Feature Space Optimization for Semantic Video Segmentation

Abhijit Kundu* Georgia Tech
Vibhav Vineet* Intel Labs
Vladlen Koltun Intel Labs

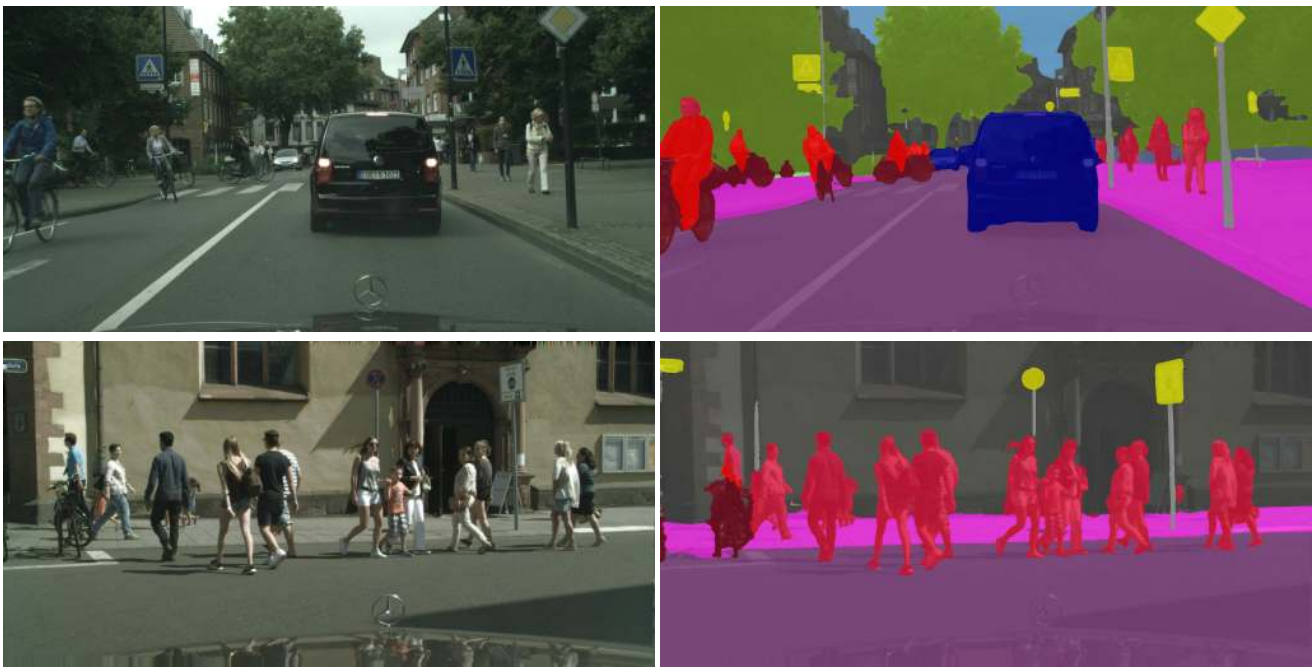

Figure 1. Semantic video segmentation on the Cityscapes dataset [6]. Input frame on the left, semantic segmentation computed by our approach on the right.

Abstract

We present an approach to long-range spatio-temporal regularization in semantic video segmentation. Temporal regularization in video is challenging because both the camera and the scene may be in motion. Thus Euclidean distance in the space-time volume is not a good proxy for correspondence. We optimize the mapping of pixels to a Euclidean feature space so as to minimize distances between corresponding points. Structured prediction is performed by a dense CRF that operates on the optimized features. Experimental results demonstrate that the presented approach increases the accuracy and temporal consistency of semantic video segmentation.

1. Introduction

Structured prediction has become a standard means of achieving maximal accuracy in semantic segmentation. In structured prediction, all pixels are labeled jointly and labeling coherence is explicitly enforced. This alleviates the noise and inconsistency that can arise when pixels are classified independently. In particular, the fully-connected CRF [10, 11] – also known as the dense CRF – often yields significant improvements in semantic segmentation accuracy. For example, after the recent breakthrough of Long et al. [18], who developed a new model for semantic image segmentation, an application of the dense CRF over the new model yielded substantial accuracy gains [4, 15, 31].

The natural form of input for vision systems that operate in the physical world is video. For this reason, we consider semantic segmentation of video sequences, rather than individual images. In a typical video sequence, each frame depicts a different view of the scene. Thus structured prediction can be used not only for spatial regularization within

*Joint first authors

individual frames but also for temporal consistency across frames. In this paper, we address the challenges brought up by such spatio-temporal regularization.

Long-range temporal regularization in video is complicated by the fact that both the camera and the scene may be in motion. In particular, camera motion can induce significant optical flow across the visual field. For example, when the camera rotates, a point in the scene can quickly translate across the image plane. For this reason, simply appending the time dimension to the feature space used for regularization can lead to incorrect associations and cause misprediction in the presence of significant camera and object motion. The underlying problem is that Euclidean distance in the space-time video volume is not a good proxy for correspondence.

Our solution is to optimize the feature space used by the dense CRF so that distances between features associated with corresponding points in the scene are minimized. The dense CRF operates on an embedding of the pixels into a Euclidean feature space [10]. The Euclidean norm in this space is used to define a continuous measure of correspondence. All pairs of pixels are connected and all pairs of pixels are regularized. In our setting, the regularization is performed over a fully-connected graph over the video volume. The strength of the connection between a pair of pixels is a function of their distance in the feature space. Our approach optimizes the feature space embedding such that Euclidean distance in feature space is a more accurate measure of correspondence in the underlying scene.

Specifically, we establish temporal correspondences via optical flow and long-term tracks and optimize the feature space embedding to minimize distances between corresponding points, subject to second-order regularization constraints. We express the embedding objective as a linear least-squares problem and show that feature space optimization can be performed efficiently over high-resolution video volumes. The resulting embedding is used by a fully-connected space-time CRF that performs direct long-range regularization across the video volume, while operating at full resolution and producing sharp pixel-level boundaries.

We evaluate the proposed semantic video segmentation approach through extensive experiments on the CamVid and Cityscapes datasets [2, 6]. Experimental results demonstrate that feature space optimization increases the accuracy of semantic video segmentation. Our approach yields a 66.1% mean IoU on CamVid and a 70.3% mean IoU on the Cityscapes validation set. Both results are the highest reported to date. In addition, the presented approach substantially increases the temporal consistency of the labeling. This is evaluated quantitatively in our experiments and is also evident in the supplementary video. Figure 1 shows results produced by the presented approach on two frames from the Cityscapes dataset.

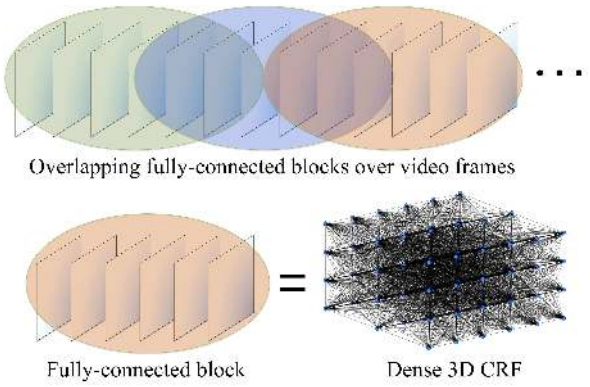

Figure 2. The temporal structure of the model. The video is covered by overlapping blocks. A dense CRF is defined over each block and feature space optimization is performed within blocks. Structured prediction is performed over multiple blocks.

2. Model

Our model is a set of cliques that cover overlapping blocks in the video volume. We cover the video by overlapping temporal blocks, define a dense CRF over each block, and build in provisions for temporally smooth prediction across block boundaries. The temporal structure of the model is illustrated in Figure 2. The block construction is described in Section 5.

Each pixel in the video is identified by a vector $\mathbf{p} = (b, t, i) \in \mathbb{R}^3$, where b is the block number, t is the frame number within block b, and i is the index of the pixel within the frame. The color of pixel $\mathbf{p}$ is denoted by $\mathbf{I_p} \in \mathbb{R}^3$ and the coordinates of pixel $\mathbf{p}$ in its frame are denoted by $\bar{\mathbf{s}}_\mathbf{p} \in \mathbb{R}^2$. Let $\mathbf{P}$ be the set of pixels in the video.

Given pixel $\mathbf{p}$, let $X_\mathbf{p}$ be a random variable with the domain $\mathcal{L} = \{l_1, ..., l_L\}$. The states l_i will be referred to as labels. Let $\mathcal{X}$ be a random field over $\mathbf{P}$ and let $\mathbf{x} : \mathbf{P} \rightarrow \mathcal{L}$ be a label assignment. The random field $\mathcal{X}$ is characterized by a Gibbs distribution $P(\mathbf{x}|\mathbf{P})$ and the corresponding Gibbs energy $E(\mathbf{x}|\mathbf{P})$ associated with each label assignment:

$$\begin{aligned} P(\mathbf{x}|\mathbf{P}) &= \frac{1}{Z(\mathbf{P})} \exp\big(-E(\mathbf{x}|\mathbf{P})\big), \\ E(\mathbf{x}|\mathbf{P}) &= \sum_\mathbf{p} \psi^u_\mathbf{p}(x_\mathbf{p}) + \sum_{(\mathbf{p},\mathbf{q})\in\mathcal{E}} \psi^p_{\mathbf{p},\mathbf{q}}(x_\mathbf{p}, x_\mathbf{q}). \end{aligned} \tag{1}$$

Here $Z(\mathbf{P}) = \sum_\mathbf{x} \exp\big(-E(\mathbf{x}|\mathbf{P})\big)$ is the partition function and $\mathcal{E}$ is a neighborhood structure defined on pairs of variables. The neighborhood structure is a union of cliques: each block is covered by a clique, each pixel is covered by two blocks, and each variable is correspondingly covered by two fully-connected subgraphs in the random field. Our goal is to find a label assignment $\mathbf{x}^*$ that minimizes the Gibbs energy.

The unary term $\psi^u_\mathbf{p}(x_\mathbf{p})$ specifies the cost of assigning

label $x_{\mathbf{p}}$ to pixel $\mathbf{p}$. Pairwise terms $\psi^p_{\mathbf{p},\mathbf{q}}(x_{\mathbf{p}}, x_{\mathbf{q}})$ couple pairs of variables and penalize inconsistent labeling. These terms are defined using Gaussian kernels [10]:

$$\psi^p_{\mathbf{p},\mathbf{q}}(x_{\mathbf{p}}, x_{\mathbf{q}}) = \mu(x_{\mathbf{p}}, x_{\mathbf{q}}) \sum_{m=1}^{M} w^m \kappa^m(\mathbf{f}_{\mathbf{p}}, \mathbf{f}_{\mathbf{q}}), \quad (2)$$

where $\mu(x_{\mathbf{p}}, x_{\mathbf{q}})$ is a label compatibility term and w^m are the mixture weights. $\mathbf{f}_{\mathbf{p}}$ and $\mathbf{f}_{\mathbf{q}}$ are features associated with $x_{\mathbf{p}}$ and $x_{\mathbf{q}}$, respectively. Each kernel has the following form:

$$\kappa^m(\mathbf{f}_{\mathbf{p}}, \mathbf{f}_{\mathbf{q}}) = \exp\left(-\frac{\|\mathbf{f}_{\mathbf{p}} - \mathbf{f}_{\mathbf{q}}\|^2}{\sigma_m^2}\right). \quad (3)$$

Given point $\mathbf{p}$, the feature $\mathbf{f}_{\mathbf{p}} \in \mathbb{R}^D$ is a vector in a D-dimensional feature space. In semantic image segmentation, the canonical feature space is five-dimensional and combines image position and color [10]. A natural feature space for semantic video segmentation is six-dimensional and combines time, color, and position: $\mathbf{f}_{\mathbf{p}} = (t_{\mathbf{p}}, \mathbf{I}_{\mathbf{p}}, \bar{\mathbf{s}}_{\mathbf{p}})$. We will use this feature space as a starting point.

3. Feature Space Optimization

Feature-sensitive models of the kind described in Section 2 have been very successful in semantic image segmentation [10, 31]. However, applying such models to space-time video volumes is not straightforward. A key difficulty is that both the camera and the objects may be in motion and can carry corresponding pixels apart. Thus the natural six-dimensional feature space yields a distance measure that does not appropriately model spatio-temporal correspondence.

A hypothetical solution that would address this issue is to obtain a dense metric 3D reconstruction of the scene through time, associate each pixel with the true 3D position of the corresponding surface element in the environment, and use this 3D position along with time as a feature. This would enforce a coherence assumption on surfaces that are truly proximate in space-time. However, dense monocular 3D reconstruction of dynamic scenes is an open problem. We therefore develop an alternative approach that does not require understanding the three-dimensional layout of the scene.

Our approach involves optimizing a subspace of the feature space to reduce Euclidean distance between corresponding points while adhering to regularization terms that aim to preserve object shapes. Specifically, for all points $\{\mathbf{p}\}$, we optimize position features $\{\mathbf{s}_{\mathbf{p}}\}$. (The time and color dimensions are fixed.) Thus the feature mapping $(t_{\mathbf{p}}, \mathbf{I}_{\mathbf{p}}, \bar{\mathbf{s}}_{\mathbf{p}})$ is replaced by $(t_{\mathbf{p}}, \mathbf{I}_{\mathbf{p}}, \mathbf{s}_{\mathbf{p}})$.

Consider a block b that consists of $T \times N$ points, where T is the number of frames in the block and N is the number of pixels in each frame. The optimization objective is defined as follows:

$$\mathbf{s}^* = \arg\min_{\mathbf{s}} E(\mathbf{s}),$$
$$E(\mathbf{s}) = E_u(\mathbf{s}) + \gamma_1 E_s(\mathbf{s}) + \gamma_2 E_t(\mathbf{s}). \quad (4)$$

Here $\mathbf{s}$ are the position features for all pixels in the block and $\mathbf{s}^*$ are the optimal features. The objective $E(\mathbf{s})$ comprises a data term $E_u(\mathbf{s})$, a spatial regularizer $E_s(\mathbf{s})$, and a temporal regularizer $E_t(\mathbf{s})$. We now explain each of these three terms. We will use $\mathbf{p}$ and (b, t, i) interchangeably to denote a point in the block.

Data term $E(\mathbf{s})$. The data term prevents the feature space embedding from drifting or collapsing under the strength of the regularization terms. The middle frame in the block is used as an anchor. Let $a = \lfloor T/2 \rfloor$ be the frame number of the anchor frame and let $\mathbf{P}^a$ be the set of pixels in frame a. Let $\{\bar{\mathbf{s}}_{\mathbf{p}} : \mathbf{p} \in \mathbf{P}^a\}$ be the unoptimized natural feature space for $\mathbf{P}^a$. The data term ensures that points in the anchor frame do not drift far from their natural positions:

$$E_u = \sum_{\mathbf{p} \in \mathbf{P}^a} \left(\mathbf{s}_{\mathbf{p}} - \bar{\mathbf{s}}_{\mathbf{p}}\right)^2. \quad (5)$$

Spatial regularization term $E_s(\mathbf{s})$. The spatial regularizer preserves shapes within color boundaries and detected contours. We use anisotropic second-order regularization over the 4-connected pixel grid [14, 12]:

$$E_s(\mathbf{s}) = \sum_{t=1}^{T} \sum_{i=1}^{N} \left(\mathbf{s}_{(b,t,i)} - \sum_{j \in \mathcal{N}_i} w_{ij} \mathbf{s}_{(b,t,j)}\right)^2. \quad (6)$$

Here $\mathcal{N}_i$ is the set of neighbors of point (b, t, i). The weight w_{ij} attenuates the regularization at object boundaries:

$$w_{ij} = \exp\left(-\frac{\|\mathbf{I}_{(b,t,i)} - \mathbf{I}_{(b,t,j)}\|^2}{\sigma_1}\right) \exp\left(-\frac{c_{\mathbf{p}}^2}{\sigma_2}\right). \quad (7)$$

The first factor in (7) is based on the color difference between the two pixels and the second factor is based on the contour strength at pixel $\mathbf{p}$. We use structured forests to compute contour strength $c_{\mathbf{p}}$ [7], such that $c_{\mathbf{p}} \in [0, 1]$ and $c_{\mathbf{p}} = 1$ indicates the presence of a boundary.

Temporal regularization term $E_t(\mathbf{s})$. The temporal regularizer pulls corresponding points in different frames to assume similar positions in feature space:

$$E_t(\mathbf{s}) = \sum_{(\mathbf{p},\mathbf{q}) \in \mathcal{K}} \left(\mathbf{s}_{\mathbf{p}} - \mathbf{s}_{\mathbf{q}}\right)^2. \quad (8)$$

This is the term that minimizes distances between corresponding points. $\mathcal{K}$ is a collection of correspondence pairs $(\mathbf{p}, \mathbf{q})$, where $\mathbf{p}$ and $\mathbf{q}$ are in different frames. Correspondences are established via optical flow and long-term tracks, as described in Section 5.

Optimization. Objective (4) is a large-scale linear least-squares problem with second-order regularization. We optimize the objective using the biconjugate gradient stabilized method [29] with algebraic multigrid preconditioning [22].

4. Inference

Efficient inference in the model specified by Equation (1) can be performed by an extension of the mean-field inference algorithm introduced by Krähenbühl and Koltun [10]. Note that our model is a collection of overlapping cliques and is thus different from the fully-connected model considered by Krähenbühl and Koltun.

Define a distribution Q that approximates the true distribution P, where similarity between distributions is measured by the KL-divergence. Assume that Q factorizes over the individual variables: $Q(\mathbf{x}) = \prod_{\mathbf{p}} Q_{\mathbf{p}}(x_{\mathbf{p}})$, where $Q_{\mathbf{p}}$ is a distribution over the random variable $X_{\mathbf{p}}$. The mean-field updates have the following form:

$$\begin{aligned} Q_{\mathbf{p}}(l) &= \frac{1}{Z_{\mathbf{p}}} \exp\Big(-\psi^u_{\mathbf{p}}(l) - S_1(l) - S_2(l)\Big), \\ S_1(l) &= -\sum_{l' \in \mathcal{L}} \sum_{\mathbf{q} \in \mathcal{N}^1_{\mathbf{p}}} Q_{\mathbf{q}}(l') \psi^p_{\mathbf{p},\mathbf{q}}(l, l'), \\ S_2(l) &= -\sum_{l' \in \mathcal{L}} \sum_{\mathbf{q} \in \mathcal{N}^2_{\mathbf{p}}} Q_{\mathbf{q}}(l') \psi^p_{\mathbf{p},\mathbf{q}}(l, l'), \\ Z_{\mathbf{p}} &= \sum_{l} \exp\Big(-\psi^u_{\mathbf{p}}(l) - S_1(l) - S_2(l)\Big), \end{aligned} \tag{9}$$

where $\mathcal{N}^1_{\mathbf{p}}$ and $\mathcal{N}^2_{\mathbf{p}}$ are sets of neighbors of $\mathbf{p}$ in the two blocks that cover $\mathbf{p}$. The updates can be performed efficiently using Gaussian filtering in feature space [10]. Given the Q distribution at the end of the final iteration, a labeling can be obtained by assigning $x^*_{\mathbf{p}} = \arg\max_l Q_{\mathbf{p}}(l)$.

We now consider what happens when the video volume is too large to fit in memory. We can partition the video into chunks of consecutive blocks, such that inference in each chunk is performed separately. To align the predicted distributions across blocks, we could use a distributed optimization strategy such as dual decomposition [9]. However, the convergence of such schemes can be quite slow. We therefore opt for a simple heuristic that has the added benefit that chunks can be processed in a streaming fashion.

Consider two overlapping blocks b^1 and b^2, such that b^1 is the last block in one chunk and b^2 is the first block in the next chunk. Let Q^1 and Q^2 be the distributions produced by mean-field inference for these blocks in their respective chunks. Let $[t_1, t_2]$ be the overlap region. Let Q^t be the sought-after distribution for frame $t \in [t_1, t_2]$ and let $Q^{1,t}$ and $Q^{2,t}$ be the corresponding slices of Q^1 and Q^2. We transition between chunks via simple linear interpolation:

$$Q^t = Q^{1,t} + \frac{t - t_1}{t_2 - t_1} Q^{2,t}. \tag{10}$$

5. Implementation

We use two sets of unary potentials in our experiments. The first is the classical TextonBoost classifier of Shotton et al. [23], as implemented by Ladicky et al. [13]. This classifier was used in a number of prior semantic video segmentation systems and enables a fair comparison to prior work. Second, we use a convolutional network based on the work of Yu and Koltun [30], which we refer to as the Dilation unary. This network consists of a front-end prediction module and a context aggregation module. The front-end module is an adaptation of the VGG-16 network based on dilated convolutions [24, 30]. The context module uses dilated convolutions to systematically expand the receptive field and aggregate contextual information [30]. In combination, the two modules form a high-performing convolutional network for dense prediction. In particular, the Dilation network yielded the highest semantic segmentation accuracy among all models evaluated by Cordts et al. [6], without using structured prediction.

In all experiments, we use optical flow computed by LDOF [3]. To evaluate the influence of the input flow, we also conduct a controlled experiment with Discrete Flow [19]. Long-term tracks are computed using the approach of Sundaram et al. [26]. CRF parameters are optimized using grid search on a subset of the validation set.

The decomposition into blocks can be performed using a fixed block size, such as 100 frames. Our implementation uses a different approach that adapts block boundaries to the content of the video. Specifically, we consider long-term tracks [26] and spawn a new block when more than half of the tracks in the frame were not present at the beginning of the block. This increases the internal coherence of each block.

6. Experiments

We evaluate the presented approach on two datasets for road scene understanding: the CamVid dataset [2] and the Cityscapes dataset [6]. Both datasets provide video input along with pixel-level semantic annotations of selected frames.

6.1. CamVid dataset

We begin by performing experiments on the CamVid dataset. We use the split of Sturgess et al. [25], which has been adopted in a number of prior works. This split partitions the dataset into 367 training images, 100 validation images, and 233 test images. 11 semantic classes are used.

The primary accuracy measure we use is mean IoU (intersection over union). The IoU score for a particular class is defined as $\frac{\text{TP}}{\text{TP+FP+FN}}$, where TP, FP, and FN is the number of true positives, false positives, and false negatives for this class, respectively [8].

We have also evaluated global pixel accuracy, defined as the total fraction of correctly classified pixels. We have ascertained that our approach outperforms the prior work in terms of pixel accuracy. However, this measure is severely biased in favor of large classes, such as "sky" and "road", and discounts small but important classes such as "pole" or "sign". We therefore do not report it and discourage other researchers from using it.

In addition to accuracy evaluation on frames that have ground-truth label maps, we have also evaluated the temporal consistency of the labeling produced by each technique. To this end, we have defined a consistency measure in terms of long-term tracks [26]. A track is said to be consistently labeled if all pixels along the track are assigned the same label. The consistency of a labeling is defined to be the fraction of tracks that are consistently labeled. Note that perfect consistency can be achieved trivially at the expense of accuracy: all pixels in all frames in the video can simply be assigned the same label. However, a combination of high accuracy and high consistency is not easy to achieve and we have found that high consistency does correspond to qualitative stability.

Ablation study. We first perform a controlled study to isolate the effect of feature space optimization on labeling accuracy. The results of this experiment are provided in Table 1. We use the TextonBoost unary [23]. Applying a dense 2D CRF within each frame independently improves both mean IoU and consistency. Applying a dense 3D CRF over the video volume improves both metrics further. Performing feature space optimization as proposed in this paper improves both metrics further still.

	mean IoU	Consistency
TextonBoost unary	47.43	60.88
Dense 2D CRF	51.08	74.37
Dense 3D CRF	53.08	81.68
Our approach	55.23	87.33

Table 1. Ablation study with TextonBoost unaries. Spatio-temporal resularization over the video volume increases both accuracy and consistency. Feature space optimization outperforms the baselines.

Comparison to prior work. We now compare the presented approach against state-of-the-art methods for semantic video segmentation. The first set of baseline methods – SuperParsing [27] and the method of Liu and He [17] – perform semantic video segmentation at the supervoxel level. The second set of methods – Tripathi et al. [28] and Miksik et al. [20] – operate at the pixel level. Tripathi et al. define a dense 3D CRF in the space-time volume, but do not optimize the feature space. Miksik et al. enforce temporal smoothness by other means. Finally, we compare to SegNet, a recent convolutional network that has been evaluated on CamVid [1].

Quantitative results are provided in Table 2. (Quantitative comparison against Miksik et al. [20] is provided separately in supplementary material, since Miksik et al. only provided the results of their approach on a subset of the CamVid test set.) Using the classical TextonBoost unary, our approach achieves an accuracy gain of 8 percentage points over the recent method of Liu and He [16] and an improvement of 2 percentage points over Tripathi et al. [28].

The Dilation network outperforms SegNet by 19 percentage points. Feature space optimization and structured prediction yield a further accuracy gain and a 9 percentage point boost in consistency over the Dilation unary. To assess the sensitivity of feature space optimization to the input optical flow, we provide the results of feature space optimization when the input flow fields are computed by LDOF [3] and Discrete Flow [19], respectively. As shown in Table 2, the performance of the approach is virtually identical in the two conditions.

Qualitative results are provided in Figure 3 and in the supplementary video.

6.2. Cityscapes dataset

Cityscapes is a new dataset for scene understanding in urban environments [6]. The dataset contains 2975 training images, 500 validation images, and 1525 test images. 19 semantic classes are used. We report results on the validation set. Results on the test set will be provided in supplementary material.

The results are reported in Table 3. We compare to the recent Adelaide model, a comprehensive system that integrates convolutional networks and conditional random fields [15]. The Dilation network yields slightly higher accuracy than the Adelaide model. Using the Dilation unary, our approach yields a further gain in accuracy and an improvement of more than 6 percentage points in consistency.

	mean IoU	Consistency
Adelaide [15]	68.6	-
Dilation unary [30]	68.65	88.14
Dilation + Our approach	**70.30**	**94.71**

Table 3. Quantitative results on the Cityscapes validation set.

7. Conclusion

We proposed feature space optimization for spatio-temporal regularization. The key observation is that naive regularization over the video volume does not take camera and object motion into account. To support efficient long-range temporal regularization, we optimize the positions of

	Building	Tree	Sky	Car	Sign	Road	Pedestrian	Fence	Pole	Sidewalk	Bicyclist	mean IoU	Consistency
Without ConvNet													
ALE [13]	73.4	70.2	91.1	64.24	24.4	**91.1**	29.1	31	13.6	72.4	**28.6**	53.59	72.2
SuperParsing [27]	70.4	54.8	83.5	43.3	25.4	83.4	11.6	18.3	5.2	57.4	8.9	42.03	**88.8**
Tripathi et al. [28]	74.2	67.9	91	**66.5**	23.6	90.7	26.2	28.5	16.3	71.9	28.2	53.18	76.8
Liu and He [16]	66.8	66.6	90.1	62.9	21.4	85.8	28	17.8	8.3	63.5	8.5	47.2	77.6
TextonBoost + FSO	**74.4**	**71.8**	**91.6**	64.9	**27.7**	91.0	**33.8**	**34.1**	**16.8**	**73.9**	27.6	**55.2**	87.3
With ConvNet													
SegNet [1]	68.7	52	87	58.5	13.4	86.2	25.3	17.9	16.0	60.5	24.8	46.4	62.5
Dilation [30]	82.6	76.2	89.9	84.0	46.9	92.2	56.3	35.8	**23.4**	75.3	55.5	65.29	79.0
Dilation + FSO – LDOF	**84.0**	**77.2**	**91.3**	**85.7**	49.8	**92.6**	**59.3**	**37.6**	16.9	**76.2**	56.8	66.11	**88.3**
Dilation + FSO – DiscreteFlow	**84.0**	**77.2**	**91.3**	85.6	**49.9**	92.5	59.1	**37.6**	16.9	76.0	**57.2**	**66.12**	**88.3**

Table 2. Quantitative results on the CamVid dataset [2]. This table reports per-class IoU, mean IoU, and temporal consistency. *Top:* comparison to prior work that did not use convolutional networks. Using the classical TextonBoost classifier [23], our approach outperforms the prior work. *Bottom:* comparison to prior work that used convolutional networks and evaluated on the CamVid dataset. Using the Dilation network [30], our approach (FSO) yields the highest accuracy reported on the CamVid dataset to date. The performance of the presented approach is virtually identical when two different optical flow algorithms – LDOF and Discrete Flow – are used to compute the input flow fields.

points in the space so that distances between corresponding points are minimized. Applying a dense random field over this optimized feature space yields state-of-the-art semantic video segmentation accuracy.

The presented approach can directly benefit from more accurate optical flow and more stable and temporally extended point trajectories. We encourage further development of these basic building blocks [5, 21]. More broadly, the presented feature space optimization formulation has significant limitations and more flexible approaches should be explored.

References

[1] V. Badrinarayanan, A. Handa, and R. Cipolla. SegNet: A deep convolutional encoder-decoder architecture for robust semantic pixel-wise labelling. *arXiv:1505.07293*, 2015. 5, 6, 7

[2] G. J. Brostow, J. Fauqueur, and R. Cipolla. Semantic object classes in video: A high-definition ground truth database. *Pattern Recognition Letters*, 30(2), 2009. 2, 4, 6, 7

[3] T. Brox and J. Malik. Large displacement optical flow: descriptor matching in variational motion estimation. *PAMI*, 33(3), 2011. 4, 5

[4] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected CRFs. In *ICLR*, 2015. 1

[5] Q. Chen and V. Koltun. Full flow: Optical flow estimation by global optimization over regular grids. In *CVPR*, 2016. 6

[6] 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 *CVPR*, 2016. 1, 2, 4, 5, 8

[7] P. Dollár and C. L. Zitnick. Fast edge detection using structured forests. *PAMI*, 37(8), 2015. 3

[8] M. Everingham, L. J. V. Gool, C. K. I. Williams, J. M. Winn, and A. Zisserman. The Pascal visual object classes (VOC) challenge. *IJCV*, 88(2), 2010. 4

[9] N. Komodakis, N. Paragios, and G. Tziritas. MRF energy minimization and beyond via dual decomposition. *PAMI*, 33(3), 2011. 4

[10] P. Krähenbühl and V. Koltun. Efficient inference in fully connected CRFs with Gaussian edge potentials. In *NIPS*, 2011. 1, 2, 3, 4

[11] P. Krähenbühl and V. Koltun. Parameter learning and convergent inference for dense random fields. In *ICML*, 2013. 1

[12] D. Krishnan, R. Fattal, and R. Szeliski. Efficient preconditioning of Laplacian matrices for computer graphics. *ACM Transactions on Graphics*, 32(4), 2013. 3

[13] L. Ladicky, C. Russell, P. Kohli, and P. H. S. Torr. Associative hierarchical CRFs for object class image segmentation. In *ICCV*, 2009. 4, 6

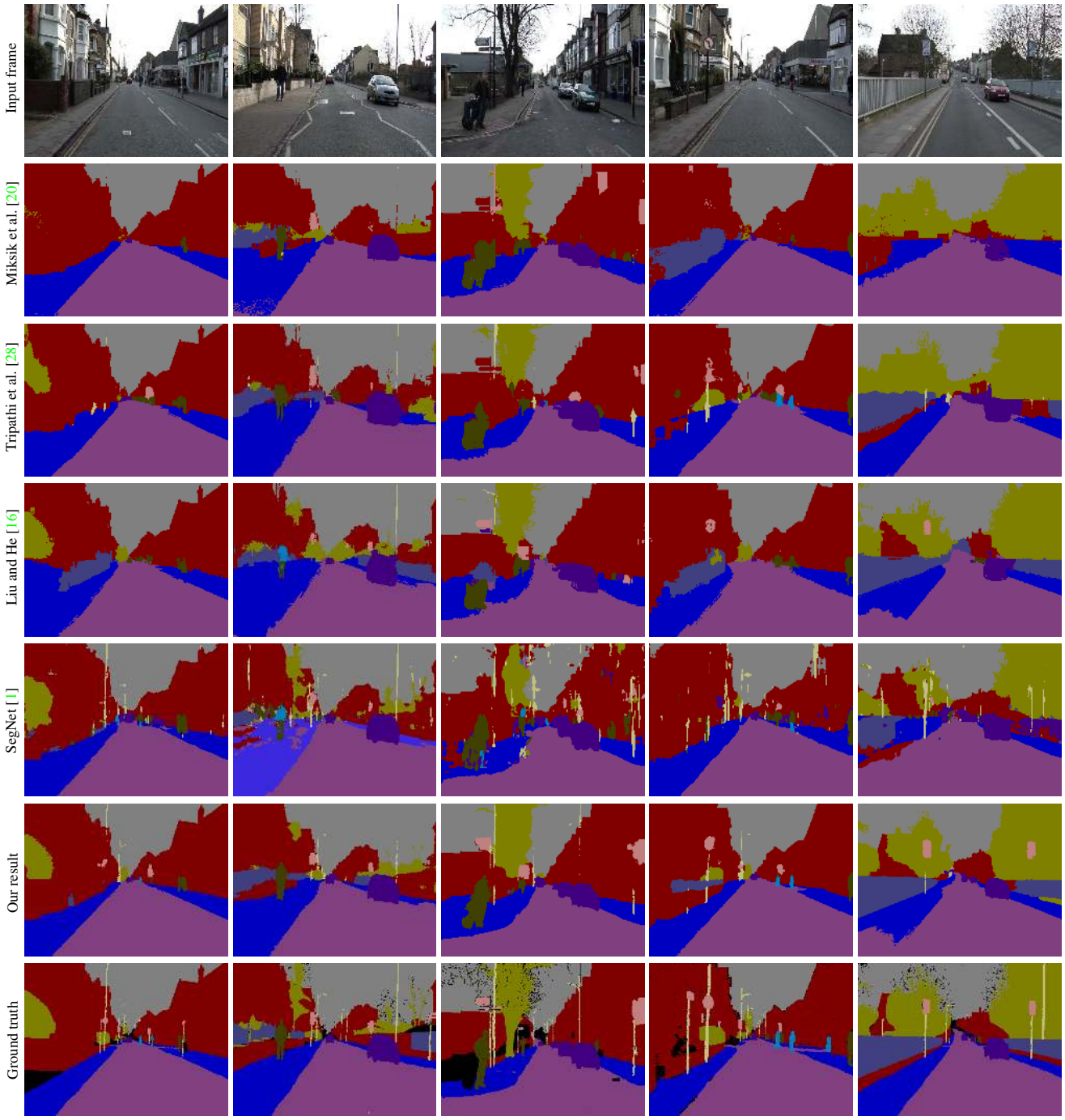

Figure 3. Qualitative results on the CamVid dataset [2]. From top to bottom: input frame, Miksik et al. [20], Tripathi et al. [28], Liu and He [16], SegNet [1], semantic segmentation produced by the presented approach, and ground truth.

[14] A. Levin, D. Lischinski, and Y. Weiss. Colorization using optimization. *ACM Transactions on Graphics*, 23(3), 2004. 3

[15] G. Lin, C. Shen, A. van dan Hengel, and I. Reid. Efficient piecewise training of deep structured models for semantic segmentation. In *CVPR*, 2016. 1, 5

[16] B. Liu and X. He. Multiclass semantic video segmentation with object-level active inference. In *CVPR*, 2015. 5, 6, 7

[17] B. Liu, X. He, and S. Gould. Multi-class semantic video segmentation with exemplar-based object reasoning. In *WACV*, 2015. 5

[18] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*,

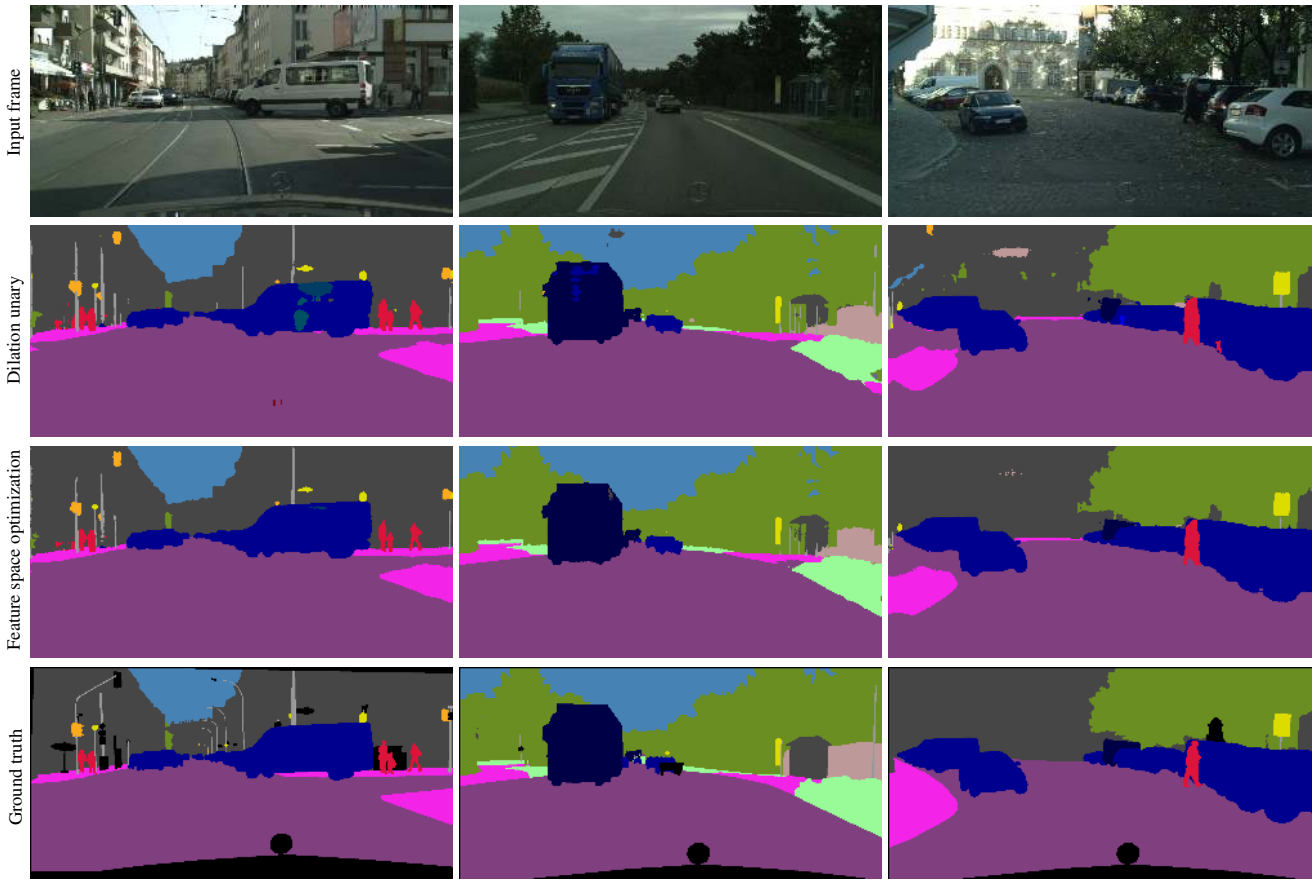

Figure 4. Qualitative results on the Cityscapes dataset [6]. From top to bottom: input frame, output from the Dilation unary [30], semantic segmentation produced by the presented approach, and ground truth.

2015. 1

[19] M. Menze, C. Heipke, and A. Geiger. Discrete optimization for optical flow. In *GCPR*, 2015. 4, 5

[20] O. Miksik, D. Munoz, J. A. Bagnell, and M. Hebert. Efficient temporal consistency for streaming video scene analysis. In *ICRA*, 2013. 5, 7

[21] M. Rubinstein, C. Liu, and W. T. Freeman. Towards longer long-range motion trajectories. In *BMVC*, 2012. 6

[22] J. W. Ruge and K. Stüben. Algebraic multigrid. In *Multigrid Methods*. SIAM, 1987. 4

[23] J. Shotton, J. M. Winn, C. Rother, and A. Criminisi. TextonBoost for image understanding: Multi-class object recognition and segmentation by jointly modeling texture, layout, and context. *IJCV*, 81(1), 2009. 4, 5, 6

[24] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 4

[25] P. Sturgess, K. Alahari, L. Ladicky, and P. H. S. Torr. Combining appearance and structure from motion features for road scene understanding. In *BMVC*, 2009. 4

[26] N. Sundaram, T. Brox, and K. Keutzer. Dense point trajectories by GPU-accelerated large displacement optical flow. In *ECCV*, 2010. 4, 5

[27] J. Tighe and S. Lazebnik. Superparsing – scalable nonparametric image parsing with superpixels. *IJCV*, 101(2), 2013. 5, 6

[28] S. Tripathi, S. Belongie, Y. Hwang, and T. Q. Nguyen. Semantic video segmentation: Exploring inference efficiency. In *ISOCC*, 2015. 5, 6, 7

[29] H. A. van der Vorst. Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of non-symmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 13(2), 1992. 4

[30] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. In *ICLR*, 2016. 4, 5, 6, 8

[31] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. H. S. Torr. Conditional random fields as recurrent neural networks. In *ICCV*, 2015. 1, 3