2020 IEEE International Conference on Robotics and Automation (ICRA)
31 May - 31 August, 2020. Paris, France

Temporal information integration for video semantic segmentation

G. Guarino[1], T. Chateau[1], C. Teulière[1], V. Antoine[2]
[1]Université Clermont Auvergne, CNRS, SIGMA Clermont, Institut Pascal, F-63000 Clermont-Ferrand, France
[2]Université Clermont Auvergne, CNRS, LIMOS, F-63000 Clermont-Ferrand, France.

Abstract— **We present a temporal Bayesian filter for semantic segmentation of a video sequence. Each pixel is a random variable following a discrete probabilistic distribution function representing possible semantic classes. Bayesian filtering consists in two main steps: 1) a prediction model and 2) an observation model (likelihood). We propose to use a data-driven prediction function derived from a dense optical flow between images t and $t+1$ achieved by a deep neural network [1]. Moreover, the observation function uses a semantic segmentation network. The resulting approach is evaluated on the public dataset Cityscapes. We show that using the temporal filtering increases the accuracy of the semantic segmentation.**

I. INTRODUCTION

Semantic segmentation becomes a very popular task in many applications such as posture recognition, face parsing or autonomous driving. It consists in associating a semantic class (for instance road, sidewalk, rider, car, etc.) to each pixel of an image. This information may then be used for trajectory planning or obstacles avoidance. The current state of the art methods use Deep Neural Convolutional Networks (DCNN) [2], [1], [3]. Most of the conventional segmentation networks do not take into account the temporal link between images and process all images from a sequence as independent images. In this paper, we propose to express the semantic segmentation of video-sequences as a filtering problem. Each pixel is a random variable. A recursive Bayesian filter implementation is proposed to allow a continuous integration of the temporal information and to improve segmentation consistency (see Figure 1). The prediction function is achieved by a data-driven model (optical flow network) and the observation function uses a semantic segmentation network. To summarize, the main contributions of this work are:

- a formulation of semantic segmentation of a video sequence as a Bayesian filter,
- a data-driven prediction function using a dense optical flow network,
- an evaluation of the filter on the public dataset Cityscapes.

First, the last advances in the field of semantic segmentation will be detailed. Then, the structure of the proposed Bayesian filter and the modeling of temporal information will be presented. After that, our optical flow model will be described. In this part, a method to detect wrong temporal information (due to objects appearances/disappearances) so that only relevant information is utilized will also be detailed and analyzed. Finally, we will present the evaluation results of the filter on Cityscapes dataset.

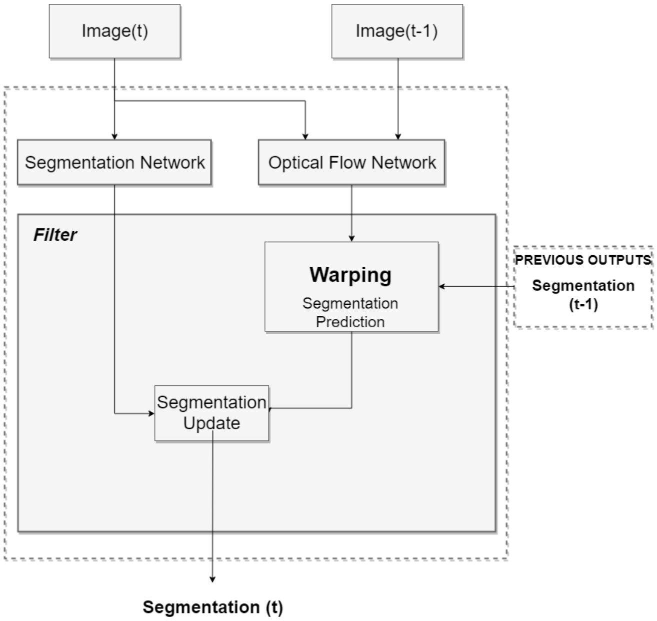

Fig. 1: Structure of the proposed Recursive Bayesian Filter

II. RELATED WORK

This section reports some recent advances in the field of semantic segmentation and more precisely in the field of video semantic segmentation.

Several DCNN models have been proposed for conventional semantic segmentation. Among them, the PSPNet (Pyramid Scene Parsing Network) [1] is based on a multi-scale analysis and information fusion. Therefore, this network allows the understanding of both the general context and the local contexts and details which leads to a better scene understanding. The DeepLab network [3] stands out by the use of atrous convolutions to take into account the widest context. The latest results from high-frequency networks such as BiSeNet [4] show that real-time use of these networks becomes possible with a high accuracy.

In the case of video segmentation, these networks do not take into account the temporal information contained in the image sequences. Each image is processed as if it were independent from others. This is why approaches have been developed to take into account the temporal link and to improve segmentation consistency.

Long Short Term Memory networks (LSTM) enable a transfer of temporal information. They are already used in video classification tasks [5]. In this case the LSTMs are

978-1-7281-7395-5/20/$31.00 ©2020 IEEE

associated with Convolutional Neural Networks (CNN) in the form of hybrid networks where the LSTMs networks take into consideration the previous outputs of the CNNs to create a temporal link. Recurrent networks can also be used for semantic video segmentation [6] [7] [8] but their accuracy is lower than conventional segmentation networks accuracy.

Temporal information are sometimes explicitly utilized through optical flow. In [9], the segmentation algorithm is applied only at regular intervals and the optical flow enables a set of predictions on all images located between two segmented images. The optical flow is used to propagate segmentation on unprocessed images. Since optical flow estimation is faster than the segmentation process, this method can speed up the processing but doesn't improve segmentation. In [10] internal feature maps are propagated during the following image processing with the optical flow estimation. [11] presents a similar method since the feature maps corresponding to certain keyframes, where the segmentation is obtained through a conventional segmentation network, are propagated to speed up the processing of other images. Movement estimation is no longer done via the optical flow but via Block Motion Vectors.

Some works aim to predict segmentation according to the kinematics [12] or the dynamics of the video [13]. In [12], the evolution of the optical flow in consecutive images is used to predict future segmentation. In [13], a network learns to recognize the 3D dynamics of the video in order to predict, up to half a second in advance of the image capture, the scene segmentation. This type of prediction is more accurate than optical flow predictions but has a high computational cost.

In summary, semantic video segmentation is still a very challenging problem. Most semantic video segmentation approaches consist in getting single-frame predictions using a neural network. Then, this information is propagated using optical flow or LSTM to make the result temporally more consistent. Previous approaches use temporal information on a small time lapse and present a lack of precision when the scene is complex, with moving objects and significant occlusions. The proposed method is different from other methods as spatio-temporal data is integrated all along the segmentation process with a recursive Bayesian filter (III) without being influenced by objects' movements and occlusions using the direct flow detection (IV).

III. Recursive Bayesian Filter for Semantic Segmentation

This section details the proposed Bayesian filtering framework for temporal semantic segmentation (see Figure 1).

Let $\mathcal{I} \doteq \{\mathtt{I}_t\}_{k=1,...T}$ be a temporal set of images extracted from the input video sequence. Let $\mathtt{I}_t(\mathbf{u})$ be the value of the $\mathbf{u} = (x, y)^\top$ pixel (gray level or colour) of an image t.

The semantic segmentation aims at estimating the semantic class of each pixel of the video sequence. Let $Y_{\mathbf{u},t} \in \{1, ...N\}$ be a discrete random variable representing the semantic label (class) associated to the pixel $\mathtt{I}_t(\mathbf{u})$.

The probability function $p(Y_{\mathbf{u},t}|\mathtt{I}_t)$ at iteration t is given by the following equation:

$$p(Y_{\mathbf{u},t}|\mathtt{I}_t) \doteq \frac{1}{N}\sum_{n=1}^{N} q_{\mathbf{u},t}(n)\delta_n(Y_{\mathbf{u},t}) \tag{1}$$

where $q_{\mathbf{u},t}(n)$ is the probability that the pixel $\mathtt{I}_t(\mathbf{u})$ belongs to the class n. $\delta_n(Y_{\mathbf{u},t})$ is the delta Kronecker function[1]. $\mathsf{Q}_t(\mathbf{u}, n)$ denotes the tensor of the probability values associated to the image $\mathtt{I}_t$ such as $\mathsf{Q}_t(\mathbf{u}, n) = q_{\mathbf{u},t}(n)$. The global filter equation, given the present state according to the previous one is:

$$\begin{aligned} &p(Y_{\mathbf{u},t}|\mathtt{I}_t) \propto \\ &p(\mathtt{I}_t|Y_{\mathbf{u},t}) \sum_{Y_{\mathbf{u},t-1}} p(Y_{\mathbf{u},t}|Y_{\mathbf{u},t-1})p(Y_{\mathbf{u},t-1}|\mathtt{I}_{t-1})dY_{\mathbf{u},t-1} \end{aligned} \tag{2}$$

The resolution of this equation is divided into two main steps: prediction and update. The prediction step is solved using the Chapman-Kolmogorov equation:

$$p(Y_{\mathbf{u},t}|\mathtt{I}_{t-1}) = \sum_{Y_{\mathbf{u},t-1} \in \{1,..,N\}} p(Y_{\mathbf{u},t}|Y_{\mathbf{u},t-1})p(Y_{\mathbf{u},t-1}|\mathtt{I}_{t-1}) \tag{3}$$

We propose to compute the proposal discrete probability distribution $p(Y_{\mathbf{u},t}|\mathtt{I}_{t-1})$ from a dense optical flow deep neural network (Figure 1) that estimates $\mathbf{\Delta}_{\mathbf{u},t}$ (the motion of the pixel $\mathtt{I}_t(\mathbf{u})$ between $t-1$ and t):

$$p(Y_{\mathbf{u},t}|\mathtt{I}_{t-1}) = p(Y_{(\mathbf{u}-\mathbf{\Delta}_{\mathbf{u},t}),t-1}|\mathtt{I}_{t-1}) \tag{4}$$

We shall denote $\tilde{\mathsf{Q}}_t(\mathbf{u}, n)$ the tensor with the predicted probabilities:

$$\tilde{\mathsf{Q}}_t(\mathbf{u}, n) = \mathsf{Q}_{t-1}(\mathbf{u} - \mathbf{\Delta}_{\mathbf{u},t}, n) \tag{5}$$

This is the result of warping operation in Figure 1. Probability distributions from previous pixels $(\mathbf{u} - \mathbf{\Delta}_{\mathbf{u},t})$ that do not belong to the image are initialised with a uniform law.

The second step is the update step, using a likelihood function and the Bayes rule:

$$p(Y_{\mathbf{u},t}|\mathtt{I}_t) \propto p(\mathtt{I}_t|Y_{\mathbf{u},t})p(Y_{\mathbf{u},t}|\mathtt{I}_{t-1}). \tag{6}$$

We propose to use a discriminative semantic segmentation deep neural network (see Figure 1) that provides a score associated to each class in the tensor $\mathsf{Z}_t(\mathbf{u}, n)$. We propose here to use classes scores directly as the likelihood function:

$$p(\mathtt{I}_t|Y_{\mathbf{u},t}) \propto \mathsf{Z}_t(\mathbf{u}, Y_{\mathbf{u},t}). \tag{7}$$

The resulting posterior (Segmentation Update, Figure 1) is then expressed by:

$$p(Y_{\mathbf{u},t}|\mathtt{I}_t) \propto \frac{1}{N}\sum_{n=1}^{N} \tilde{\mathsf{Q}}_t(\mathbf{u}, n)\mathsf{Z}_t(\mathbf{u}, n)\delta_n(Y_{\mathbf{u},t}) \tag{8}$$

[1] In probability theory and statistics, the Kronecker delta and Dirac delta function can both be used to represent a discrete distribution. If the support of a distribution consists of points $x = \{x_1, ..., x_n\}$, with corresponding probabilities $p_1, ..., p_n$, then the probability mass function $p(x)$ of the distribution over x can be written, using the Kronecker delta, as $p(x) = \sum_{i=1}^{n} p_i \delta_{xx_i}$. For visibility reason, we will write: $\delta_{xx_i} = \delta_x(x_i)$

IV. Optical flow Model

Here we detail the optical flow model used in the filter (Optical flow Network with Warping operation in Figure 1).

We assume that in a video sequence, the frames are fairly close to each other so that the brightness changes and the point of view modifications are small enough not to significantly impact the quality of the optical flow estimation and therefore the warping operation quality. It means that the displacement of the camera between two frames must be less than a few meters which is not a strong constraint.

Because of objects displacements, hidden areas on a frame may appear in the next frame and visible areas, on the contrary, may disappear. This phenomenon is significant because of the possible appearance of "ghosts" after the warping operation (see the Predicted Images in Figure 2). Note that images are used instead of probabilities distributions to show the copies. The frame at time (t-1) was warped using the optical flow to predict the frame at time (t) (details about the warping operation are given in the following paragraphs). The errors are located especially at the moving objects edges. Their surface area is small compared to the image size but it has an impact on the results as it will be shown in V.

Two types of optical flow can be computed, the direct flow and the reverse flow. The direct flow is the optical flow from the image at time $t-1$ (I_{t-1}) to the image at time t (I_t), and vice versa for the reverse flow. These two flows have very different properties in the case of a prediction from $t-1$ to t, which is the purpose of the warping operation. Indeed, the domains and codomains of the two warping functions are different as the optical flows are computed in opposite directions with the direct and inverse flow.

The optical flow used in our prediction function is a dense optical flow. The output of the optical flow must be a dense matrix giving the displacement of each pixel of image I_t between $t-1$ and t. Standard optical flow algorithms [14][15][16], when given a couple (I_1, I_2) as input, provide the output dense matrix of motion for each pixel of I_1 towards I_2. However, in our case an output dense matrix of motion for each pixel I_2 from I_1 is necessary. This is why the reverse flow is used for the prediction.

The reverse warping operation therefore consists in locating the pixels of frame $t-1$ which most closely match the pixels of frame t and then move the distributions to their position at time t. Problems occur when pixels in frame t do not match any pixel of frame $t-1$, for example, when a moving object reveals a portion of the scene that was invisible on the previous frame. The optical flow associated with the pixels in these zones is the same as the direct environment optical flow (mostly the background which does not move). The distributions in these areas will therefore be the distributions at time $t-1$ at these same positions. However, in the image $t-1$, these areas are masked by objects. This implies that the distributions associated with the objects will be transferred to areas where the object is no longer present. Objects copies may therefore appear (Figure 2).

The purpose of the prediction in our model is to bring coherence to the segmentation process. The prediction must therefore not convey wrong information. To avoid these wrong information problems, the direct optical flow is also used. The direct optical flow connects the pixels from frame $t-1$ to those from frame t. The pixels that appear in frame t are not connected to any pixel from frame $t-1$ and therefore, they are easy to detect.

The direct flow is therefore utilized to avoid occlusions and objects copies problems. A high-amplitude noise is added to these areas to simulate a uniform distribution (after a renormalization of the distributions) and therefore bring no information. Examples of errors detection, i.e. areas where the prediction is impossible due to objects appearances, with the direct flow are shown on Figure 3. The detections are efficient but there are some false positives (pixels wrongly associated to appearing objects). As the purpose of this method is to unable the transmission of wrong information, these false positives are not critical. Nevertheless, it is necessary to avoid false negatives (pixels from appearing objects associated to already existing objects) that are fortunately less numerous. Effects of the false negatives are studied in Part V-C when the noise is set to 0. In this case, the predictions are always taken into account even if the objects appeared for the first time on the image.

V. Experiments

In the following, implementation details of the approach are described, as well as the training data and the evaluation metrics. Finally, the performance of this approach is discussed.

A. Implementation details

The implementation is based on the public library TensorFlow [17]. Spatial dependencies (relations between pixels at fixed time in an image) are modeled by a CNN and temporal ones (relations between pixels intensity through time) are established by optical flow. In this work, considering efficiency and accuracy, a PWC-Net [1] was selected as the baseline for flow estimation (temporal dependencies). To show the efficiency of the method, spatial dependencies are modeled by several segmentation networks: PSPNet [1], DeepLab101 [3], DeepLab50 [3] and BiSeNet [4]. These networks enable an evaluation of the method performance for different baselines.

All the experiments and runtime analyses are performed using a Nvidia GTX Titan X 12 Gb and Intel Core i7-6700K 4.0GHz machine. Pretrained models are used for both optical flow and segmentation networks. As the optical flow network is much faster than the segmentation network, the speed of the whole filter including both networks is similar to the speed of the segmentation network alone.

B. Data set and evaluation metrics

To evaluate our proposed framework, an extensive evaluation on the data set CityScapes [18] is performed. The

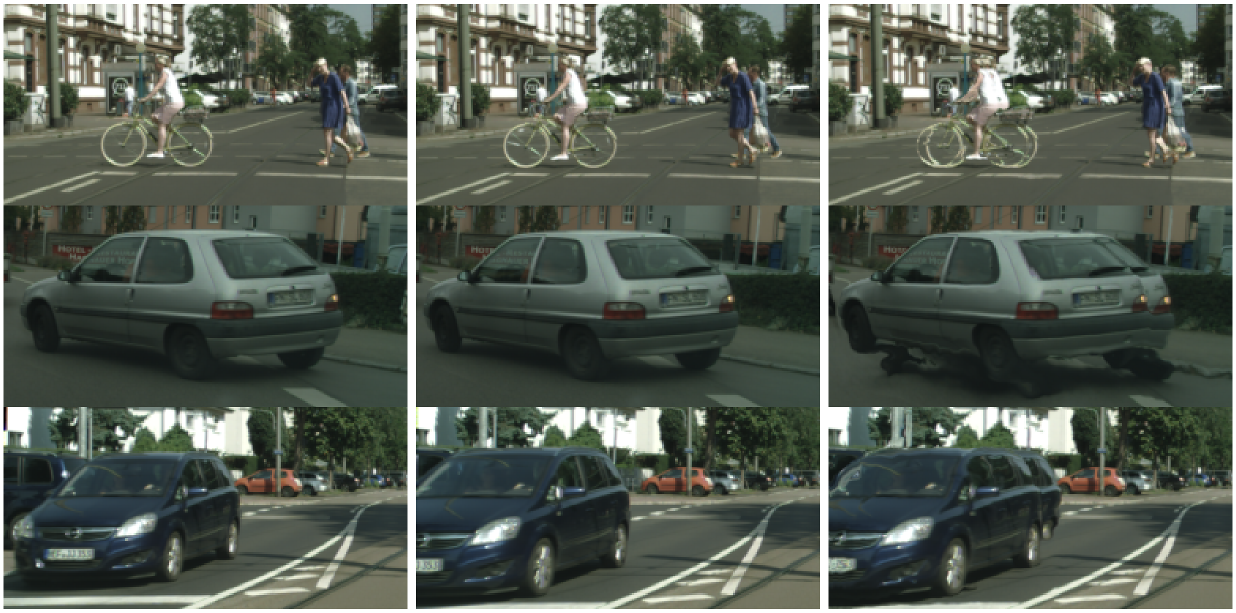

Fig. 2: Example of artifacts due to warping operation on Cityscapes data set

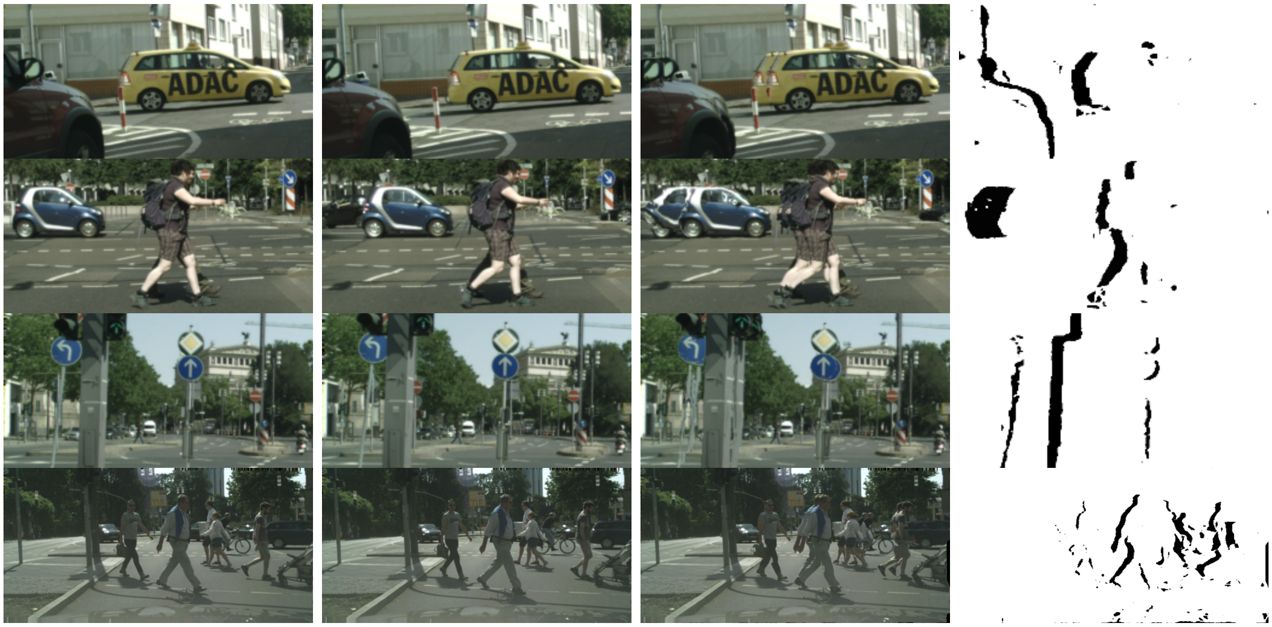

Fig. 3: Examples of artifacts detection with direct optical flow in Cityscapes data set

predicted segmentation maps resulting from our method are evaluated using the mean Intersection over Union[18] (mIoU) on the validation set. The results with temporal consistency are then compared to the baseline method where the predictions are performed per image.

Cityscapes[18] Data set: The Cityscapes data set focuses on street-scene segmentation and autonomous driving. All images are captured from a moving vehicle in various conditions and cities. It contains snippets of street scenes collected from 50 different cities with a resolution of 2048×1024 pixels. The training, validation, and test sets respectively contain 2975, 500, and 1525 snippets. Each snippet has 30 images, where the 20^{th} image is annotated with pixel-level ground-truth labels for semantic segmentation with 19 categories for evaluation (see Table IV for the details of the different categories).

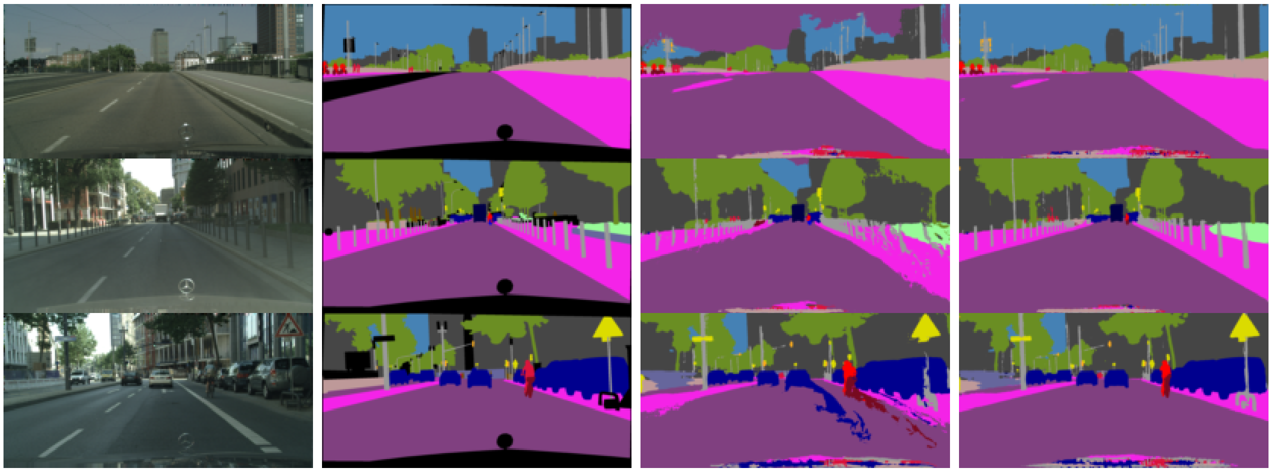
Image (t)
Segmentation groundtruth (t)
Segmentation with noise=0
Segmentation with noise=0.2

Fig. 4: Visual results on Cityscapes dataset for different noise values

Prediction Noise	0	0.05	0.1	0.15	0.2	0.25	Segmentation Network alone
mIoU Filter + PSP-Net	72.60	77.41	**77.45**	77.43	77.39	77.36	76.40
mIoU Filter + DeepLab101	73.23	78.80	78.93	78.98	**79.00**	78.99	78.67
mIoU Filter + DeepLab50	68.01	72.09	**72.11**	72.09	72.07	71.94	71.35
mIoU Filter + BiSeNet	54.62	57.01	56.90	56.81	**57.55**	56.66	56.11

TABLE I: Influence of the prediction noise on the mIoU

Prediction Noise	0	0.05	0.1	0.15	0.2	0.25
mIoU Filter + PSP-Net	62.43	77.27	**77.38**	**77.38**	77.36	77.33
mIoU Filter + DeepLab101	63.03	78.62	78.82	78.90	78.94	**78.96**
mIoU Filter + DeepLab50	59.02	71.88	71.99	**72.01**	72.00	71.99
mIoU Filter + BiSeNet	48.50	**56.87**	56.82	56.75	56.69	56.63

TABLE II: Influence of the prediction noise on the mIoU without direct flow detection

Network used in the filter	PSPNet[1]	DeepLab101[3]	DeepLab50[3]	BiSeNet[4]
Direct flow impact	6.7%	18.2%	15.8%	59.7%

TABLE III: Proportion of mIoU improvement due to the direct flow detection

C. *Prediction uncertainty*

To model the uncertainty on the segmentation prediction, a noise is added to the probability distributions. They are then renormalized to ensure that the resulting distributions are still probability distributions (before the Segmentation Update in Figure 1). This noise makes the balance between the observation (segmentation network) and the segmentation prediction with the optical flow. The higher the noise, the more the segmentation network will influence the result and the less the temporal information will be taken into account. As the prediction is less precise than the observation and depends on the optical flow estimation, if the noise value is too small, some artifacts due to a wrong flow estimation may appear (see Figure 4). This leads to an important loss of global segmentation precision. That's why the noise value must not be randomly selected. Table I presents the influence of the noise value on the final mIoU for different segmentation networks. The maximum precision is reached for noise values between 0.1 and 0.2. Choosing a constant noise of 0.2 assures an optimal improvement for all types of networks. The noise used is the same for each class. Some studies have been carried out to evaluate the noise for each class separately (by computing the mIoU of a groundtruth prediction) but the results did not lead to a significant precision improvement.

D. *Warping and direct flow detection*

In the following, the results of an ablation study on the direct flow detection during reverse warping operation are presented.

The mIoU on Cityscapes validation set with the direct flow

	road	sidewalk	building	wall	fence	pole	traffic light	traffic sign	vegetation	terrain	sky	person	rider	car	truck	bus	train	motorcycle	bicycle
PSPNet [1]	97.3	81.6	91.8	58.1	61.3	55.6	67.9	75.5	91.6	63.2	94.0	79.1	60.4	93.7	74.1	86.1	81.5	63.3	75.3
Filter + PSPNet[1]	**97.4**	**82.0**	**92.2**	**59.0**	**64.3**	**57.2**	**68.9**	**77.2**	**91.9**	**63.9**	**94.1**	**79.6**	**62.0**	**93.9**	**75.8**	**86.9**	**83.0**	**66.3**	**75.9**

TABLE IV: Per-class IoU of the PSPNet (with and without filter) on Cityscapes dataset

detection (Table I) was already computed for different noise values. Now, the mIoU without this detection (Table II) is computed for the same noise values. When the noise is equal to 0 the influence is maximum. The detection improves the mIoU of 10% for the PSP-Net[1] and the DeepLab101[3] for example. This comes from the side effects of the temporal information whose quality rapidly decreases. As the noise is null, the segmentation network has a minimum influence on the result and, therefore, cannot correct the resulting segmentation. When the temporal information is filtered with the direct flow detection, the prediction quality is better and the results are, therefore, better. As the noise increases, temporal information will have less and less impact on the result. Therefore, the improvement due to the defaults detection will become negligible. When the noise is small, a defect on an image may have an impact on the segmentation one second after. But with a higher noise, the defects only have influence on the results when they appear directly on the studied frame or right before it. As the area per frame which may cause defaults is relatively small, the impact is mechanically lowered. That is why the improvement due to the detection lies between less than 0.1% for the DeepLab101[3] and 1% for the BiSeNet[4] when the offset linked to the highest mIoU value is used.

Table III presents the proportion of the mIoU improvement due to the direct flow detection for each studied segmentation network. It represents more than 15% for the DeepLab models and the BiSeNet[4]. The detection method is therefore efficient and has a significant impact on the results.

E. Results

The proposed filter enhances the segmentation consistency and the precision. The results in Table I show that the global mIoU increases approximately of 1% with some variations due to the segmentation network. The lower the segmentation network mIoU, the higher this increase. It reaches 1.6% for a low precision BiSeNet[4]. The extended results for all 19 classes show that the mIoU per class is systematically better with our filter than with the segmentation network alone (example with PSPNet[1] results on Figure IV). In particular, the highest increases concern the low mIoU classes (wall, fence, pole, traffic sign...) where it often reaches between 2 and 3%. Therefore, temporal information enables a better detection of small objects but it also improves the detection of high mIoU classes (building, sky, cars...) which correspond to the backgrounds. Choosing a constant offset of 0.2 ensures the best global results for all the tested segmentation networks. Therefore, it is not valuable to test different offset values.

VI. Conclusion

A spatio-temporal filter is introduced to improve semantic video segmentation through the use of temporal information. Our objective is, therefore, to couple the decisions taken by a CNN with the temporal coherence in video images via optical flow. The optical flow is used to predict a semantic segmentation and the CNN is considered as the observator which will substantiate, or not, the prediction. Temporal information is therefore integrated to correct segmentation estimation. Temporal information is filtered with a new direct flow detection method.

The Bayesian filter enhances segmentation precision by approximately 1% for all the tested segmentation networks even for the high precision networks. The most important accuracy improvement is observed for the least well-detected classes (fence, pole, traffic sign...) where this enhancement can reach 3 or 4% (Table IV). A new prediction refinement method was also developed to transmit only valuable temporal information.

Futur works include: 1) merging the two networks (segmentation and optical flow) into one multitask deep convolutional network and 2) estimating the accuracy of the segmented map by changing the `softmax` layer by a modified version.

References

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

[2] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, Eds. Cham: Springer International Publishing, 2015, pp. 234–241.

[3] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp. 834–848, 2018.

[4] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, “Bisenet: Bilateral segmentation network for real-time semantic segmentation,” in *ECCV*, 2018.

[5] Z. Wu, X. Wang, Y.-G. Jiang, H. Ye, and X. Xue, “Modeling spatial-temporal clues in a hybrid deep learning framework for video classification,” in *Proceedings of the 23rd ACM International Conference on Multimedia*, ser. MM ’15. New York, NY, USA: ACM, 2015, pp. 461–470.

[6] P. Tokmakov, K. Alahari, and C. Schmid, “Learning video object segmentation with visual memory,” *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 4491–4500, 2017.

[7] L. Ding, J. Terwilliger, R. Sherony, B. Reimer, and L. Fridman, “Value of temporal dynamics information in driving scene segmentation,” *arXiv:1904.00758*, 2019.

[8] L. Mou and X. X. Zhu, “Rifcn: Recurrent network in fully convolutional network for semantic segmentation of high resolution remote sensing images,” *CoRR*, vol. abs/1805.02091, 2018.

[9] X. Zhu, Y. Xiong, J. Dai, L. Yuan, and Y. Wei, “Deep feature flow for video recognition,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017, pp. 4141–4150.

[10] R. Gadde, V. Jampani, and P. V. Gehler, “Semantic video cnns through representation warping,” in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct 2017, pp. 4463–4472.

[11] S. Jain and J. E. Gonzalez, “Fast semantic segmentation on video using block motion-based feature interpolation,” *ECCV*, 2018.

[12] S. shahabeddin Nabavi, M. Rochan, and Y. Wang, “Future semantic segmentation with convolutional lstm,” in *BMVC*, 2018.

[13] P. Luc, N. Neverova, C. Couprie, J. Verbeek, and Y. LeCun, “Predicting deeper into the future of semantic segmentation,” *ICCV*, 2017.

[14] B. Horn and B. G. Schunck, “Determining optical flow,” *Artificial Intelligence*, vol. 17, pp. 185–203, 08 1981.

[15] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid, “DeepFlow: Large displacement optical flow with deep matching,” in *IEEE Intenational Conference on Computer Vision (ICCV)*, Sydney, Australia, Dec. 2013.

[16] J. Revaud, P. Weinzaepfel, Z. Harchaoui, and C. Schmid, “Epicflow: Edge-preserving interpolation of correspondences for optical flow,” *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1164–1172, 2015.

[17] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, “TensorFlow: Large-scale machine learning on heterogeneous systems,” 2015.

[18] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, “The cityscapes dataset for semantic urban scene understanding,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 3213–3223.