 270–286, 2018. 6

[85] Shuai Zheng, Sadeep Jayasumana, Bernardino Romera-Paredes, Vibhav Vineet, Zhizhong Su, Dalong Du, Chang Huang, and Philip HS Torr. Conditional random fields as recurrent neural networks. In *Proceedings of the IEEE international conference on computer vision*, pages 1529–1537, 2015. 3

[86] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *arXiv preprint arXiv:1608.05442*, 2016. 2, 5

[87] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 2