Improving Semantic Segmentation via Video Propagation and Label Relaxation

Yi Zhu[1*]   Karan Sapra[2*]   Fitsum A. Reda[2]   Kevin J. Shih[2]   Shawn Newsam[1]
Andrew Tao[2]   Bryan Catanzaro[2]
[1]University of California at Merced   [2]Nvidia Corporation

{yzhu25,snewsam}@ucmerced.edu   {ksapra,freda,kshih,atao,bcatanzaro}@nvidia.com

Abstract

Semantic segmentation requires large amounts of pixel-wise annotations to learn accurate models. In this paper, we present a video prediction-based methodology to scale up training sets by synthesizing new training samples in order to improve the accuracy of semantic segmentation networks. We exploit video prediction models' ability to predict future frames in order to also predict future labels. A joint propagation strategy is also proposed to alleviate mis-alignments in synthesized samples. We demonstrate that training segmentation models on datasets augmented by the synthesized samples leads to significant improvements in accuracy. Furthermore, we introduce a novel boundary label relaxation technique that makes training robust to annotation noise and propagation artifacts along object boundaries. Our proposed methods achieve state-of-the-art mIoUs of 83.5% on Cityscapes and 82.9% on CamVid. Our single model, without model ensembles, achieves 72.8% mIoU on the KITTI semantic segmentation test set, which surpasses the winning entry of the ROB challenge 2018.

1. Introduction

Semantic segmentation is the task of dense per pixel predictions of semantic labels. Large improvements in model accuracy have been made in recent literature [44, 14, 10], in part due to the introduction of Convolutional Neural Networks (CNNs) for feature learning, the task's utility for self-driving cars, and the availability of larger and richer training datasets (*e.g.*, Cityscapes [15] and Mapillary Vista [32]). While these models rely on large amounts of training data to achieve their full potential, the dense nature of semantic segmentation entails a prohibitively expensive dataset annotation process. For instance, annotating all pixels in a 1024×2048 Cityscapes image takes on average 1.5 hours [15]. Annotation quality plays an important role for training better models. While coarsely annotating large contiguous
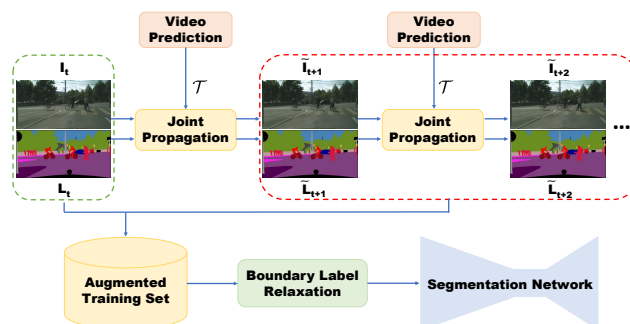

Figure 1: Framework overview. We propose joint image-label propagation to scale up training sets for robust semantic segmentation. The green dashed box includes manually labelled samples, and the red dashed box includes our propagated samples. $\mathcal{T}$ is the transformation function learned by the video prediction models to perform propagation. We also propose boundary label relaxation to mitigate label noise during training. Our framework can be used with most semantic segmentation and video prediction models.

regions can be performed quickly using annotation toolkits, finely labeling pixels along object boundaries is extremely challenging and often involves inherently ambiguous pixels.

Many alternatives have been proposed to augment training processes with additional data. For example, Cords *et al*. [15] provided 20K coarsely annotated images to help train deep CNNs, an annotation cost effective alternative used by all top 10 performers on the Cityscapes benchmark. Nevertheless, coarse labeling still takes, on average, 7 minutes per image. An even cheaper way to obtain more labeled samples is to generate synthetic data [35, 36, 18, 47, 45]. However, model accuracy on the synthetic data often does not generalize to real data due to the domain gap between synthetic and real images. Luc *et al*. [28] use a state-of-the-art image segmentation method [42] as a teacher to generate extra annotations for unlabelled images. However, their performance is bounded by the teacher method. Another approach exploits the fact that many semantic segmentation datasets are based on continuous video frame sequences sparsely labeled at regular intervals. As such, several works [2, 9, 31, 16, 33] propose to use temporal

consistency constraints, such as optical flow, to propagate ground truth labels from labeled to unlabeled frames. However, these methods all have different drawbacks.

In this work, we propose to utilize video prediction models to efficiently create more training samples (image-label pairs) as shown in Fig. 1. Given a sequence of video frames having labels for only a subset of the frames, we exploit the prediction models' ability to predict future frames in order to also predict future labels (new labels for unlabelled frames). Specifically, we propose leveraging such models in two ways. **1) Label Propagation (LP)**: We create new training samples by pairing a propagated label with the original future frame. **2) Joint image-label Propagation (JP)**: We create a new training sample by pairing a propagated label with the corresponding propagated image. In approach (2), it is of note that since both past labels and frames are jointly propagated using the same prediction model, the resulting image-label pair will have a higher degree of alignment. We separately apply each approach for multiple future steps to scale up the training dataset.

While great progress has been made in video prediction, it is still prone to producing unnatural distortions along object boundaries. For synthesized training examples, this means that the propagated labels along object boundaries should be trusted less than those within an object's interior. Here, we present a novel **boundary label relaxation** technique that can make training more robust to such errors. We demonstrate that by maximizing the likelihood of the *union* of neighboring class labels along the boundary, the trained models not only achieve better accuracy, but are also able to benefit from longer-range propagation.

As we will show in our experiments, training segmentation models on datasets augmented by our synthesized samples leads to improvements on several popular datasets. Furthermore, by performing training with our proposed boundary label relaxation technique, we achieve even higher accuracy and training robustness, producing state-of-the-art results on the Cityscapes, CamVid, and KITTI semantic segmentation benchmarks. Our contributions are summarized below:

- We propose to utilize video prediction models to propagate labels to immediate neighbor frames.

- We introduce joint image-label propagation to alleviate the mis-alignment problem.

- We propose to relax one-hot label training by maximizing the likelihood of the union of class probabilities along boundary. This results in more accurate models and allows us to perform longer-range propagation.

- We compare our video prediction-based approach to standard optical flow-based ones in terms of segmentation performance.

2. Related Work

Here, we discuss additional work related to ours, focusing mainly on the differences.

Label propagation There are two main approaches to propagating labels: patch matching [2, 9] and optical flow [31, 16, 33]. Patch matching-based methods, however, tend to be sensitive to patch size and threshold values, and, in some cases, they assume prior-knowledge of class statistics. Optical flow-based methods rely on very accurate optical flow estimation, which is difficult to achieve. Erroneous flow estimation can result in propagated labels that are mis-aligned with their corresponding frames.

Our work falls in this line of research but has two major differences. First, we use motion vectors learned from video prediction models to perform propagation. The learned motion vectors can handle occlusion while also being class agnostic. Second, we conduct joint image-label propagation to greatly reduce the mis-alignments.

Boundary handling Some prior works [12, 29] explicitly incorporate edge cues as constraints to handle boundary pixels. Although the idea is straightforward, this approach has at least two drawbacks. One is the potential error propagation from edge estimation and the other is fitting extremely hard boundary cases may lead to over-fitting at the test stage. There is also literature focusing on structure modeling to obtain better boundary localization, such as affinity field [21], random walk [5], relaxation labelling [37], boundary neural fields [4], etc. However, none of these methods deals directly with boundary pixels but they instead attempt to model the interactions between segments along object boundaries. The work most similar to ours is [22] which proposes to incorporate uncertainty reasoning inside Bayesian frameworks. The authors enforce a Gaussian distribution over the logits to attenuate loss when uncertainty is large. Instead, we propose a modification to class label space that allows us to predict multiple classes at a boundary pixel. Experimental results demonstrate higher model accuracy and increased training robustness.

3. Methodology

We present an approach for training data synthesis from sparsely annotated video frame sequences. Given an input video $\mathbf{I} \in \mathbb{R}^{n \times W \times H}$ and semantic labels $\mathbf{L} \in \mathbb{R}^{m \times W \times H}$, where $m \leq n$, we synthesize $k \times m$ new training samples (image-label pairs) using video prediction models, where k is the length of propagation applied to each input image-label pair $(\mathbf{I}_i, \mathbf{L}_i)$. We will first describe how we use video prediction models for label synthesis.

3.1. Video Prediction

Video prediction is the task of generating future frames from a sequence of past frames. It can be modeled as the
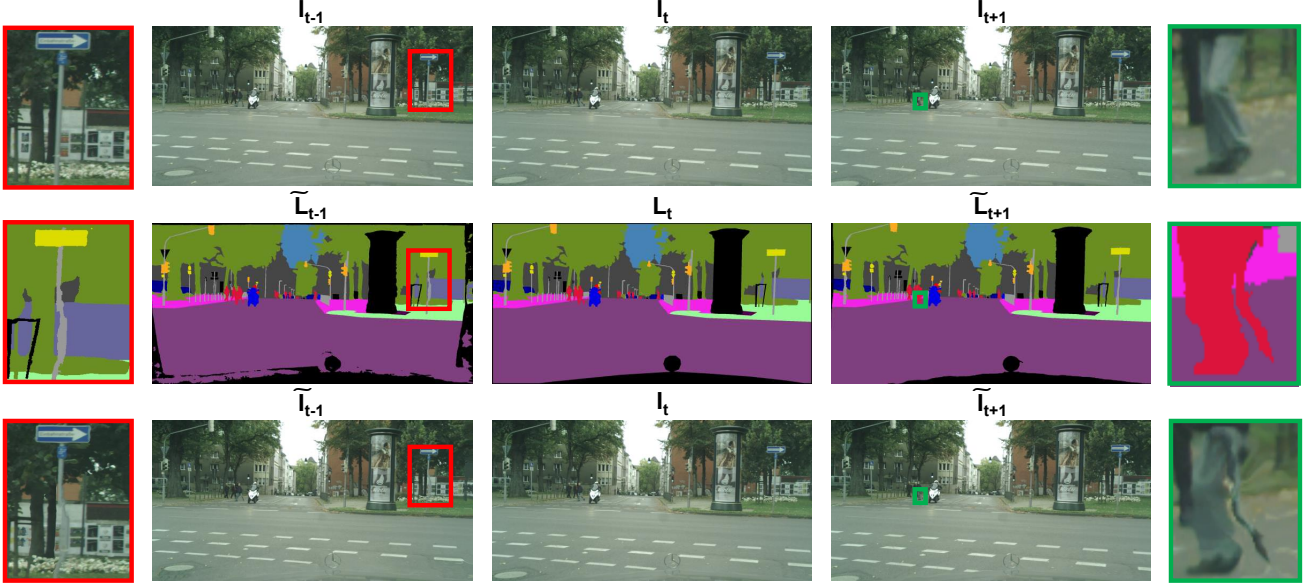
Figure 2: Motivation of joint image-label propagation. Row 1: original frames. Row 2: propagated labels. Row 3: propagated frames. The red and green boxes are two zoomed-in regions which demonstrate the mis-alignment problem. Note how the propagated frames align perfectly with propagated labels as compared to the original frames. The black areas in the labels represent a void class. (Image brightness has been adjusted for better visualization.)

process of direct pixel synthesis or learning to transform past pixels. In this work, we use a simple and yet effective vector-based approach [34] that predicts a motion vector (u, v) to translate each pixel (x, y) to its future coordinate. The predicted future frame $\widetilde{\mathbf{I}}_{t+1}$ is given by,

$$\widetilde{\mathbf{I}}_{t+1} = \mathcal{T}\Big(\mathcal{G}\big(\mathbf{I}_{1:t}, \mathbf{F}_{2:t}\big), \mathbf{I}_t\Big), \qquad (1)$$

where $\mathcal{G}$ is a 3D CNN that predicts motion vectors (u, v) conditioned on input frames $\mathbf{I}_{1:t}$ and estimated optical flows $\mathbf{F}_i$ between successive input frames $\mathbf{I}_i$ and $\mathbf{I}_{i-1}$. $\mathcal{T}$ is an operation that bilinearly samples from the most recent input $\mathbf{I}_t$ using the predicted motion vectors (u, v).

Note that the motion vectors predicted by $\mathcal{G}$ are not equivalent to optical flow vectors $\mathbf{F}$. Optical flow vectors are undefined for pixels that are visible in the current frame but not visible in the previous frame. Thus, performing past frame sampling using optical flow vectors will duplicate foreground objects, create undefined holes or stretch image borders. The learned motion vectors, however, account for disocclusion and attempt to accurately predict future frames. We will demonstrate the advantage of learned motion vectors over optical flow in Sec. 4.

In this work, we propose to *reuse* the predicted motion vectors to also synthesize future labels $\widetilde{\mathbf{L}}_{t+1}$. Specifically:

$$\widetilde{\mathbf{L}}_{t+1} = \mathcal{T}\Big(\mathcal{G}\big(\mathbf{I}_{1:t}, \mathbf{F}_{2:t}\big), \mathbf{L}_t\Big), \qquad (2)$$

where a sampling operation $\mathcal{T}$ is applied on a past label $\mathbf{L}_t$. $\mathcal{G}$ in equation 2 is the same as in equation 1 and is pre-

trained on the underlying video frame sequences for the task of accurately predicting future frames.

3.2. Joint Image-Label Propagation

Standard label propagation techniques create new training samples by pairing a propagated label with the original future frame as $\big(\mathbf{I}_{i+k}, \widetilde{\mathbf{L}}_{i+k}\big)$, with k being the propagation length. For regions where the frame-to-frame correspondence estimation is not accurate, we will encounter mis-alignment between $\mathbf{I}_{i+k}$ and $\widetilde{\mathbf{L}}_{i+k}$. For example, as we see in Fig. 2, most regions in the propagated label (row 2) correlate well with the corresponding original video frames (row 1). However, certain regions, like the pole (red) and the leg of the pedestrian (green), do not align with the original frames due to erroneous estimated motion vectors.

To alleviate this mis-alignment issue, we propose a joint image-label propagation strategy; *i.e.*, we jointly propagate both the video frame and the label. Specifically, we apply equation 2 to each input training sample $(\mathbf{I}_i, \mathbf{L}_i)$ for k future steps to create km new training samples by pairing a predicted frame with a predicted label as $(\widetilde{\mathbf{I}}_{i+k}, \widetilde{\mathbf{L}}_{i+k})$. As we can see in Fig. 2, the propagated frames (row 3) correspond well to the propagated labels (row 2). The pole and the leg experience the same distortion. Since semantic segmentation is a dense per-pixel estimation problem, such good alignment is crucial for learning an accurate model.

Our joint propagation approach can be thought of as a special type of data augmentation because both the frame and label are synthesized by transforming a past frame and the corresponding label using the same learned transforma-
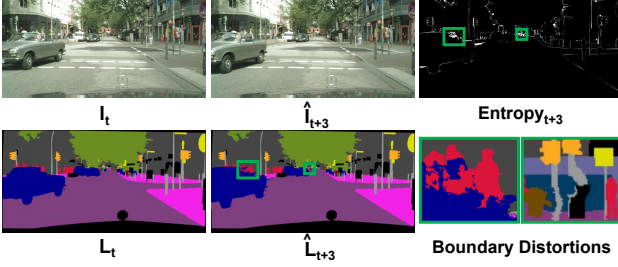
Figure 3: Motivation of boundary label relaxation. For the entropy image, the lighter pixel value, the larger the entropy. We find that object boundaries often have large entropy, due to ambiguous annotations or propagation distortions. The green boxes are zoomed-in figures showing such distortions.

tion parameters (u, v). It is an approach similar to standard data augmentation techniques, such as random rotation, random scale or random flip. However, joint propagation uses a more fundamental transformation which was trained for the task of accurate future frame prediction.

In order to create more training samples, we also perform reversed frame prediction. We equivalently apply joint propagation to create additional km new training samples as $(\widetilde{\mathbf{I}}_{i-k}, \widetilde{\mathbf{L}}_{i-k})$. In total, we can scale the training dataset by a factor of $2k+1$. In our study, we set k to be $\pm 1, \pm 2, \pm 3, \pm 4$ or ± 5, where $+$ indicates a forward propagation, and $-$ a backward propagation.

We would like to point out that our proposed joint propagation has broader applications. It could also find application in datasets where *both* the raw frames and the corresponding labels are scarce. This is different from label propagation alone for synthesizing new training samples for typical video datasets, for instance Cityscapes [15], where raw video frames are abundant but only a subset of the frames have human annotated labels.

3.3. Video Reconstruction

Since, in our problem, we know the actual future frames, we can instead perform not just video prediction but video reconstruction to synthesize new training examples. More specifically, we can condition the prediction models on both the past and future frames to more accurately reconstruct "future" frames. The motivation behind this reformulation is that because future frames are observed by video reconstruction models, they are, in general, expected to produce better transformation parameters than video prediction models which only observe only past frames.

Mathematically, a reconstructed future frame $\hat{\mathbf{I}}_{t+1}$ is given by,

$$\hat{\mathbf{I}}_{t+1} = \mathcal{T}\Big(\mathcal{G}\big(\mathbf{I}_{1:t+1}, \mathbf{F}_{2:t+1}\big), \mathbf{I}_t\Big). \qquad (3)$$

In a similar way to equation 2, we also apply $\mathcal{G}$ from equation 3 (which is learned for the task of accurate future frame reconstruction) to generate a future label $\hat{\mathbf{L}}_{t+1}$.

Table 1: Effectiveness of Mapillary pre-training and class uniform sampling on both fine and coarse annotations.

Method	mIoU (%)
Baseline	76.60
+ Mapillary Pre-training	78.32
+ Class Uniform Sampling (Fine + Coarse)	**79.46**

3.4. Boundary Label Relaxation

Most of the hardest pixels to classify lie on the boundary between object classes [25]. Specifically, it is difficult to classify the center pixel of a receptive field when potentially half or more of the input context could be from a different class. This problem is further compounded by the fact that the annotations are nowhere near pixel-perfect along the edges.

We propose a modification to class label space, applied exclusively during training, that allows us to predict multiple classes at a boundary pixel. We define a boundary pixel as any pixel that has a differently labeled neighbor. Suppose we are classifying a pixel along the boundary of classes A and B for simplicity. Instead of maximizing the likelihood of the target label as provided by annotation, we propose to maximize the likelihood of $P(A \cup B)$. Because classes A and B are mutually exclusive, we aim to maximize the union of A and B:

$$P(A \cup B) = P(A) + P(B), \qquad (4)$$

where $P()$ is the softmax probability of each class. Specifically, let $\mathcal{N}$ be the set of classes within a 3×3 window of a pixel. We define our loss as:

$$\mathcal{L}_{boundary} = -log \sum_{C \in \mathcal{N}} P(C). \qquad (5)$$

Note that for $|C| = 1$, this loss reduces to the standard one-hot label cross-entropy loss.

One can see that the loss over the modified label space is minimized when $\sum_{C \in \mathcal{N}} P(C) = 1$ without any constraints on the relative values of each class probability. We demonstrate that this relaxation not only makes our training robust to the aforementioned annotation errors, but also to distortions resulting from our joint propagation procedure. As can be seen in Fig. 3, the propagated label (three frames away from the ground truth) distorts along the moving car's boundary and the pole. Further, we can see how much the model is struggling with these pixels by visualizing the model's entropy over the class label . As the high entropy would suggest, the border pixel confusion contributes to a large amount of the training loss. In our experiments, we show that by relaxing the boundary labels, our training is more robust to accumulated propagation artifacts, allowing us to benefit from longer-range training data propagation.

Table 2: Comparison between (1) label propagation (LP) and joint propagation (JP); (2) video prediction (VPred) and video reconstruction (VRec). Using the proposed video reconstruction and joint propagation techniques, we improve over the baseline by 1.08% mIoU (79.46% → 80.54%).

	0	±1	±2	±3	±4	±5
VPred + LP	79.46	79.79	79.77	79.71	79.55	79.42
VPred + JP	79.46	80.26	80.21	80.23	80.11	80.04
VRec + JP	79.46	**80.54**	80.47	80.51	80.34	80.18

4. Experiments

In this section, we evaluate our proposed method on three widely adopted semantic segmentation datasets, including Cityscapes [15], CamVid [7] and KITTI [1]. For all three datasets, we use the standard mean Intersection over Union (mIoU) metric to report segmentation accuracy.

4.1. Implementation Details

For the video prediction/reconstruction models, the training details are described in the supplementary materials. For semantic segmentation, we use an SGD optimizer and employ a polynomial learning rate policy [27, 13], where the initial learning rate is multiplied by $(1 - \frac{epoch}{max\_epoch})^{power}$. We set the initial learning rate to 0.002 and power to 1.0. Momentum and weight decay are set to 0.9 and 0.0001 respectively. We use synchronized batch normalization [44, 43] with a batch size of 16 distributed over 8 V100 GPUs. The number of training epochs is set to 180 for Cityscapes, 120 for Camvid and 90 for KITTI. The crop size is 800 for Cityscapes, 640 for Camvid and 368 for KITTI due to different image resolutions. For data augmentation, we randomly scale the input images (from 0.5 to 2.0), and apply horizontal flipping, Gaussian blur and color jittering during training. Our network architecture is based on DeepLabV3Plus [14]. For the network backbone, we use ResNeXt50 [39] for the ablation studies, and WideResNet38 [38] for the final test-submissions. In addition, we adopt the following two effective strategies.

Mapillary Pre-Training Instead of using ImageNet pre-trained weights for model initialization, we pre-train our model on Mapillary Vistas [32]. This dataset contains street-level scenes annotated for autonomous driving, which is close to Cityscapes. Furthermore, it has a larger training set (*i.e.*, 18K images) and more classes (*i.e.*, 65 classes).

Class Uniform Sampling We introduce a data sampling strategy similar to [10]. The idea is to make sure that all classes are approximately uniformly chosen during training. We first record the centroid of areas containing the class of interest. During training, we take half of the samples from the standard randomly cropped images and the other half from the centroids to make sure the training crops for all classes are approximately uniform per epoch. In this case, we are actually oversampling the underrepresented cate-
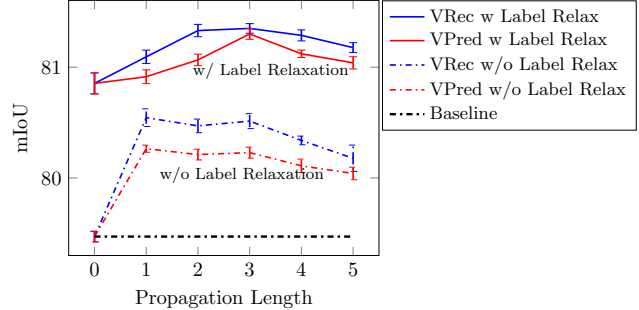

Figure 4: Boundary label relaxation leads to higher mIoU at all propagation lengths. The longer propagation, the bigger the gap between the solid (with label relaxation) and dashed (without relaxation) lines. The black dashed line represents our baseline (79.46%). x-axis equal to 0 indicates no augmented samples are used. For each experiment, we perform three runs and report the mean and sample standard deviation as the error bar [8].

gories. For Cityscapes, we also utilize coarse annotations based on class uniform sampling. We compute the class centroids for all 20K samples, but we can choose which data to use. For example, classes such as fence, rider, train are underrepresented. Hence, we only augment these classes by providing extra coarse samples to balance the training.

4.2. Cityscapes

Cityscapes is a challenging dataset containing high quality pixel-level annotations for 5000 images. The standard dataset split is 2975, 500, and 1525 for the training, validation, and test sets respectively. There are also 20K coarsely annotated images. Cityscapes defines 19 semantic labels containing both objects and stuff, and a void class for do-not-care regions. We perform several ablation studies below on the validation set to justify our framework design.

Stronger Baseline First, we demonstrate the effectiveness of Mapillary pre-training and class uniform sampling. As shown in Table 1, Mapillary pre-training is highly beneficial and improves mIoU by 1.72% over the baseline (76.60% → 78.32%). This makes sense because the Mapillary Vista dataset is close to Cityscape in terms of domain similarity, and thus provides better initialization than ImageNet. We also show that class uniform sampling is an effective data sampling strategy to handle class imbalance problems. It brings an additional 1.14% improvement (78.32% → 79.46%). We use this recipe as our baseline.

Label Propagation versus Joint Propagation Next, we show the advantage of our proposed joint propagation over label propagation. For both settings, we use the motion vectors predicted by the video prediction model to perform propagation. The comparison results are shown

in Table 2. Column 0 in Table 2 indicates the baseline ground-truth-only training (no augmentation with synthesized data). Columns 1 to 5 indicate augmentation with sythesized data from timesteps $\pm k$, not including intermediate sythesized data from timesteps $< |k|$. For example, ± 3 indicates we are using $+3, -3$ and the ground truth samples, but not ± 1 and ± 2. Note that we also tried the accumulated case, where ± 1 and ± 2 is included in the training set. However, we observed a slight performance drop. We suspect this is because the cumulative case significantly decreases the probability of sampling a hand-annotated training example within each epoch, ultimately placing too much weight on the synthesized ones and their imperfections. Comparisons between the non-accumulated and accumulated cases can be found in the supplementary materials.

As we can see in Table 2 (top two rows), joint propagation works better than label propagation at all propagation lengths. Both achieve highest mIoU for ± 1, which is basically using information from just the previous and next frames. Joint propagation improves by 0.8% mIoU over the baseline ($79.46\% \rightarrow 80.26\%$), while label propagation only improves by 0.33% ($79.46\% \rightarrow 79.79\%$). This clearly demonstrates the usefulness of joint propagation. We believe this is because label noise from mis-alignment is outweighed by additional dataset diversity obtained from the augmented training samples. Hence, we adopt joint propagation in subsequent experiments.

Video Prediction versus Video Reconstruction Recall from Sec. 3.1 that we have two methods for learning the motion vectors to generate new training samples through propagation: video prediction and video reconstruction. We experiment with both models in Table 2.

As shown in Table 2 (bottom two rows), video reconstruction works better than video prediction at all propagation lengths, which agrees with our expectations. We also find that ± 1 achieves the best result. Starting from ± 4, the model accuracy starts to drop. This indicates that the quality of the augmented samples becomes lower as we propagate further. Compared to the baseline, we obtain an absolute improvement of 1.08% ($79.46\% \rightarrow 80.54\%$). Hence, we use the motion vectors produced by the video reconstruction model in the following experiments.

Effectiveness of Boundary Label Relaxation Theoretically, we can propagate the labels in an auto-regressive manner for as long as we want. The longer the propagation, the more diverse information we will get. However, due to abrupt scene changes and propagation artifacts, longer propagation will generate low quality labels as shown in Fig. 2. Here, we will demonstrate how the proposed boundary label relaxation technique can help to train a better model by utilizing longer propagated samples.
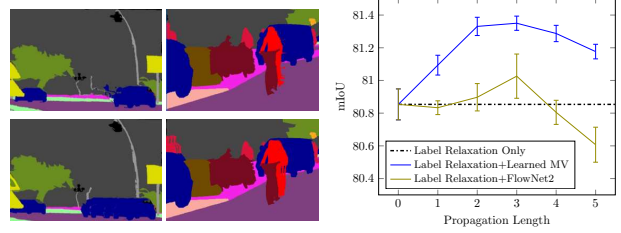
(a) Top: MVs; Bottom: Flow   (b) Propagation length performance

Figure 5: Our learned motion vectors from video reconstruction are better than optical flow (FlowNet2). 5a Qualitative result. The learned motion vectors are better in terms of occlusion handling. 5b Quantitative result. The learned motion vectors are better at all propagation lengths in terms of mIoU.

We use boundary label relaxation on datasets created by video prediction (red) and video reconstruction (blue) in Fig. 4. As we can see, adopting boundary label relaxation leads to higher mIoU at all propagation lengths for both models. Take the video reconstruction model for example. Without label relaxation (dashed lines), the best performance is achieved at ± 1. After incorporating relaxation (solid lines), the best performance is achieved at ± 3 with an improvement of 0.81% mIoU ($80.54\% \rightarrow 81.35\%$). The gap between the solid and dashed lines becomes larger as we propagate longer. The same trend can be observed for the video prediction models. This demonstrates that our boundary label relaxation is effective at handling border artifacts. It helps our model obtain more diverse information from ± 3, and at the same time, reduces the impact of label noise brought by long propagation. Hence, we use boundary label relaxation for the rest of the experiments.

Note that even for no propagation (x-axis equal to 0) in Fig. 4, boundary label relaxation improves performance by a large margin ($79.46\% \rightarrow 80.85\%$). This indicates that our boundary label relaxation is versatile. Its use is not limited to reducing distortion artifacts in label propagation, but it can also be used in normal image segmentation tasks to handle ambiguous boundary labels.

Learned Motion Vectors versus Optical Flow Here, we perform a comparison between the learned motion vectors from the video reconstruction model and optical flow, to show why optical flow is not preferred. For optical flow, we use the state-of-the-art CNN flow estimator FlowNet2 [20] because it can generate sharp object boundaries and generalize well to both small and large motions.

First, we show a qualitative comparison between the learned motion vectors and the FlowNet2 optical flow. As we can see in Fig. 5a, FlowNet2 suffers from serious doubling effects caused by occlusion. For example, the dragging car (left) and the doubling rider (right). In contrast, our learned motion vectors can handle occlusion quite well. The propagated labels have only minor artifacts along the object

Table 3: Per-class mIoU results on Cityscapes. Top: our ablation improvements on the validation set. Bottom: comparison with top-performing models on the test set.

| Method	split	road	swalk	build.	wall	fence	pole	tlight	tsign	veg.	terrain	sky	person	rider	car	truck	bus	train	mcycle	bicycle	mIoU
Baseline	val	98.4	86.5	93.0	57.4	65.5	66.7	70.6	78.9	92.7	65.0	95.3	80.8	60.9	95.3	87.9	91.0	84.3	65.8	76.2	79.5
+ VRec with JP	val	98.0	86.5	94.7	47.6	67.1	69.6	71.8	80.4	92.2	58.4	95.6	88.3	71.1	95.6	76.8	84.7	90.3	79.6	80.3	80.5
+ Label Relaxation	val	98.5	87.4	93.5	64.2	66.1	69.3	74.2	81.5	92.9	64.6	95.6	83.5	66.5	95.7	87.7	91.9	85.7	70.1	78.8	81.4
ResNet38 [38]	test	98.7	86.9	93.3	60.4	62.9	67.6	75.0	78.7	93.7	73.7	95.5	86.8	71.1	96.1	75.2	87.6	81.9	69.8	76.7	80.6
PSPNet [44]	test	98.7	86.9	93.5	58.4	63.7	67.7	76.1	80.5	93.6	72.2	95.3	86.8	71.9	96.2	77.7	91.5	83.6	70.8	77.5	81.2
InPlaceABN [10]	test	98.4	85.0	93.6	61.7	63.9	67.7	77.4	80.8	93.7	71.9	95.6	86.7	72.8	95.7	79.9	93.1	89.7	72.6	78.2	82.0
DeepLabV3+ [14]	test	98.7	87.0	93.9	59.5	63.7	71.4	78.2	82.2	94.0	73.0	95.8	88.0	73.0	96.4	78.0	90.9	83.9	73.8	78.9	82.1
DRN-CRL [46]	test	98.8	87.7	94.0	**65.1**	64.2	70.1	77.4	81.6	93.9	73.5	95.8	88.0	74.9	96.5	**80.8**	92.1	88.5	72.1	78.8	82.8
Ours	test	**98.8**	**87.8**	**94.2**	64.1	**65.0**	**72.4**	**79.0**	**82.8**	**94.2**	**74.0**	**96.1**	**88.2**	75.4	**96.5**	78.8	**94.0**	**91.6**	**73.8**	**79.0**	**83.5**
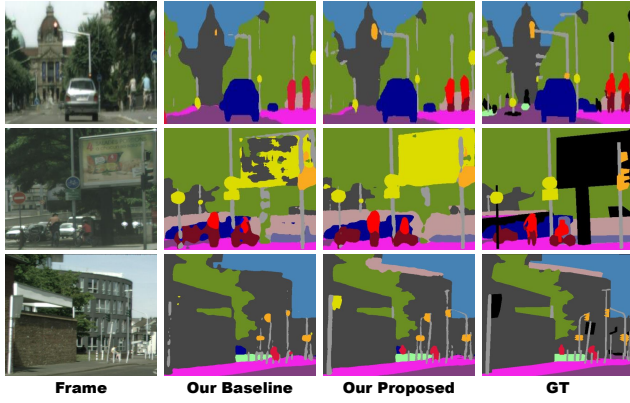
Figure 6: Visual comparisons on Cityscapes. The images are cropped for better visualization. We demonstrate our proposed techniques lead to more accurate segmentation than our baseline. Especially for thin and rare classes, like street light and bicycle (row 1), signs (row 2), person and poles (row 3). Our observation corresponds well to the class mIoU improvements in Table 3.
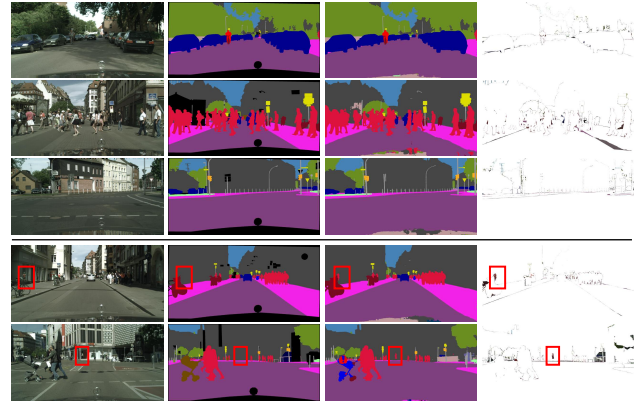

Figure 7: Visual examples on Cityscapes. From left to right: image, GT, prediction and their differences. We demonstrate that our model can handle situations with multiple cars (row 1), dense crowds (row 2) and thin objects (row 3). The bottom two rows show failure cases. We mis-classify a reflection in the mirror (row 4) and a model inside the building (row 5) as person (red boxes).

borders which can be remedied by label relaxation. Next, we show their quantitative comparison. As we can see in Fig. 5b, the learned motion vectors (blue) perform significantly better than FlowNet2 (red) at all propagation lengths. As we propagate longer, the gap between them becomes larger, which indicates the low quality of the FlowNet2 augmented samples. Note that when the propagation length is ± 1, ± 4 and ± 5, the performance of FlowNet2 is even lower than the baseline.

Comparison to State-of-the-Art As shown in Table 3 top, our proposed video reconstruction-based data synthesis together with joint propagation improves by 1.0% mIoU over the baseline. Incorporating label relaxation brings another 0.9% mIoU improvement. We observe that the largest improvements come from small/thin object classes, such as pole, street light/sign, person, rider and bicycle. This can be explained by the fact that our augmented samples result in more variation for these classes and helps with model generalization. We show several visual comparisons in Fig. 6.

For test submission, we train our model using the best recipe suggested above, and replace the network backbone with WideResNet38 [38]. We adopt a multi-scale strategy

[44, 14] to perform inference on multi-scaled (0.5, 1.0 and 2.0), left-right flipped and overlapping-tiled images, and compute the final class probabilities after averaging logits per inference. More details can be found in the supplementary materials. As shown in Table 3 bottom, we achieve an mIoU of 83.5%, outperforming all prior methods. We get the highest IoU on 18 out of the 20 classes except for wall and truck. In addition, we show several visual examples in Fig. 7. We demonstrate that our model can handle situations with multiple cars (row 1), dense crowds (row 2) and thin objects (row 3). We also show two interesting failure cases in Fig. 7. Our model mis-classifies a reflection in the mirror (row 4) and a model inside the building (row 5) as person (red boxes). However, in terms of appearance without reasoning about context, our predictions are correct. More visual examples can be found in the supplementary materials.

4.3. CamVid

CamVid is one of the first datasets focusing on semantic segmentation for driving scenarios. It is composed of 701 densely annotated images with size 720×960 from five video sequences. We follow the standard protocol proposed in [3] to split the dataset into 367 training, 101 val-

Figure 8: Visual comparison between our results and those of the winning entry [10] of ROB challenge 2018 on KITTI. From left to right: image, prediction from [10] and ours. Boxes indicate regions in which we perform better than [10]. Our model can predict semantic objects as a whole (bus), detect thin objects (poles and person) and distinguish confusing classes (sidewalk and road, building and sky).

Table 4: Results on the CamVid test set. Pre-train indicates the source dataset on which the model is trained.

Method	Pre-train	Encoder	mIoU (%)
SegNet [3]	ImageNet	VGG16	60.1
RTA [19]	ImageNet	VGG16	62.5
Dilate8 [42]	ImageNet	Dilate	65.3
BiSeNet [41]	ImageNet	ResNet18	68.7
PSPNet [44]	ImageNet	ResNet50	69.1
DenseDecoder [6]	ImageNet	ResNeXt101	70.9
VideoGCRF [11]	Cityscapes	ResNet101	75.2
Ours (baseline)	Cityscapes	WideResNet38	79.8
Ours	Cityscapes	WideResNet38	**81.7**

Table 5: Results on KITTI test set.

Method	IoU class	iIoU class	IoU category	iIoU category
APMoE_seg [23]	47.96	17.86	78.11	49.17
SegStereo [40]	59.10	28.00	81.31	60.26
AHiSS [30]	61.24	26.94	81.54	53.42
LDN2 [24]	63.51	28.31	85.34	59.07
MapillaryAI [10]	69.56	43.17	86.52	68.89
Ours	**72.83**	**48.68**	**88.99**	**75.26**

idation and 233 test images. A total of 32 classes are provided. However, most literature only focuses on 11 due to the rare occurrence of the remaining classes. To create the augmented samples, we directly use the video reconstruction model trained on Cityscapes without fine tuning on CamVid. The training strategy is similar to Cityscapes. We compare our method to recent literature in Table 4. For fair comparison, we only report single-scale evaluation scores. As can be seen in Table 4, we achieve an mIoU of 81.7%, outperforming all prior methods by a large margin. Furthermore, our multi-scale evaluation score is 82.9%. Per-class breakdown can be seen in the supplementary materials.

One may argue that our encoder is more powerful than prior methods. To demonstrate the effectiveness of our proposed techniques, we perform training under the same settings without using the augmented samples and boundary label relaxation. The performance of this configuration on the test set is 79.8%, a significant IoU drop of 1.9%.

4.4. KITTI

The KITTI Vision Benchmark Suite [17] was introduced in 2012 but updated with semantic segmentation ground truth [1] in 2018. The data format and metrics conform with Cityscapes, but with a different image resolution of 375×1242. The dataset consists of 200 training and 200 test images. Since the dataset is quite small, we perform 10-split cross validation fine-tuning on the 200 training images. Eventually, we determine the best model in terms of mIoU on the whole training set because KITTI only allows

one submission for each algorithm. For 200 test images, we run multi-scale inference by averaging over 3 scales (1.5, 2.0 and 2.5). We compare our method to recent literature in Table 5. We achieve significantly better performance than prior methods on all four evaluation metrics. In terms of mIoU, we outperform previous state-of-the-art [10] by 3.3%. Note that [10] is the winning entry to Robust Vision Challenge 2018. We show two visual comparisons between ours and [10] in Fig. 8.

5. Conclusion

We propose an effective video prediction-based data synthesis method to scale up training sets for semantic segmentation. We also introduce a joint propagation strategy to alleviate mis-alignments in synthesized samples. Furthermore, we present a novel boundary relaxation technique to mitigate label noise. The label relaxation strategy can also be used for human annotated labels and not just synthesized labels. We achieve state-of-the-art mIoUs of 83.5% on Cityscapes, 82.9% on CamVid, and 72.8% on KITTI. The superior performance demonstrates the effectiveness of our proposed methods. We hope our approach inspires other ways to perform data augmentation, such as GANs [26], to enable cheap dataset collection and achieve improved accuracy in target tasks. For future work, we would like to explore soft label relaxation using the learned kernels in [34] for better uncertainty reasoning. Our state-of-the-art implementation, will be made publicly available.

References

[1] Hassan Alhaija, Siva Mustikovela, Lars Mescheder, Andreas Geiger, and Carsten Rother. Augmented Reality Meets Computer Vision: Efficient Data Generation for Urban Driving Scenes. *International Journal of Computer Vision (IJCV)*, 2018.

[2] V. Badrinarayanan, F. Galasso, and R. Cipolla. Label Propagation in Video Sequences. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[3] 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 (PAMI)*, 2017.

[4] Gedas Bertasius, Jianbo Shi, and Lorenzo Torresani. Semantic Segmentation with Boundary Neural Fields. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[5] Gedas Bertasius, Lorenzo Torresani, Stella X. Yu, and Jianbo Shi. Convolutional Random Walk Networks for Semantic Image Segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[6] Piotr Bilinski and Victor Prisacariu. Dense Decoder Shortcut Connections for Single-Pass Semantic Segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[7] Gabriel J. Brostow, Jamie Shotton, Julien Fauqueur, and Roberto Cipolla. Segmentation and Recognition Using Structure from Motion Point Clouds. In *European Conference on Computer Vision (ECCV)*, 2008.

[8] George W Brown. Standard Deviation, Standard Error: Which 'Standard' Should We Use? *American Journal of Diseases of Children*, 1982.

[9] Ignas Budvytis, Patrick Sauer, Thomas Roddick, Kesar Breen, and Roberto Cipolla. Large Scale Labelled Video Data Augmentation for Semantic Segmentation in Driving Scenarios. In *International Conference on Computer Vision (ICCV) Workshop*, 2017.

[10] Samuel Rota Bul, Lorenzo Porzi, and Peter Kontschieder. In-Place Activated BatchNorm for Memory-Optimized Training of DNNs. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[11] Siddhartha Chandra, Camille Couprie, and Iasonas Kokkinos. Deep Spatio-Temporal Random Fields for Efficient Video Segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[13] 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 (PAMI)*, 2018.

[14] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. In *European Conference on Computer Vision (ECCV)*, 2018.

[15] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[16] Raghudeep Gadde, Varun Jampani, and Peter V. Gehler. Semantic Video CNNs Through Representation Warping. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[17] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are We Ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[18] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei Efros, and Trevor Darrell. CyCADA: Cycle-Consistent Adversarial Domain Adaptation. In *International Conference on Machine Learning (ICML)*, 2018.

[19] Po-Yu Huang, Wan-Ting Hsu, Chun-Yueh Chiu, Ting-Fan Wu, and Min Sun. Efficient Uncertainty Estimation for Semantic Segmentation in Videos. In *European Conference on Computer Vision (ECCV)*, 2018.

[20] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox. FlowNet 2.0: Evolution of Optical Flow Estimation with Deep Networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[21] Tsung-Wei Ke, Jyh-Jing Hwang, Ziwei Liu, and Stella X. Yu. Adaptive Affinity Fields for Semantic Segmentation. In *European Conference on Computer Vision (ECCV)*, 2018.

[22] Alex Kendall and Yarin Gal. What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? In *Conference on Neural Information Processing Systems (NIPS)*, 2017.

[23] Shu Kong and Charless Fowlkes. Pixel-wise Attentional Gating for Parsimonious Pixel Labeling. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2019.

[24] Ivan Kreo, Josip Krapac, and Sinia egvi. Ladder-style DenseNets for Semantic Segmentation of Large Natural Images. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[25] Xiaoxiao Li, Ziwei Liu, Ping Luo, Chen Change Loy, and Xiaoou Tang. Not All Pixels Are Equal: Difficulty-aware Semantic Segmentation via Deep Layer Cascade. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[26] Shuangting Liu, Jiaqi Zhang, Yuxin Chen, Yifan Liu, Zengchang Qin, and Tao Wan. Pixel Level Data Augmentation for Semantic Image Segmentation using Generative Adversarial Networks. *arXiv preprint arXiv:1811.00174*, 2018.

[27] Wei Liu, Andrew Rabinovich, and Alexander C. Berg. ParseNet: Looking Wider to See Better. In *International Conference on Learning Representations (ICLR)*, 2016.

[28] Pauline Luc, Natalia Neverova, Camille Couprie, Jakob Verbeek, and Yann LeCun. Predicting Deeper into the Future of Semantic Segmentation. In *International Conference on Computer Vision (ICCV)*, 2017.

[29] Dimitrios Marmanis, Konrad Schindler, Jan Dirk Wegner, Silvano Galliani, Mihai Datcu, and Uwe Stilla. Classification With an Edge: Improving Semantic Image Segmentation with Boundary Detection. *ISPRS Journal of Photogrammetry and Remote Sensing*, 2018.

[30] Panagiotis Meletis and Gijs Dubbelman. Training of Convolutional Networks on Multiple Heterogeneous Datasets for Street Scene Semantic Segmentation. *arXiv preprint arXiv:1803.05675*, 2018.

[31] Siva Karthik Mustikovela, Michael Ying Yang, and Carsten Rother. Can Ground Truth Label Propagation from Video help Semantic Segmentation? In *European Conference on Computer Vision (ECCV) Workshop*, 2016.

[32] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bul, and Peter Kontschieder. The Mapillary Vistas Dataset for Semantic Understanding of Street Scenes. In *International Conference on Computer Vision (ICCV)*, 2017.

[33] David Nilsson and Cristian Sminchisescu. Semantic Video Segmentation by Gated Recurrent Flow Propagation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[34] Fitsum A. Reda, Guilin Liu, Kevin J. Shih, Robert Kirby, Jon Barker, David Tarjan, Andrew Tao, and Bryan Catanzaro. SDC-Net: Video Prediction using Spatially-Displaced Convolution. In *European Conference on Computer Vision (ECCV)*, 2018.

[35] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez. The SYNTHIA Dataset: A Large Collection of Synthetic Images for Semantic Segmentation of Urban Scenes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[36] Swami Sankaranarayanan, Yogesh Balaji abd Arpit Jain, Ser Nam Lim, and Rama Chellappa. Learning from Synthetic Data: Addressing Domain Shift for Semantic Segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[37] Remi Vieux, Jenny Benois-Pineau, Jean-Philippe Domenger, and Achille Braquelaire. Segmentation-based Multi-class Semantic Object Detection. *Multimedia Tools and Applications*, 2012.

[38] Zifeng Wu, Chunhua Shen, and Anton van den Hengel. Wider or Deeper: Revisiting the ResNet Model for Visual Recognition. *arXiv:1611.10080*, 2016.

[39] Saining Xie, Ross Girshick, Piotr Dollr, Zhuowen Tu, and Kaiming He. Aggregated Residual Transformations for Deep Neural Networks. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[40] Guorun Yang, Hengshuang Zhao, Jianping Shi, Zhidong Deng, and Jiaya Jia. SegStereo: Exploiting Semantic Information for Disparity Estimation. In *European Conference on Computer Vision (ECCV)*, 2018.

[41] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. BiSeNet: Bilateral Segmentation Network for Real-time Semantic Segmentation. In *European Conference on Computer Vision (ECCV)*, 2018.

[42] Fisher Yu and Vladlen Koltun. Multi-Scale Context Aggregation by Dilated Convolutions. In *International Conference on Learning Representations (ICLR)*, 2016.

[43] Hang Zhang, Kristin Dana, Jianping Shi, Zhongyue Zhang, Xiaogang Wang, Ambrish Tyagi, and Amit Agrawal. Context Encoding for Semantic Segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[44] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid Scene Parsing Network. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[45] Xinge Zhu, Hui Zhou, Ceyuan Yang, Jianping Shi, and Dahua Lin. Penalizing Top Performers: Conservative Loss for Semantic Segmentation Adaptation. In *European Conference on Computer Vision (ECCV)*, 2018.

[46] Yueqing Zhuang, Fan Yang, Li Tao, Cong Ma, Ziwei Zhang, Yuan Li, Huizhu Jia, Xiaodong Xie, and Wen Gao. Dense Relation Network: Learning Consistent and Context-Aware Representation For Semantic Image Segmentation. In *IEEE International Conference on Image Processing (ICIP)*, 2018.

[47] Aleksandar Zlateski, Ronnachai Jaroensri, Prafull Sharma, and Frdo Durand. On the Importance of Label Quality for Semantic Segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.