Supplementary materials: Efficient Semantic Video Segmentation with Per-frame Inference

Anonymous ECCV submission

Paper ID 1094

S1 Details of distillation mechanism

S1.1 Single frame distillation

Following Liu et.al. [5], we employ pixel-wise distillation and pair-wise distillation for each single frame. For the pixel-wise distillation, we use the class probabilities Q produced from the cumbersome model as soft targets for training the compact network.

The loss function based on the Kullback-Leibler divergence is given as follows,

$$\ell_{pi} = \frac{1}{N}\sum_{i=1}^{N} \mathrm{KL}(\mathbf{q}_i^s \| \mathbf{q}_i^t), \tag{1}$$

where $\mathbf{q}_i$ represent the class probabilities of the ith pixel of the segmentation map and N is the number of the pixels.

The pair-wise distillation is built on the self-similarity map $\mathbf{A}$ as described in multi-frame dependency. We adopt the squared difference to formulate the pair-wise similarity distillation loss,

$$\ell_{pa} = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{i=j}^{N}(a_{ij}^s - a_{ij}^t)^2. \tag{2}$$

The similarity between two pixels is simply computed from the features $\mathbf{f}_i$ and $\mathbf{f}_j$ as $a_{ij} = \mathbf{f}_i^\top \mathbf{f}_j / (\|\mathbf{f}_i\|_2 \|\mathbf{f}_j\|_2)$. The final loss for sing frame distillation is $\ell_{SF}^t = \ell_{pi} + \ell_{pa}$

S1.2 Multi-frame distillation

We employ a ConvLSTM [6] unit to capture the correlations among all frames in a video sequence. The input sequence is consists of the self-similarity maps of the feature map for each frame, $\mathcal{A} = \{\dots \mathbf{A}_{\mathbf{F}_{t-1},\mathbf{F}_{t-1}}, \mathbf{A}_{\mathbf{F}_t,\mathbf{F}_t}, \mathbf{A}_{\mathbf{F}_t+1,\mathbf{F}_t+1} \dots\}$. For each time step,the key equations are shown in below:

$$\begin{aligned}
\mathbf{i}_t =& \sigma(\mathbf{W}_{ai} * \mathbf{A}_{\mathbf{F}_t,\mathbf{F}_t} + \mathbf{W}_{hi} * \mathbf{H}_{t-1} + \mathbf{W}_{ei} \circ \mathbf{E}_{t-1} + \mathbf{b}_i) \\
\mathbf{f}_t =& \sigma(\mathbf{W}_{af} * \mathbf{A}_{\mathbf{F}_t,\mathbf{F}_t} + \mathbf{W}_{hf} * \mathbf{H}_{t-1} + \mathbf{W}_{ef} \circ \mathbf{E}_{t-1} + \mathbf{b}_f) \\
\mathbf{E}_t =& \mathbf{f}_t \circ \mathbf{E}_{t-1} + \\
& \mathbf{i}_t \circ \tanh(\mathbf{W}_{ae} * \mathbf{A}_{\mathbf{F}_t,\mathbf{F}_t} + \mathbf{W}_{he} * \mathbf{H}_{t-1} + \mathbf{b}_e) \\
\mathbf{o}_t =& \sigma(\mathbf{W}_{ao} * \mathbf{A}_{\mathbf{F}_t,\mathbf{F}_t} + \mathbf{W}_{ho} * \mathbf{H}_{t-1} + \mathbf{W}_{eo} \circ \mathbf{E}_t + \mathbf{b}_o) \\
\mathbf{H}_t =& \mathbf{o}_t \circ \tanh(\mathbf{E}_t)
\end{aligned} \tag{3}$$

where '$\circ$' denotes the Hadamard product,'$*$' denotes the convolution operator, 'σ' is the sigmoid activation function and the activation of input gate i_t controls whether the new input of this time step will be engaged in the memory cell. $\mathbf{f}_t$ controls how much to keep from the past cell status $\mathbf{E}_{t-1}$. o_t decides the propagation from $\mathbf{E}_t$ to the hidden state $\mathbf{H}_t$. $\mathbf{W}$ and $\mathbf{b}$ represent the trainable parameters in the ConvLSTM unit. We employ the memory state of the final time step $\mathbf{E}_T$ as the distillation item, which contains multi-frame dependency. We align the multi-frame dependency from the teacher net and the student net to enhance the performance of the student net. According to [6], the state of ConvLSTM unit can be viewed as the hidden representations of moving objects, therefore the multi-frame dependency distillation can help to transfer the temporal consistency from teacher net to the student net.

S2 Training and evaluation details

S2.1 Dataset

Cityscapes [2] is collected for urban scene understanding and contains 30-frame snippets of the street scene with 17 frames per second. The dataset contains $5,000$ high quality pixel-level finely annotated images at 20^{th} frame in each snippets, which are divided into $2,975$, 500, $1,525$ images for training, validation and testing. The CamVid dataset [1] is an automotive dataset. It contains five different videos, which has ground truth labels every 30 frames. Three train videos contain 367 frames, while two test videos contain 233 frames.

S2.2 Training and inference.

On Cityscapes, the segmentation networks in this paper are trained by mini-batch stochastic gradient descent (SGD) for 200 epochs. We sample 8 training triplets for each mini-batch. The learning rate is initialized as 0.01 and is multiplied by $(1 - \frac{iter}{max-iter})^{0.9}$. We randomly cut the images into 769×769 as the training input. Normal data augmentation methods are applied during training, such as random scaling (from 0.5 to 2.1) and random flipping. On Camvid, we use a crop size of 640×640. We use the official implementation of PSPNet in Pytorch[7] and train all the network with 4 cards of Tesla Volta 100.

S2.3 Details of the evaluating temporal consistency

We follow [4] to measure the temporal stability of a video based on the flow warping error between two frames. Different from [4], we use the mIoU score instead of the mean square error to evaluate the semantic segmentation results

$$E_{warp}(\mathbf{Q}_{t-1}, \mathbf{Q}_t) = \frac{\mathbf{Q}_t \cap \hat{\mathbf{Q}}_{t-1}}{\mathbf{Q}_t \cup \hat{\mathbf{Q}}_{t-1}} \tag{4}$$

where $\mathbf{Q}_t$ represents for the predict segmentation map of frame t and $\hat{\mathbf{Q}}_{t-1}$ represents for the warped segmentation map from frame $t-1$ to frame t. We calculate a statistical average warp IoU on each sequence, and using an average mean on the validation set to evaluate the temporal stability:

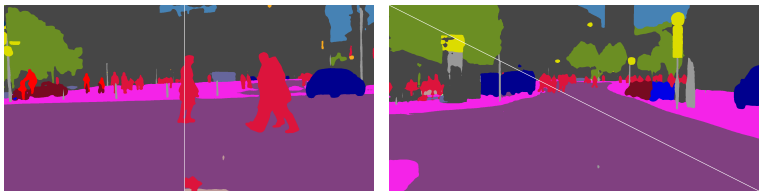
$$E_{warp} = \frac{1}{N}\sum_{i=1}^{N}\frac{\mathcal{Q}^i \cap \hat{\mathcal{Q}}^i}{\mathcal{Q}^i \cup \hat{\mathcal{Q}}^i} \tag{5}$$

where $\mathcal{Q} = \{\mathbf{Q}_2, \ldots, \mathbf{Q}_T\}$ and $\hat{\mathcal{Q}} = \{\hat{\mathbf{Q}}_1, \ldots, \hat{\mathbf{Q}}_{T-1}\}$. T is the total frames of the sequence and N is the number of the sequence. On Cityscapes [2], we random sample 100 video sequence from the validation set, which contains 3000 images to evaluate the temporal stability. On Camvid [1], we evaluate the temporal stability of the video sequence 'seq05' from the test set.

S3 Description of videos and visualization results

We include three videos in the supplementary materials, named 'demo_seq00.mp4', 'val.mp4', and 'Baseline_SKD_Accel_Ours.mp4' to show the improvement of the temporal consistency. Sampled frames are shown in Figure 1, Figure 2 and Figure 3. From the video, we can see that the proposed method can improve the accuracy and the temporal consistency compared with the baseline models. We can also observe that in some situations, both our method and the baseline method will produce inconsistent predictions. From the comparison with Accel [3] and SKD [5], we can see that keyframe based methods suffer from the jitters while GAN based distillation methods will produce inconsistent results. Our method can produce stable and smooth sequence with high accuracy.

Figure 4 shows some segmentation results on CamVid dataset. We can observe that the proposed method outperforms the baseline method in the red region.



(a) An example in val.mp4 (b) An example in demo_seq00.mp4

Fig. 1: We use a white line to divide the whole scene into two parts. 'val.mp4' is the sampled from the validation set we use to evaluate the temporal consistency. In 'val.mp4', our method is shown on the left of the line while the baseline method is on the right. 'demo_seq00.mp4' is the prediction results on the provided demo video 'sequence00' in the Cityscapes dataset, and our method is above the line while the baseline is below the line.

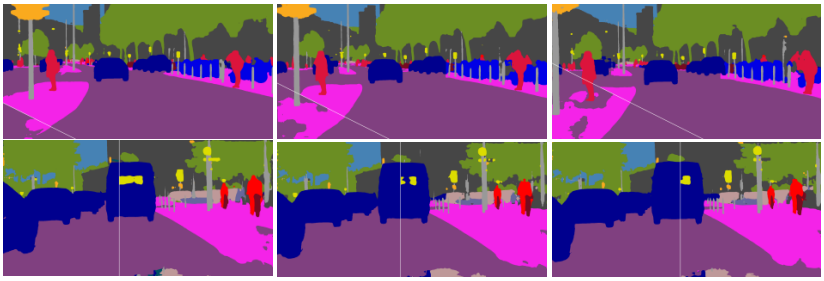


Fig. 2: Consecutive frames in two videos. **First row**:'demo_seq00.mp4'. Our results are on the top right. **Second row**:'val.mp4'. Our results are on the left. More results can be found in the supplementary videos.

S4 Results on each class

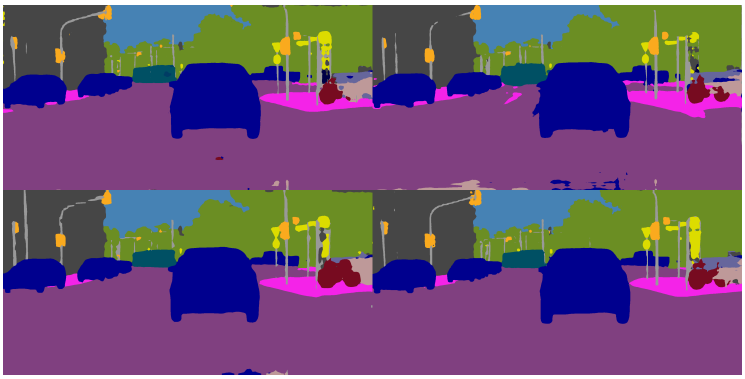
We compare our method with the baseline methods of PSPNet18 in terms of the accuracy and temporal consistency of each class on Cityscapes. The results are shown in Table 1. For the moving objects with regular structures, e.g. 'train', 'bus', both segmentation accuracy and temporal consistency are improved significantly. For the 'road', 'sidewalk' and 'terrain', the temporal consistency are also improved although the accuracy only have limited improvements.

Table 1: Accuracy (mIoU, %) and temporal consistency (TC, %) for each class on Cityscapes. Baseline: PSPNet18 trained on each frame independently. Ours: PSPNet18 trained with temporal loss and distillation items.

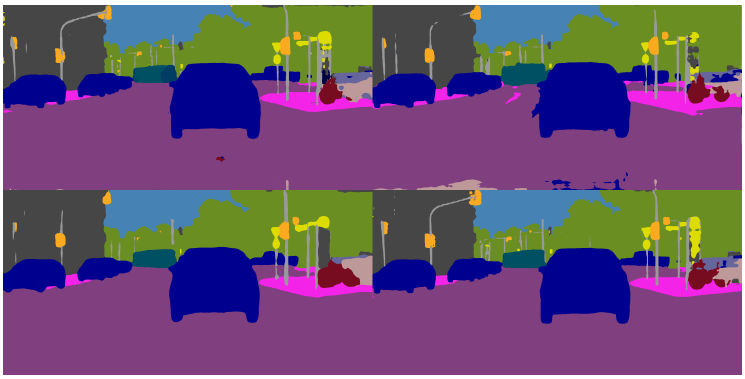
Class Name		road	sidewalk	building	wall	fence	pole	tra. light	tra. sign	vegetation	terrain
mIoU	Baseline	97.0	78.7	90.1	41.8	54.7	50.3	63.6	72.0	90.8	60.0
	Ours	**97.2**	**79.4**	**91.0**	**49.8**	**57.4**	**53.1**	**67.0**	**73.6**	**91.0**	60.0
TC	Baseline	97.2	80.2	91.2	**50.0**	62.1	42.6	47.2	52.6	91.7	72.0
	Ours	**97.7**	**81.4**	**91.6**	49.6	**62.6**	**43.9**	**48.5**	**53.2**	**91.9**	**73.3**
Class Name		sky	person	rider	car	truck	bus	train	motorbike	bicycle	mean
mIoU	Baseline	92.8	75.8	52.7	91.6	61.4	77.1	56.9	46.9	71.8	69.8
	Ours	**93.1**	**77.1**	**57.1**	**92.1**	**65.5**	**82.2**	**73.1**	**55.6**	**72.8**	**73.1**
TC	Baseline	92.8	68.7	28.7	86.4	74.8	78.5	55.5	55.9	73.7	68.5
	Ours	**93.0**	**69.6**	**30.1**	**87.0**	**76.3**	**82.2**	**76.4**	**57.5**	**74.9**	**70.6**

References

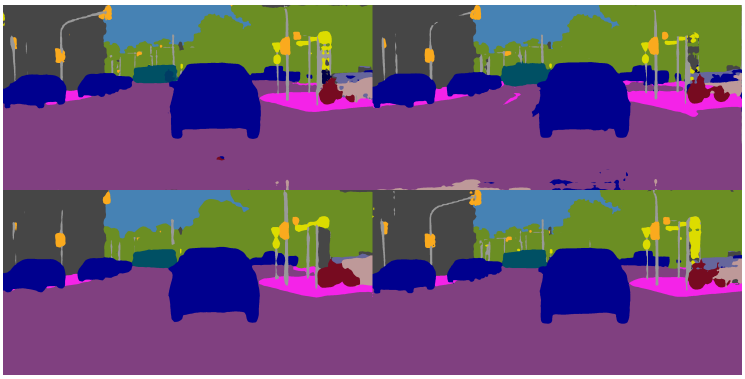
1. Brostow, G.J., Shotton, J., Fauqueur, J., Cipolla, R.: Segmentation and recognition using structure from motion point clouds. In: Proc. Eur. Conf. Comp. Vis. pp. 44–57. Springer (2008)
2. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: Proc. IEEE Conf. Comp. Vis. Patt. Recogn. (2016)



(a) frame k



(b) frame k+1



(c) frame k+2

Fig. 3: Consecutive frames in 'Baseline_SKD_Accel_Ours.mp4'. **Top left**: Baseline.**Top right**: SKD [5]. **Bottom left**: Accel [3]. **Bottom right**: Ours. There are jitters between keyframe and normal frame in the results sequence of Accel. More results can be found in the supplementary videos.

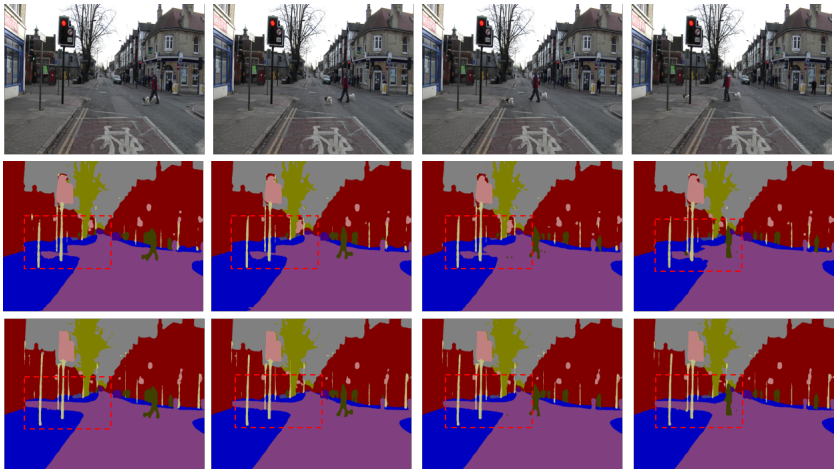


Fig. 4: Consecutive frames in Camvid dataset. **First row**: input frames. **Second row**: MobileNet trained with cross-entropy loss. **Third row**:MobileNet trained with the temporal loss and distillation items. In the baseline method, the region in the red box keep changing while the proposed method can produce similar results on the still stuff.

3. Jain, S., Wang, X., Gonzalez, J.E.: Accel: A corrective fusion network for efficient semantic segmentation on video. In: Proc. IEEE Conf. Comp. Vis. Patt. Recogn. pp. 8866–8875 (2019)
4. Lai, W.S., Huang, J.B., Wang, O., Shechtman, E., Yumer, E., Yang, M.H.: Learning blind video temporal consistency. In: Proc. Eur. Conf. Comp. Vis. pp. 170–185 (2018)
5. Liu, Y., Chen, K., Liu, C., Qin, Z., Luo, Z., Wang, J.: Structured knowledge distillation for semantic segmentation. In: Proc. IEEE Conf. Comp. Vis. Patt. Recogn. pp. 2604–2613 (2019)
6. Shi, X., Chen, Z., Wang, H., Yeung, D.Y., Wong, W.k., Woo, W.c.: Convolutional lstm network: A machine learning approach for precipitation nowcasting. In: Proc. Advances in Neural Inf. Process. Syst. pp. 802–810 (2015)
7. Zhao, H.: Semseg. `https://github.com/hszhao/semseg` (2019)