Improving Semantic Video Segmentation by Dynamic Scene Integration

Farhad Ghazvinian Zanjani[1] Marcel van Gerven[1,2]

[1]Radboud University Nijmegen – Institute for Computing and Information Sciences
[2]Donders Institute for Brain, Cognition and Behaviour

Abstract: Multi-class image segmentation and pixel-level labeling of the frames that make up a video could be made more efficient by incorporating temporal information. Recently, Convolutional Neural Networks (ConvNets) have made an impressive positive impact on the single image segmentation problem. In this paper, in order to further increase labeling accuracy, we propose a method for integrating short-term temporal information with structural scene information by using a conditional random field (CRF). In our proposed method, ConvNet prediction refinement was achieved by exploiting a fully connected CRF as a post-processing step. Our main contribution is focused on taking into account the scene dynamics in semantic image segmentation. For extracting these dynamics we used scene dense optical flow. Inference in this dynamic CRF will consider both scene appearance and dense optical flow information. We show that by utilizing temporal information, the accuracy of semantic image segmentation can be improved with a small incurring additional computational overhead. Our proposed method, achieved 64.5 average IoU score applied on the Cityscapes urban data set with 19 different semantic classes compared to 62.1 IoU when no optical flow information was employed.

Introduction

Low-level video segmentation is an important objective in many application areas such as robotics, object tracking, video coding, video perception, action recognition and scene understanding. The video segmentation problem, often boils down to processing of the individual images of a sequence while ignoring the dynamic information among frames that can be derived from the relative scene displacements. Because the performance of scene parsing based on processing of single images suffers from changes in the appearance of objects due to e.g. considerable changes in illumination, available motion based clues in video are a valuable source of information that can be considered for image semantic segmentation [Zhang et al. 2010]. Although the beneficial effects of using the temporal data in video are acknowledged, in comparison with the extensive research that has been carried out on single image segmentation,

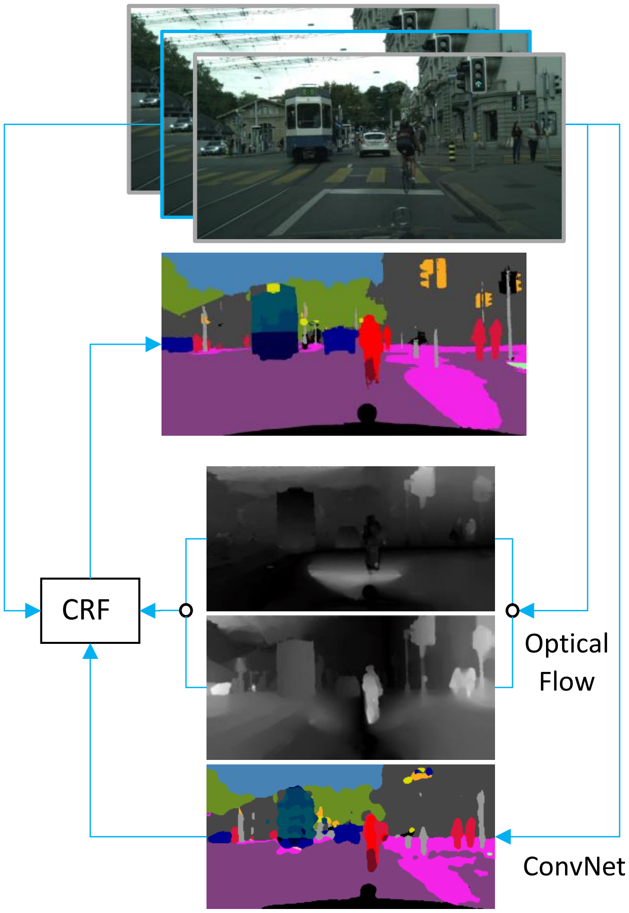


Fig. 1. Semantic video segmentation workflow

just a few studies were carried out on dynamic video segmentation. The present study tries to fill a gap in the literature by proposing a new methodology for integrating dynamic and structural information of a scene for pixel-level labeling of video frames.

The past decade has seen the rapid successful development of Convolutional Neural Networks (ConvNets) in many fields of machine learning applications. As a matter of fact, ConvNets have been successfully applied to pixel level classification [Ciresan et al. 2012, Farabet et al. 2013, Ronneberger et al. 2015, Long et al. 2015]. Research has shown that the use of standard ConvNet architectures for low-level processing such as pixel labeling, has some shortcomings [Long et al. 2015, Chen et al. 2016]. While these models show promising results in coding of high-level image structure, which is an advantage in object-oriented recognition tasks such as object categorization, bounding box localization, action recognition and video classification, because of their low spatial accuracy, these models have some shortcomings in pixel-level classification tasks [Chen et al. 2014].

Recently, several attempts have been made to address the problem of low spatial accuracy of standard ConvNets. Previous studies on the image segmentation problem have introduced two successful approaches. One approach is tangled multi-resolution processing of the input image by introducing bypass connections between intermediate and output layers of a ConvNet [Long et al. 2015]. This modification in the standard architecture of a ConvNet implements data flows fusion by combining semantic information from a deep, coarse layer with appearance information from a shallow, fine layer. Another approach is constructing coarse-to-fine hierarchical predictor networks, by passing the preceding coarse prediction to another ConvNet [Eigen & Fergus 2015]. In the context of the second approach, some researchers used probabilistic graphical models such as Conditional Random Fields (CRFs) as a post-processing stage for a pixel classifier that can be a ConvNet [Chen et al. 2014]. The CRF model is applied to the low-resolution prediction of the ConvNet for increasing the accuracy of the pixel-level label assignment. A similar method can be seen in [Zheng et al. 2015] where the CRF was embedded as the last layer of a ConvNet. By reformulating the inference stage of the CRF model, it became possible to train both the ConvNet and CRF model parameters simultaneously end-to-end by using gradient-based optimizations.

Our method follows the second approach. Here we use the CRF as a post-processing unit not just for pairwise labeling of pixels in a single image but also for integrating temporal information between successive frames (Fig. 1).

In order to take into account the temporal information among the frames of a video in a ConvNet, two different strategies have been devised by researchers. These strategies can be encountered in the computer vision and image perception community, mostly for video classification and action recognition tasks. According to the first strategy, multiple frames of a video in a predefined time window (overlapping blocks) are used as input to a ConvNet for performing prediction on a single reference frame such as predicting the tag of a reference frame in video classification task. In this context, information fusion among multiple frames depends on the structure of ConvNet [Karpathy et al. 2014]. The ConvNet architecture in this approach similar to Time Delay Neural Network (TDNN) needs to train on the sequential data by using a sliding window on different time points [Wöhler & Anlauf 2001]. Although these sort of architectures are ideal for analyzing motion patterns but providing a large scale sequential data with pixel-level label annotated could be a problematic issue in video segmentation task. Furthermore, the advantage of pre-training the ConvNet on the available single image large-scale databases like Pascal VOC [Everingham et.al, 2015] cannot be held anymore.

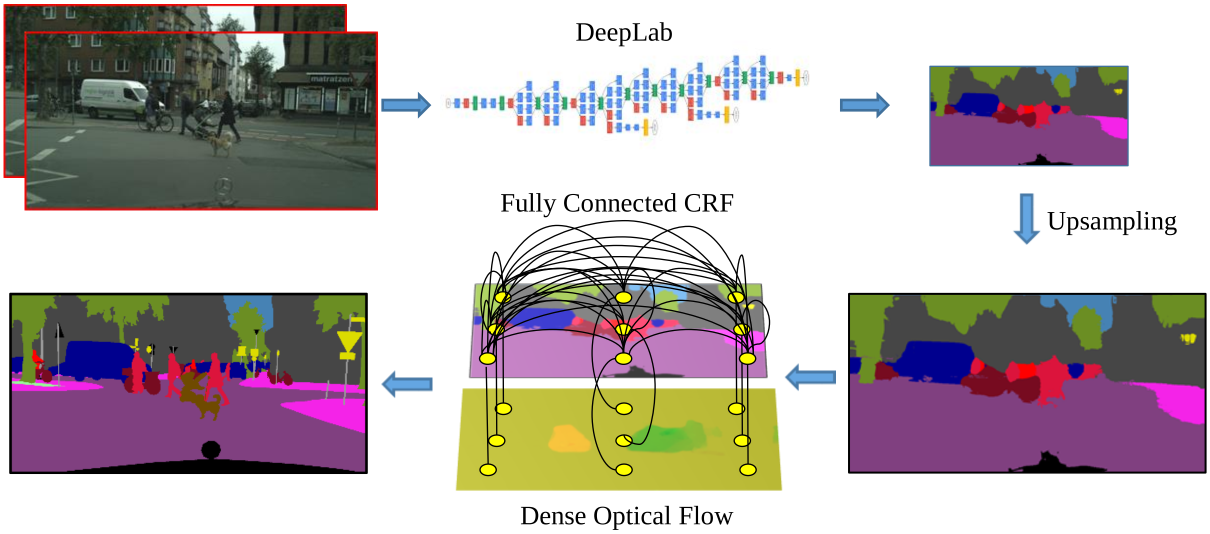


Fig. 2. Schematic of the processing sequence on input data [Chen et al. 2014]

Another strategy for including temporal information is by using short or long-term optical flow displacement fields between several successive frames [Wu et al. 2015, Simonyan & Zisserman 2014, Kundu et al. 2016]. To this end, in recent work, besides training the ConvNet for learning the spatial structure in images, a second network was used for learning the scene optical flow to capture the temporal information of the stream of images [Wu et al. 2015, Simonyan & Zisserman, 2014]. Similar to this approach, we utilize the scene optical flow of successive frames in the context of a semantic video segmentation task but instead of employing a second network that should be trained in parallel for learning the optical flow, we include the optical flow as a temporal consistency criterion between semantic objects among frames in the post processing stage. Incorporating the temporal data among frames can be done by adding the displacement vector field to the feature space of the pairwise potential term of the CRF model. This approach has two advantages in comparison with the method that has been mentioned previously. Firstly, different to the training of a ConvNet with an architecture similar to a TDNN, our model can be trained also on a database with single annotated images. Secondly, we don't need to define a dense CRF with edges between two or more frames which reduces the computational cost dramatically. Our approach is an extension to [Chen et al. 2014] that used a ConvNet as a pixel label unary classifier (called *DeepLab*) and a post-processing CRF model. We extend the post processing spatial CRF model to work on sequences of images by importing scene optical flow and doing inference on spatiotemporal data for increasing the accuracy of pixel labeling (Fig. 2).

Integrating ConvNet, CRF and Dense Optical Flow Components

For performing efficient inference in a fully connected CRF model, we used the method proposed by [Krähenbühl & Koltun, 2012]. In this section, we will describe how the CRF model is used for segmenting a sequence of images by using spatiotemporal data.

Inference in a Fully Connected CRF on Spatiotemporal Data

Let us consider a random field for the labels assigned to the pixels which is defined over a set of random variables **X**,

$$\mathbf{X} = \{X_1,..., X_n\}$$

Each random variable can take on a value in a set of predefined labels. In addition, we need to consider a random field defined over random variables **I**,

$$\mathbf{I} = \{I_1,...,I_N\}$$

where **I** represents input images of size N. For example, in case of color images, I_j is the RGB vector of pixel j and x_j indicates the label assigned to pixel j.

The distribution of (**X**, **I**) is a CRF when the random variables **X** conditioned on **I**, obey the Markov property w.r.t. the graph:

$$p\left(X_i \middle| I, X_j, i \neq j\right) = p\left(X_i \middle| I, X_j, i \sim j\right) \qquad (1)$$

where i~j indicates that *i* and *j* are neighbors in graph. This CRF is characterized by a Gibbs distribution:

$$P(X|I) = \frac{1}{Z(I)} e^{-E(X|I)} \qquad (2)$$

where the energy is defined as

$$E(X|I) = \sum_{c \in C_G} \psi_c(X_c|I) \qquad (3)$$

and *G* is the graph on **X** and each clique *c* in the set of cliques C_G in *G* induces a potential ψ_c. In a fully connected pairwise CRF model, *G* is a complete graph on X and C_G is the set of all unary and pairwise cliques. So, the corresponding Gibbs energy can be written as:

$$E(X|I) = \sum_i \psi_u(x_i) + \sum_{i<j} \psi_p\left(x_i, x_j\right) \qquad (4)$$

where *i* and *j* range from *1* to *N*. The unary potential $\psi_u(x_i)$ is computed independently for each pixel by training a classifier. The classifier produces a distribution over X_i given image features such as shape, texture, location and color, as used in [Krähenbühl and Koltun, 2011]. This classifier can be a deep ConvNet [Chen et al. 2014, Zheng et al. 2015].

As computing $P(\boldsymbol{X}|\boldsymbol{I})$ is intractable, similar to [Krähenbühl and Koltun, 2011], we used mean field approximation that replaces the P(**X**) distribution with an easier factorial distribution $Q(\boldsymbol{X})$ for doing inference.

$$Q(\boldsymbol{X}) = \prod_i Q_i(X_i) \qquad (5)$$

By minimizing the KL-divergence D(Q||P) among all distributions Q we obtain an approximation for P(**X**).

The key idea for efficient inference in fully connected models lies in defining the pairwise edge potentials as a linear combination of Gaussian kernels in an arbitrary features space. In [Krähenbühl and Koltun, 2011], this feature space for single image segmentation has been taken to consist of spatial location and RGB values. The author introduces a pairwise potential of the form:

$$\psi_p\left(x_i, x_j\right) = \mu\left(x_i, x_j\right) \sum_{m=1}^{K} \left[w^{(m)}. k^{(m)}\left(f_i, f_j\right)\right] \qquad (6)$$

where $\mu(x_i, x_j)$ is a label compatibility function which can be defined in a simple way by using a Potts model:

$$\mu(x_i, x_j) = \begin{cases} 1 & \text{if } x_i \neq x_j \\ 0 & \text{otherwise} \end{cases} \qquad (7)$$

and where $k^{(m)}$ is a Gaussian kernel defined as:

$$k^{(m)}\left(f_i, f_j\right) = \exp\left[-\frac{1}{2}\left(f_i, f_j\right)^T \Lambda^{(m)}\left(f_i, f_j\right)\right] \qquad (8)$$

The vectors f_i and f_j are feature vectors consisting of spatial position, color and, in our work, also field displacement between two successive frames for pixels i and j. The terms $w^{(m)}$ are linear combination weights. The shape of each kernel $k^{(m)}$ is characterized by a symmetric positive-definite precision matrix $\Lambda^{(m)}$.

For the multi-class image sequence segmentation and labeling, we propose using three kernels (*m*=3) that are defined in terms of the positions p_i and p_j, color vectors I_i and I_j and the difference of field displacement vectors (optical flow) U_i and U_j.

$$\begin{aligned} k\left(f_i, f_j\right) = {} & w^{(1)} \exp\left(\frac{\left|p_i - p_j\right|^2}{2\sigma_\alpha^2}\right) \\ & + w^{(2)} \exp\left(\frac{\left|p_i - p_j\right|^2}{2\sigma_\beta^2} - \frac{\left|I_i - I_j\right|^2}{2\sigma_\gamma^2}\right) \\ & + w^{(3)} \exp\left(\frac{\left|p_i - p_j\right|^2}{2\sigma_\beta^2} - \frac{\left|U_i - U_j\right|^2}{2\sigma_\delta^2}\right) \end{aligned} \qquad (9)$$

Here, the U_i and U_j are the absolute differences of optical flow in two directions between preceding and following frames w.r.t. the reference frame for pixel i and j. The first term in equation (9) is a smoothness kernel which removes small disjoint regions. The second term is an appearance kernel which emphasizes nearby pixels with similar color are likely to be in the same class [Krähenbühl and Koltun, 2011]. The third term is a temporal kernel which measures the temporal consistency between pixels of two successive frames in video.

The kernel parameters will be learnt from data by using grid search on a validation set that consists of 10 images with known ground truth. Figure 3 plots the high-dimensional grid search space of hyper-parameters in equation (9) against the segmentation performance (IoU score) assessed on a validation set. Five different choices for the contribution of temporal information ($w^{(3)} \in \{0,1,2,3,5\}$) are indicated by different graph colors.

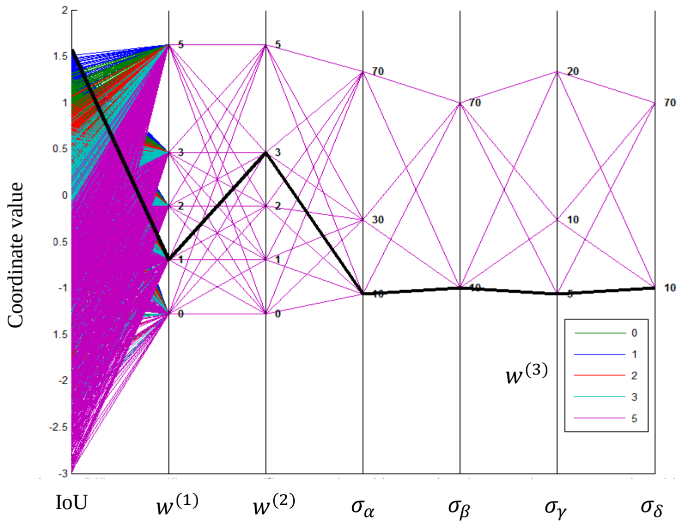


Fig. 3. Parallel coordinates plot of grid search space for adjusting seven hyper-parameters of equation (9). Different color graphs represent different values for the weight of temporal kernel ($w^{(3)}$). The left axis show normalized average IoU score and the solid black line represent the best parameter set by evaluating on validation set.

Embedding the optical flow information by defining a temporal kernel in the pairwise potential $\psi_p(x_i, x_j)$ of the energy function has the advantage of doing inference in a complete graph that has been defined just on pixels of a single image field. Therefore, we do not need to add new edges between consecutive frames for incorporating the dynamic information as introduced in [Wang & Ji, 2005] and [Xiao and Quan, 2009] or by defining a massive complete graph on a block of frames [Kundu et al. 2016].

Experimental Results

We evaluated our method on a new dataset for scene understanding in an urban environment. The Cityscapes Dataset [Cordts et al. 2015 and 2016] consists of a sequence of urban street scene images with 5000 annotated frames with 2048×1024 pixel size (2975 train/500 validation/ 1525 unrevealed test frames). The images have been recorded from 50 different cities during different seasons. The Cityscapes dataset consists of 30 classes of which 19 are used as semantic labels in evaluations. Any other object in the frames that do not belong to these 19 classes should be assigned a void (i.e. unknown) label in prediction. In addition, each annotated frame in the dataset is the 20th image from 30 frame video snippets.

For tackling the limitation of GPU memory in training the ConvNet on full resolution images, we trained the ConvNet on randomly cropped patches of size 318×318 pixels from original training images. In the prediction mode due to the use of a fully convolutional network in the DeepLab architecture, the network can perform prediction on arbitrary input image sizes, including the original full resolution images.

For assessing the semantic segmentation result, the standard Jaccard Index that commonly known as Pascal VOC Intersection over Union (IoU) metric [Everingham et.al, 2015],was measured on the unrevealed test set[1].

$$IoU = \frac{TP}{TP+FP+FN} \tag{10}$$

Here, TP, FP and FN are true positive, false positive and false negative for each class. Table 1 shows the average and individual IoU on 1525 test images by different state-of-the-art methods.

[1] www.cityscapes-dataset.com

	Average	Road	Sidewalk	Building	Wall	Fence	Pole	Traffic light	Traffic sign	Vegetation	Terrain	Sky	Person	Rider	Car	Truck	Bus	Train	Motorcycle	Bicycle
Adelaide_context [Shen & Reid, 2015]	71.6	98	83	91	44	51	51	65	72	92	72	94	82	61	94	61	65	54	62	71
DeepLabv2-CRF [Chen et al. 2016]	70.4	98	81	90	49	47	50	58	67	92	69	94	80	60	94	57	68	58	58	69
LLR-4x [Ghiasi & Fowlkes, 2016]	68.3	98	79	90	42	48	57	65	69	92	69	95	81	59	94	42	55	44	52	68
Dilation10 [Yu and Koltun, 2015]	67.1	98	79	90	37	48	53	59	65	92	69	94	79	55	93	46	53	48	52	66
DPN [Li et al. 2015]	66.8	98	79	90	40	46	51	57	65	92	69	95	78	54	93	45	53	50	52	65
Adelaide [Lin et al. 2015]	66.4	97	79	88	45	48	34	56	62	90	70	92	73	52	91	55	62	52	55	63
FCN 8s [Long et al. 2015]	65.3	97	78	89	35	44	47	60	65	91	69	94	77	51	93	35	49	47	52	67
Our method (DeepLab+DynamicCRF)	**64.5**	97	78	89	38	45	39	51	62	91	58	94	77	54	93	42	53	50	53	64
Pixel-level Encoding for Instance Segmentation [Uhrig et a. 2016]	64.3	97	78	89	28	40	52	60	65	91	68	94	78	54	92	34	42	43	53	67
DeepLab LargeFOV Strong [Chen et al. 2014]	63.1	97	78	88	44	41	30	45	55	89	67	93	71	49	91	49	57	49	48	59
CRFasRNN [Zheng et al. 2015]	62.5	96	74	88	48	41	35	50	60	91	66	94	70	35	90	39	58	55	44	55
ENet [Paszke et al. 2016]	58.3	96	74	85	32	33	44	34	44	89	61	91	66	38	91	37	51	48	39	55
Segnet basic [Badrinarayanan et al. 2015]	57.0	96	73	84	29	29	36	40	45	87	64	92	63	43	89	38	43	44	36	52

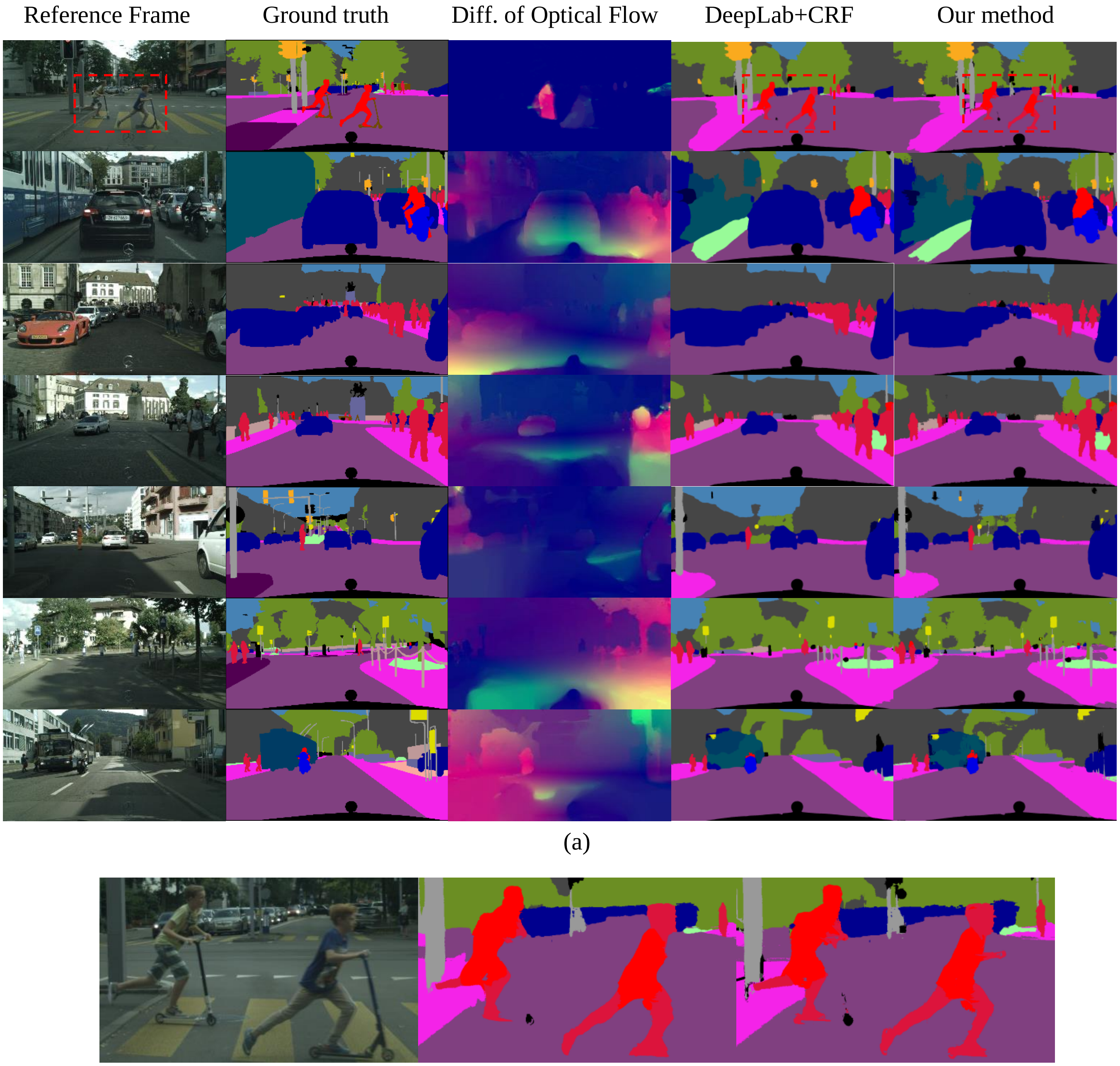
Table 1. Benchmark on Cityscapes dataset based on average of IoU score

Figure 4 shows some example frame and the scene optical flow, ground truth and the prediction of DeepLab+CRF method with and without using temporal data. As can be seen from the figure, the label prediction over some regions of image improves when the temporal information is incorporated in the energy function. This has been shown quantitatively in Table 2.

Experiment	Average IoU
DeepLab+CRF (baseline)	62.1
DeepLab+CRF+Opticalflow	64.5

Table 2. Segmentation performance on the test set with and without using temporal information.

Related Work



(a)

(b)

Fig. 4 (a) Results on some untrained Cityscapes images (b) zoomed inside rectangle

Improvement of the accuracy of scene parsing by incorporating temporal information in a video has been reported before [Zhang et al. 2010, Xiao & Quan 2009, Wang & Ji 2005, Yu & Koltun 2015]. In [Zhang et al. 2010] a set of extracted features from a dense depth map for each superpixel of an image were classified using a Random Forest. To enhance segmentation accuracy, the author used a weighted average of the posterior probabilities of label assignment for each superpixel over neighbor frames. Although this kind of smoothing approach can help for improving the accuracy when the displacement field between neighboring frames is small, this condition can be violated easily at the boundaries of objects. For example those regions that belong to moving cars in urban scene images. In other related work for image sequence segmentation by using probabilistic graphical models [Xiao & Quan 2009, Wang & Ji 2005], the authors attempt to include the temporal information among multiple frames in the video for predicting the pixel label of the reference frame. The key idea is to extend the graph edges across the frames. This approach will capture temporal consistency but to make inference in such a huge graph feasible, two modifications were devised which potentially can limit the performance of the image segmentation. Firstly, the graph has been defined on superpixels which are the outcome of an over-segmenting process. Consequently, any inefficiency or occurring error in this process of finding superpixels in images cannot be corrected in the remainder of the computations. Secondly, to remedy the computational cost of inference in a dense graph, they defined some constrains on graph topology like the number of edges that can connect to each node [Xiao & Quan, 2009] or a limitation on the spatial length of edges in the graph [Wang & Ji 2005]. Due to a breakthrough method for performing efficient inference in a dense CRF by [Krähenbühl & Koltun, 2012], now it is possible to define a dense graph on the pixels of a single image.

Instead of defining a graph with edges between the frames of a video and in contrast to the previous approaches, we used scene dense optical flow to utilize the temporal consistency between frames. By using the dense optical flow as estimated by the method described in [Brox et al. 2007], we expected that together with the brightness and gradient consistency as well as spatiotemporal smoothness constraints between two successive frames label assignments could be improved. For better capturing the structure of the scene, we used the absolute difference of estimated displacement field between previous and next frames relative to the reference frame.

Very recently, it came to our attention that another group has pursued a very similar approach for extending the idea of using fully connected CRF for doing inference in multiple frames of video [Kundu et al. 2016]. Although both approaches are highly similar, there are some main differences in technical and theoretical aspects of our work and theirs. Kundu et al. used Dilation Unary classifiers [Yu & Koltun 2015] which was reported to yield a slightly higher accuracy than the unary classifier (DeepLab) that we used according to the evaluation that has been done in [Cordts et al. 2016]. The main difference between our approach and [Kundu et al. 2016] pertains to the way that temporal information has been incorporated in the CRF model. They defined a dense CRF with edges over a number of frames (around 100 frames) centered by the reference frame while we just define the CRF with edges inside the reference frame. In our method, the displacement field will be added as an extra feature to the feature space of the pairwise potential term by defining a temporal consistency kernel. In our method we just use the optical flow that has been extracted from preceding and following frames to the reference frame. This approach will reduce the computational time of our proposed method considerably relative to the alternative implementation.

Conclusion

In this paper, our aim was to consider the scene dynamic information between the streams of frames in the video in order to increase the accuracy of semantic image segmentation. We showed that the integration of temporal information such as optical flow and the appearance based visual content of the scene can be done by using a CRF model. Instead of defining a dense graph over two or multiple frames of video and consequently doing inference in this extremely high-dimensional space, we included the temporal consistency between frames as an extra feature in the pairwise potential of the CRF energy function. This CRF model has been defined over the pixels of a single image field. This makes application of our method computationally feasible. However, this study has been unable to outperform recent state-of-the-art methods on the benchmark leaderboard. Our results do support the idea that involving scene dynamic information is beneficial when comparing results with a baseline architecture that ignores temporal consistency.

References

[Badrinarayanan et al. 2015] Badrinarayanan, V., Handa, A., & Cipolla, R. (2015). Segnet: A deep convolutional encoder-decoder architecture for robust semantic pixel-wise labelling. arXiv preprint arXiv:1505.07293.

[Brox et al. 2007] Brox, T., Bruhn, A., Papenberg, N., & Weickert, J. (2004, May). High accuracy optical flow estimation based on a theory for warping. In European conference on computer vision (pp. 25-36). Springer Berlin Heidelberg.

[Ciresan et al. 2012] Ciresan, D., Giusti, A., Gambardella, L. M., & Schmidhuber, J. (2012). Deep neural networks segment neuronal membranes in electron microscopy images. In Advances in neural information processing systems (pp. 2843-2851).

[Chen et al. 2014] Chen, L. C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2014). Semantic image segmentation with deep convolutional nets and fully connected crfs. arXiv preprint arXiv:1412.7062.

[Chen et al. 2016] Chen, L. C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2016). DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. arXiv preprint arXiv:1606.00915.

[Cordts et al. 2015] Cordts, M., Omran, M., Ramos, S., Scharwächter, T., Enzweiler, M., Benenson, R. & Schiele, B. (2015). The cityscapes dataset. In Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) Workshops (Vol. 3).

[Cordts et al. 2016] Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R. & Schiele, B. (2016). The cityscapes dataset for semantic urban scene understanding. arXiv preprint arXiv:1604.01685.

[Eigen et al. 2015] Eigen, D., & Fergus, R. (2015). Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. InProceedings of the IEEE International Conference on Computer Vision (pp. 2650-2658).

[Everingham et.al, 2015] Everingham, M., Eslami, S. A., Van Gool, L., Williams, C. K., Winn, J., & Zisserman, A. (2015). The pascal visual object classes challenge: A retrospective. International Journal of Computer Vision, 111(1), 98-136.

[Farabet et al. 2013] Farabet, C., Couprie, C., Najman, L., & LeCun, Y. (2013). Learning hierarchical features for scene labeling. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 35(8), 1915-1929.

[Ghiasi & Fowlkes, 2016] Ghiasi, G., & Fowlkes, C. (2016). Laplacian Reconstruction and Refinement for Semantic Segmentation. arXiv preprint arXiv:1605.02264.

[Karpathy et al. 2014] Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., & Fei-Fei, L. (2014). Large-scale Video Classification with Convolutional Neural Networks.

[Krähenbühl & Koltun, 2012] Krähenbühl, P., & Koltun, V. (2012). Efficient inference in fully connected crfs with gaussian edge potentials. arXiv preprint arXiv:1210.5644.

[Kundu et al. 2016] Kundu, A., Vineet, V., & Koltun, V. (2016). Feature space optimization for semantic video segmentation. CVPR.

[Li et al. 2015] Liu, Z., Li, X., Luo, P., Loy, C. C., & Tang, X. (2015). Semantic image segmentation via deep parsing network. In Proceedings of the IEEE International Conference on Computer Vision (pp. 1377-1385).

[Lin et al. 2015] Lin, G., Shen, C., & Reid, I. (2015). Efficient piecewise training of deep structured models for semantic segmentation. arXiv preprint arXiv:1504.01013.

[Long et al. 2015] Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 3431-3440).

[Paszke et al. 2016] Paszke, A., Chaurasia, A., Kim, S., & Culurciello, E. (2016). ENet: A Deep Neural Network Architecture for Real-Time Semantic Segmentation. arXiv preprint arXiv:1606.02147.

[Ronneberger et al. 2015] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. In Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015 (pp. 234-241). Springer International Publishing.

[Shen & Reid, 2015] Lin, G., Shen, C., & Reid, I. (2015). Efficient piecewise training of deep structured models for semantic segmentation. arXiv preprint arXiv:1504.01013.

[Simonyan & Zisserman, 2014] Simonyan, K., & Zisserman, A. (2014). Two-stream convolutional networks for action recognition in videos. In Advances in Neural Information Processing Systems (pp. 568-576).

[Uhrig et al. 2016] Uhrig, J., Cordts, M., Franke, U., & Brox, T. (2016). Pixel-level encoding and depth layering for instance-level semantic labeling. arXiv preprint arXiv:1604.05096.

[Wang & Ji, 2005] Wang, Y., & Ji, Q. (2005, June). A dynamic conditional random field model for object segmentation in image sequences. In Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE

Computer Society Conference on (Vol. 1, pp. 264-270). IEEE.

[Wöhler & Anlauf 2001] Wöhler, C., & Anlauf, J. K. (2001). Real-time object recognition on image sequences with the adaptable time delay neural network algorithm—applications for autonomous vehicles. Image and Vision Computing, 19(9), 593-618.

[Wu et al. 2015] Wu, Z., Wang, X., Jiang, Y. G., Ye, H., & Xue, X. (2015, October). Modeling spatial-temporal clues in a hybrid deep learning framework for video classification. In Proceedings of the 23rd ACM international conference on Multimedia (pp. 461-470). ACM.

[Xiao and Quan, 2009] Xiao, J., & Quan, L. (2009, September). Multiple view semantic segmentation for street view images. In 2009 IEEE 12th international conference on computer vision (pp. 686-693). IEEE.

[Yu and Koltun, 2015] Yu, F., & Koltun, V. (2015). Multi-scale context aggregation by dilated convolutions. arXiv preprint arXiv:1511.07122.

[Zhang et al. 2010] Zhang, C., Wang, L., & Yang, R. (2010). Semantic segmentation of urban scenes using dense depth maps. In Computer Vision–ECCV 2010 (pp. 708-721). Springer Berlin Heidelberg.

[Zheng et al. 2015] Zheng, S., Jayasumana, S., Romera-Paredes, B., Vineet, V., Su, Z., Du, D. & Torr, P. H. (2015). Conditional random fields as recurrent neural networks. In Proceedings of the IEEE International Conference on Computer Vision (pp. 1529-1537).

Appendix

A1. Optical Flow

Optical flow is the distribution of apparent velocities of movement of brightness patterns in an image that can give an important information about the spatial arrangement of the objects in the scene [Horn & Schunck 1981]. The relative movement of the object and the camera is the source of optical flow in video data. In this section, at first we briefly describe the optical flow formulation and then we will explain the method that we used in our project for estimating of the scene optical flow.

Let consider the image brightness at point (x,y) and at the time t with I(x,y,t). Now if we ignore the changes in light source during a short time, the brightness of a particular point in the pattern will remain constant while the pattern moves ($\frac{\partial I}{\partial t} = 0$), then:

$$I(x, y, t) = I(x + u, y + v, t + 1) \qquad (11)$$

By using the chain rule for differentiation, we have:

$$\frac{\partial I}{\partial x} \cdot \frac{\partial x}{\partial t} + \frac{\partial I}{\partial y} \cdot \frac{\partial y}{\partial t} + \frac{\partial I}{\partial t} = 0 \rightarrow I_x . u + I_y . v + I_t = 0 \vee \nabla I^T . \vec{V} = -I_t \qquad (12)$$

where I_x, I_y and I_t are gradients of the image in x, y and t dimensions respectively and the vector $\vec{V}$ is the displacement field vector in x and y directions. As we can see this linear equation with two unknown u and v, cannot be solved. This is known as *aperture problem* of optical flow that needs additional constrain to be soluble. A considerable amount of literature has been published on devising constrains for this problem. In [Baker et al. 2011] a comprehensive study for evaluating recent works in this field has been reported. Most of the proposed algorithms for optical flow tackle the problem as the optimization of a global energy function. This energy function consists of two terms:

$$E_{Global} = E_{Data} + \lambda . E_{Prior} \qquad (13)$$

where the E_{Data} presents the consistency of the optical flow with the input images. As mentioned before because the data term is ill-posed with fewer constraints than unknowns the prior term E_{Prior} denotes the constraints favor a certain flow fields over others. One common constrains could be a smoothness prior [Baker et al. 2011].

Most accurate optical flow algorithms require several seconds to many minutes per frame for estimating a dense displacement field over all pixels of an image. There are some efficient methods that implemented on GPU and are publically available. In this project, we used the dense optical flow as estimated by the method described in [Brox et al. 2007]. This algorithm was implemented on GPU and is freely available in OpenCV[2] library for both academic and commercial use. In Brox et al. method, instead of considering equation (11) for using raw intensity values, the gradient of the image is considered. So the equation (11) is replaced with

$$\nabla I(x,y,t) = \nabla I(x+u, y+v, t+1) \qquad (14)$$

This modification benefits the robustness of gradient w.r.t. illumination changes in compare with raw intensities and could be more desirable for outdoor images such as urban scene images. According to this method, the data term is defines as

$$E_{Data}(u,v) = \int_{\Omega} \psi(|I(x+w) - I(x)|^2 + \gamma|\nabla I(x+w) - \nabla I(x)|^2)\, dx \quad (15)$$

where $\psi(s^2) = \sqrt{s^2 + \epsilon^2}$ role like L1 normalization and ϵ is just used for numerical reasons. For the prior term they introduced a spatiotemporal smoothness term:

$$E_{prior}(u,v) = \int_{\Omega} \psi(|\nabla_{\varepsilon} u|^2 + |\nabla_{\varepsilon} v|^2)\, dx \qquad (16)$$

where $\nabla_{\varepsilon}\colon \left(\partial_x, \partial_y, \partial_t\right)^T$ presents spatiotemporal gradient. Therefore, the total energy function according to equation (13) is determined. For minimization of this nonlinear energy function, the author used Euler-Lagrange minimization and numerical approximation. Here, we skip the details of the energy optimization for this method that is out of the scope of this project.

The discontinuity pattern in the estimated scene optical flow is a source of discrimination between objects in the scene for image segmentation task. We employed this optical flow information that preserves the coherence of sequential images for video segmentation. According to our experiments, the computational time of this algorithm for images with 2048×1024 pixels on GPU GTX960M is around 4 fps.

A2. Inference in Conditional Random Field (CRF)

The Conditional Random Field (CRF) is a well-known probabilistic method for structured prediction. There are many data analysis applications that involve predicting a large number

[2] opencv.org

of variables that depend both on each other and on other observed variables. As an example, for image segmentation task in computer vision field, the label of every pixel in a natural image depends on both the labels of its neighbors and the visual contents of the image. The CRF was employed in wide areas and particularly it is well suited to image segmentation in computer vision, labeling of the words in a sentence in natural language processing [Sha & Pereira 2003, Smith & Osborne 2005], sequence modeling in speech recognition [Zweig & Nguyen 2009], gene finding in bioinformatics [Settles 2004] and some other miscellaneous application like estimating the score in the GO game [Stern et al 2004].

An important advantage of employing the CRF in compare with other ordinary classifiers is the prediction of a label for a single sample by considering the other labels for the sequence of input samples. The sequential data in this context can be defined in the spatial or temporal domain.

CRF is a type of probabilistic graphical model (PGM) which a probabilistic model is expressed by a graph that structured between random variables according to their conditional dependencies. PGM can be divided into two categories; generative or discriminative models. While generative models try to model a joint probability distribution P(**X**,**I**) over input (**I**) and output (**X**) , the discriminative models attempt to model the conditional distribution P(**X**|**I**) directly which is demanded in classification purposes. However, in general, generative models have advantages but when input data has high dimensionality and strong dependencies among its variables, constructing a probability distribution over them can lead to an intractable model and on the other hand ignoring these dependencies can lead to an inaccurate model. In these cases, a discriminative model such as CRF has the advantage of compactly model the multivariate output (**X)** with the ability to conditioning on input data (**I)** for prediction [Sutton & McCallum 2010]. In continue, we explain the methods for doing inference in CRF models that refers to the computing of the marginal distribution P(**X**|**I**).

Several algorithms for performing inference in CRF model similar to any other graphical models have been proposed in textbooks. However there are some exact inference algorithms for general graphical models like the junction tree algorithm but, depends on the complexity of graph the inference procedure needs exponential time [Sutton & McCallum 2010]. Apart from some standard graph topology like linear-chain that a standard inference algorithm works well, doing exact inference for complex graph should resort with approximation algorithm. The Monte Carlo algorithms and the Variational method algorithms are two families of computational methods for this purpose.

With this brief introduction to CRF model, in continue we concentrate on using the CRF model for image segmentation task. As mentioned before defining a CRF model that has been expressed by a complete graph on every pixel in the image consists billions of edge connections between the nodes in its graph. Doing inference in such a dense graph by using MCMC method is not computationally feasible. In an example that demonstrated by Krähenbühl and Koltun, doing inference for a fully connected CRF by using the MCMC method took 36 hours for partial convergence of the algorithm for an image with a common resolution [Krähenbühl & Koltun, 2012].

The Mean Field Approximation is a popular algorithm among Variational methods in inference and data modeling. In this algorithm, instead of computing the exact intractable distribution P(**X**) a substitute distribution Q(**X**) by minimizing the KL-divergence *D*(Q||P) among all distributions Q is computed which Q(**X**) is usually expressed as a factorial distribution:

$$Q(\boldsymbol{X}) = \prod_i Q_i(X_i)$$

In [Krähenbühl & Koltun, 2012] shown that minimizing the KL-divergence will obtain the below approximate distribution:

$$Q_i(x_i = l) = \frac{1}{Z_i}\exp\left[-\psi_u(x_i) - \sum_{l'\in L}\mu(l,l')\sum\nolimits_{m=1}^{K} w^{(m)}\sum_{j\neq i} k^{(m)}\left(f_i, f_j\right)Q_j(l')\right] \quad (17)$$

then the iterative updating of Mean field approximation algorithm would be as below:

```
Step 1: Initialize Q just by using unary potential
        $Q_i(x_i) = \frac{1}{Z_i}\exp[-\psi_u(x_i)]$
Step 2: Message passing for all $X_j$ to all $X_i$
        $\tilde{Q}_i^{(m)}(l) = \sum_{i\neq j} k^{(m)}\left(f_i, f_j\right).Q_i(l),\ m = 1,..,K$
Step 3: Compatibility transform
        $\hat{Q}_i(x_i) = \sum_{l\in L}\mu^{(m)}(x_i, l).\sum_m w^{(m)}\tilde{Q}_i^{(m)}(l)$
Step 4: Local update and normalizing $Q_i(x_i)$ distribution
        $Q_i(x_i) = \exp[-\psi_u(x_i) - \hat{Q}_i(x_i)]$
```

Step 5: If not converged, return to Step 2

The details of convergence and efficient computational considerations for message passing step was discussed in [Krähenbühl & Koltun, 2012].

Although according to this approach and replacing the pairwise edge potentials of CRF model with a linear combination of Gaussian kernels has advantages but there are also some shortcomings. For example the widths of Gaussian kernels (σ) are constant value that are adjusted by evaluating the performance of algorithm on the validation set with a brute force method such as grid search. Since kernel width controls the impact of other pixels on one pixel according to their spatial distance (for spatial kernel) and the appearance similarity distance (for appearance kernel), this kernel width cannot be adjusted optimally for all diverse semantic objects in the image. For example in urban scene images the *road* region usually expands in a large area of the image while the *traffic sign* often has a small compact area. Using a same spatial kernel width for both of these two objects causes misleading in the case of large kernel width for traffic sign object or it causes missing the connection between pixels in case of small kernel width for road object. The current approach is based on using a brute force algorithm like grid search and monitoring the average IoU score on a defined validation set for finding a set of suboptimum values including the Gaussian kernel widths for CRF energy function. Further studies, which take these variables into account, will need to be undertaken.

References

[Baker et al. 2011] Baker, S., Scharstein, D., Lewis, J. P., Roth, S., Black, M. J., & Szeliski, R. (2011). A database and evaluation methodology for optical flow. International Journal of Computer Vision, 92(1), 1-31.

[Horn & Schunck 1981] Horn, B. K., & Schunck, B. G. (1981). Determining optical flow. Artificial intelligence, 17(1-3), 185-203.

[Settles 2004] Settles, B. (2004, August). Biomedical named entity recognition using conditional random fields and rich feature sets. In Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications (pp. 104-107). Association for Computational Linguistics.

[Sha & Pereira 2003] Sha, F., & Pereira, F. (2003, May). Shallow parsing with conditional random fields. In Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1 (pp. 134-141). Association for Computational Linguistics.

[Smith & Osborne 2005] Smith, A., & Osborne, M. (2005, October). Regularization techniques for conditional random fields: Parameterized versus parameter-free. In International Conference on Natural Language Processing (pp. 896-907). Springer Berlin Heidelberg.

[Stern et al 2004] Stern, D. H., Graepel, T., & MacKay, D. (2004). Modelling uncertainty in the game of Go. In Advances in neural information processing systems (pp. 1353-1360).

[Sutton & McCallum 2010] Sutton, C., & McCallum, A. (2010). An introduction to conditional random fields. arXiv preprint arXiv:1011.4088.

[Zweig & Nguyen 2009] Zweig, G., & Nguyen, P. (2009, November). A segmental CRF approach to large vocabulary continuous speech recognition. In Automatic Speech Recognition & Understanding, 2009. ASRU 2009. IEEE Workshop on (pp. 152-157). IEEE.