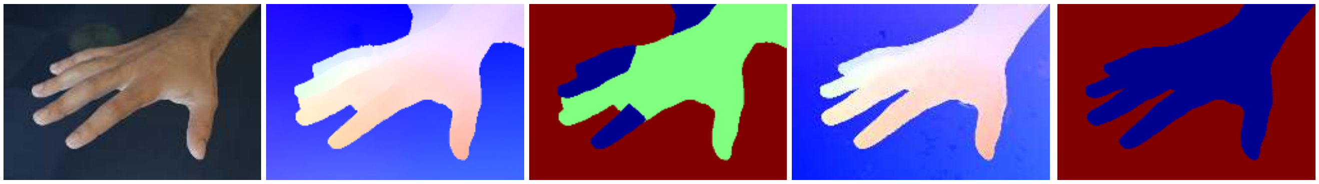
A Fully-Connected Layered Model of Foreground and Background Flow

Deqing Sun[1] Jonas Wulff[2] Erik B. Sudderth[3] Hanspeter Pfister[1] Michael J. Black[2]
[1]Harvard University [2]MPI for Intelligent Systems [3]Brown University



(a) First frame of video (b) Flow field [27] (c) Segmentation [27] (d) Flow field, proposed (e) Segmentation, proposed

Figure 1. The proposed fully-connected layered model can recover fine structures better than a locally connected layered model [27].

Abstract

Layered models allow scene segmentation and motion estimation to be formulated together and to inform one another. Traditional layered motion methods, however, employ fairly weak models of scene structure, relying on locally connected Ising/Potts models which have limited ability to capture long-range correlations in natural scenes. To address this, we formulate a fully-connected layered model that enables global reasoning about the complicated segmentations of real objects. Optimization with fully-connected graphical models is challenging, and our inference algorithm leverages recent work on efficient mean field updates for fully-connected conditional random fields. These methods can be implemented efficiently using high-dimensional Gaussian filtering. We combine these ideas with a layered flow model, and find that the long-range connections greatly improve segmentation into figure-ground layers when compared with locally connected MRF models. Experiments on several benchmark datasets show that the method can recover fine structures and large occlusion regions, with good flow accuracy and much lower computational cost than previous locally-connected layered models.

1. Introduction

Layered models [8, 12, 30] are promising for motion analysis, particularly for handling occlusion and capturing temporally consistent scene structure [27]. Their advantage derives from the fact that they combine motion estimation with segmentation. This allows the integration of image-based information with flow information to arrive at a good segmentation of the scene, which is propagated over time using motion cues. Accurate segmentations, and thus appropriate spatial segmentation and image appearance priors, are key to achieving good performance.

Ising/Potts models produce global segmentations from the interaction of neighboring pixels. While popular in many vision tasks, including layered models [31], such local dependencies have limited modeling power. For example, in the “Hand” sequence of Figure 1, ambiguous local motion and boundary cues cause locally connected layered models [27] to merge the background between the fingers into the foreground (Figure 1(b)).

The problem becomes easier with the more global view provided by a *fully-connected* model that captures pairwise interactions among every pair of pixels. By linking the narrow background regions between the fingers to other distant background regions, it becomes far easier to correctly segment foreground objects. Though appealing in modeling power, these fully-connected priors are difficult to optimize. In this case, neither gradient-based methods [10] nor graph cuts are computationally efficient [21]. Fortunately, Krähenbühl and Koltun [15] recently showed that mean field approximation algorithms can efficiently optimize densely connected CRF models for static image segmentation. Message update steps are efficiently implemented via bilateral filtering. Recent work applies their optimization scheme to optical flow [16] by directly modeling the flow field with a densely connected CRF.

For optical flow estimation, we argue that it is more powerful to utilize a fully-connected prior for layer segmentation. To that end, we formulate a new model that combines recent work on layered flow estimation with algorithms for static image segmentation with fully-connected models. The resulting objective function effectively combines information about motion, occlusion, image appearance, and time to estimate a temporally consistent segmentation of the scene and its evolution. Here we focus on a two-layer model and produce figure-ground segmentations, but our method could be readily extended to more layers. We exploit high-dimensional Gaussian filtering to implement the spatial message passing. Because flow fields are not directly observed, optimization for fully-connected layered models is more challenging than for static image segmentation models, and we proposal several innovations to improve speed and accuracy. In spite of modeling additional dependencies, our resulting mean field method is more efficient than previous locally connected formulations [27].

Our key contributions are 1) a new formulation of the layered optical flow problem that exploits a fully-connected model to achieve better segmentation and that effectively couples image and motion segmentation; 2) a mean field approximation that leverages recent work on image segmentation; 3) an objective function that effectively models occlusions to enable figure-ground segmentation; 4) a precise segmentation of the scene into regions of foreground and background; 5) new schemes for optimizing fully-connected models for layered motion estimation; 6) competitive results on benchmark datasets with a relatively low computational cost compared to locally-connected layered models.

2. Related Work

Several previous and current lines of research intersect our theme, including figure-ground segmentation, video segmentation, and layered optical flow estimation. A review of these broad fields is beyond our scope. Here we focus on layered models, and specifically ones that combine motion estimation and segmentation.

While our 2-layer method is related to figure-ground segmentation, previous work in that area typically uses motion only as a cue to segmentation [7], treats it as an observation [2], and does not attempt to accurately estimate optical flow. In contrast, our work attempts to simultaneously estimate accurate flow and solve for a figure-ground segmentation that gives good flow estimates. Our work is also completely automatic, unlike much of the work on video matting, which usually requires the user to provide a rough segmentation in the form of a trimap. Furthermore, our method can be extended beyond figure-ground to model more layers.

Recent work by Ochs and Brox [22] addresses motion segmentation using point trajectories. They show that higher-order tuples (3-affinities) and segmentation using spectral clustering can produce nice segmentations of scenes with several moving objects. Unlike our work, they do not address the dense flow estimation problem, and consequently do not test on competitive flow benchmarks.

Unger *et al*. [29] can handle scenes with hundreds of labels and handle occlusions by an outlier process without geometric modeling. The estimated motion is not very accurate as evaluated on the Middlebury benchmark.

Our work is directly descended from layered models of optical flow [12, 30, 31]. Several methods extract layered models of the scene as well as layer movement, using parametric models of the motion of each layer. Jojic and Frey [13] extract segmented regions and reason about their depth order, but focus on simple translational motions and do not provide a segmentation of the scene. Kumar *et al*. [18] address a similar problem but exploit graph cuts for optimization and assume a piecewise parametric model of the scene using a purely local MRF model.

Accurate flow estimation *and* segmentation requires richer models of flow within layers that go beyond simple parametric transformations. Previous methods [27, 31] allow the flow to vary smoothly or discontinuously within layers. Recent work [27] shows that such models can achieve good flow and segmentation accuracy, albeit with high computational cost. Layered representations are more complicated than commonly used MRF models, and often use a sequence of random fields/functions [26, 27] to model depth order and occlusions. Inference is thus more challenging. A limitation of these previous methods is that the spatial variation in the flow is modeled by a local (typically pairwise, Ising or Potts) MRF.

Graph cuts (GC) [4] and belief propagation (BP) [17] are popular optimizers for Ising/Potts models and can find better local optima than local search methods [28]. However, local MRF models are fundamentally limited in their expressive power. One way to go beyond local MRFs is to use higher-order potentials [24], but these models are difficult to optimize using GC [14] or BP [19]. Another way to add long-range interactions is to densely connect distant pixels as in the non-local means denoising methods [5]. Such methods do not address layers, segmentation or flow.

Here we build on recent fully-connected models in which every pixel is connected to every other pixel. Krähenbühl and Koltun [15] describe a mean-field approximate inference scheme for fully connected CRF models. They show that the spatial message passing step can be efficiently approximated by high-dimensional filtering [1]. Zhang and Chen [33] independently suggest a quadratic programming relaxation for fully connected CRFs, and use bilateral filtering to perform gradient descent. Our main contribution is to extend these fully-connected inference methods to layered models for optical flow estimation and segmentation.

3. Fully Connected Modeling and Inference

We first formulate our fully connected layered model, and then describe a *variational expectation maximization* (EM) inference algorithm.

3.1. A Fully Connected Layered Flow Model

Given a sequence of images $\mathbf{I}_t, 1 \leq t \leq T$, we seek to decompose the scene into foreground ($k=1$) and background ($k=2$) layers. We use the terms foreground and background loosely; the foreground layer is one that contains regions occluding the background. Generally, multiple moving objects that do not mutually occlude each other will appear in the foreground layer. A multi-layer formulation [26] can lead to semantically more meaningful segmentations of the scene but is beyond the scope of this paper. Experimentally we find that complex scenes can be surprisingly well approximated by a two-layer model.

Each layer k has its own flow field $(\mathbf{u}_{tk}, \mathbf{v}_{tk})$. We use a semi-parametric flow model [27] that biases the flow within each layer to be piecewise smooth, and roughly similar to a global affine motion. For the horizontal flow field, $\mathbf{u}_{tk}$, we define the spatial energy term, $E_{\text{mrf}}(\mathbf{u}_{tk}, \theta_{tk}) =$

$$\sum_{(p,q)\in\mathcal{E}_{\text{mrf}}} \rho_{\text{mrf}}\left(u^p_{tk}-u^q_{tk}\right)+\lambda_{\text{aff}}\sum_p \rho_{\text{aff}}(u^p_{tk}-u^p_{\theta_{tk}}), \qquad (1)$$

where p and q are pixel indices, $\mathcal{E}_{\text{mrf}}$ contains spatial edges connecting the four nearest pixels, $\rho(\cdot)$ is a robust penalty function, λ_{aff} is a weight, and $\mathbf{u}_{\theta_{tk}}$ is the horizontal component of an affine motion within the layer. The energy function for $\mathbf{v}_{tk}$, the vertical flow field, is defined similarly.

We use a binary mask $\mathbf{g}_t$ to model the foreground support at frame t. Pixels that are not visible in the foreground belong to the background layer. As shown in Figure 2, we model the binary mask spatially as a fully connected CRF and define the spatial energy term

$$E_{\text{space}}(\mathbf{g}_t)=\sum_p\sum_{q\neq p} w^p_q \delta(g^p_t \neq g^q_t), \qquad (2)$$

where a pixel is fully connected to all other pixels at the current frame, $\delta(x)$ is 1 if x is true and 0 otherwise, and the weight w^p_q is defined as

$$\begin{aligned} w^p_q &= \eta G_1(I^p_t - I^q_t, p-q) + (1-\eta) G_2(p-q) = \qquad (3) \\ &\eta \exp\left\{-\frac{\|I^p_t - I^q_t\|^2}{\sigma_I^2} - \frac{\|p-q\|^2}{\sigma_s^2}\right\} + (1-\eta)\exp\left\{-\frac{\|p-q\|^2}{\sigma_{s'}^2}\right\}, \end{aligned}$$

where σ_I, σ_s, and $\sigma_{s'}$ are the standard deviations for the Gaussian kernels, and $\eta \in [0,1]$ weights their relative importance. The first (color) term encourages pixels with similar colors, at moderate distances, to lie within the same layer. The second (spatial) term discourages small, isolated regions. Such small regions produce distracting segmentation artifacts [15]. For our layered flow model, they further lead to inaccurate flow estimates; removing these isolated regions significantly reduces outliers in our final results.

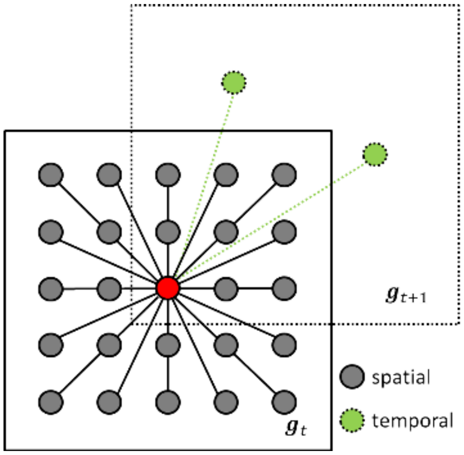


Figure 2. Spatial-temporal neighborhood structure for the binary mask defining foreground layer support. The center pixel (red) is spatially fully connected to all other pixels at the current frame. The center pixel is also temporally connected to two temporal neighbors (green), as determined by the foreground and background flow vectors. The pair of temporal neighbors are only shown for the next frame; the previous is omitted for clarity.

The binary layer support masks evolve over time according to the flow field of the foreground layer,

$$E_{\text{time}}(\mathbf{g}_t, \mathbf{g}_{t+1}, \mathbf{u}_{t1}, \mathbf{v}_{t1}) = \sum_{(p,q)\in\mathcal{E}_{t1}} \delta(g^p_t \neq g^q_{t+1}), \qquad (4)$$

where $\mathcal{E}_{t1} = \{(p,q) : q = p + (u^p_{t1}, v^p_{t1})\}$ contains all temporal neighbors linked by the foreground flow field. As discussed in more detail in Sec. 3.2, we handle subpixel motion by bilinear interpolation of the temporal neighbors.

The layer support mask provides a segmentation of the video sequence: a pixel p belongs to the foreground layer if $g^p_t = 1$, and to the background otherwise. We can then reason about occlusions and modulate the data matching term accordingly, so that at frame t we have

$$E_{\text{data}}(\mathbf{g}_t, \mathbf{g}_{t+1}, \mathbf{u}_t, \mathbf{v}_t) = \sum_{k=1}^{2}\sum_{(p,q)\in\mathcal{E}_{tk}} \phi^k_{\text{data}}(g^p_t, g^q_{t+1}), \qquad (5)$$

where the negative log-likelihoods for the two layers are

$$\phi^1_{\text{data}}(g^p_t, g^q_{t+1}) = \left(\rho_D(I^p_t - I^q_t) - \lambda_D\right) g^p_t g^q_{t+1}, \qquad (6)$$

$$\phi^2_{\text{data}}(g^p_t, g^q_{t+1}) = \left(\rho_D(I^p_t - I^q_t) - \lambda_D\right) \bar{g}^p_t \bar{g}^q_{t+1}. \qquad (7)$$

Here, $\rho_D(\cdot)$ is a robust penalty and $\bar{g} = 1 - g$. The foreground term is only "on" when a pixel and its successor at the next frame are both visible, $g^p_t = g^q_{t+1} = 1$. The background term is active when $g^p_t = g^q_{t+1} = 0$. The occlusion penalty $\lambda_D > 0$ can be derived by assigning a uniform outlier distribution to occluded (unmatched) pixels [26]. Note that occluded states are less likely than flow vectors whose robust matching costs are smaller than λ_D.

$$E(\mathbf{g},\mathbf{u},\mathbf{v},\theta)=\sum_{t=1}^{T-1}\Big\{E_{\text{data}}(\mathbf{g}_t,\mathbf{g}_{t+1},\mathbf{u}_t,\mathbf{v}_t)+\lambda_a\sum_{k=1}^{2}\big(E_{\text{mrf}}(\mathbf{u}_{tk},\theta_{tk})+E_{\text{mrf}}(\mathbf{v}_{tk},\theta_{tk})\big)+\lambda_b E_{\text{space}}(\mathbf{g}_t)$$
$$+\lambda_c E_{\text{time}}(\mathbf{g}_t,\mathbf{g}_{t+1},\mathbf{u}_{t1},\mathbf{v}_{t1})\Big\}+\lambda_b E_{\text{space}}(\mathbf{g}_T) \tag{8}$$

Combining these model potentials over a sequence of T observed frames, we arrive at the overall energy function of Eq. (8). For notational simplicity, we omit dependence on the fixed input images. The energy function is proportional to the negative log probability of the joint distribution of the binary masks and flow fields $P(\mathbf{g},\mathbf{u},\mathbf{v},\theta \mid \mathbf{I})$.

3.2. Inference

We use a variational EM algorithm [9], maximizing the posterior probability of the hidden flow fields while approximately marginalizing over possible layer support masks:

$$\max_{\mathbf{u},\mathbf{v},\theta}\ \log P(\mathbf{u},\mathbf{v},\theta \mid \mathbf{I}) = \max_{\mathbf{u},\mathbf{v},\theta}\ \log \sum_{\mathbf{g}} P(\mathbf{g},\mathbf{u},\mathbf{v},\theta \mid \mathbf{I})$$
$$\geq \max_{\mathbf{u},\mathbf{v},\theta} \sum_{\mathbf{g}} Q(\mathbf{g}) \log \frac{P(\mathbf{g},\mathbf{u},\mathbf{v},\theta|\mathbf{I})}{Q(\mathbf{g})} \tag{9}$$
$$= \min_{\mathbf{u},\mathbf{v},\theta}\ -H(Q) + \sum_{\mathbf{g}} Q(\mathbf{g}) E(\mathbf{g},\mathbf{u},\mathbf{v},\theta) \tag{10}$$

Here, $E(\mathbf{g},\mathbf{u},\mathbf{v},\theta) = -\log P(\mathbf{u},\mathbf{v},\theta \mid \mathbf{I})$ up to some unknown normalization constant. $H(Q)$ is the *entropy* of the variational distribution Q, which for algorithm efficiency is constrained to be fully factorized over both space and time, $Q(\mathbf{g}) = \prod_t \prod_p Q_t^p(g_t^p)$. Given the flow field and marginal approximations at all but one pixel, we can derive the *mean field* update of Eq. (11) via standard methods [9]; see the **Supplemental Material** for details. Alg. 1 summarizes an inference algorithm based on a mean field *message update schedule*. The following sections describe the schemes that make this approach efficient and accurate.

Parallel Spatial Messages. Let $\bar{l} = 1 - l$. At each iteration, a pixel receives messages from all the other pixels in the frame, weighted according to Eq. (3) as

$$\tilde{Q}_t^p(l) = \sum_{q \neq p} w_q^p Q_t^q(\bar{l}) = \sum_q w_q^p Q_t^q(\bar{l}) - Q_t^p(\bar{l}). \tag{12}$$

This is a convolution with a Gaussian kernel in the space and intensity dimensions [15, 23], so $\sum_q w_q^p Q_t^q(\bar{l})$

$$= \sum_q \eta G_1(I_t^p - I_t^q, p-q) Q_t^q(\bar{l}) + (1-\eta) G_2(p-q) Q_t^q(\bar{l})$$
$$= \eta\big[G_1 \otimes Q(\bar{l})\big](I^p, p) + (1-\eta)\big[G_2 \otimes Q(\bar{l})\big](p) \tag{13}$$

This high-dimensional filtering can be efficiently implemented via a permutohedral lattice [1].

Algorithm 1 Mean field for non-local layers

```
Compute $C_{tk}^p = \big[\rho_D\big(I_t^p - I_{t+1}^q\big) - \lambda_d\big]$, $(p,q) \in \mathcal{E}_{tk}$
Initialize $Q_t^p(l) \propto \exp\{-C_{t,2-l}^p\}$
while not converged do
    $Q^{\text{prev}} \leftarrow Q$
    Adjust weight on temporal term $\lambda_c$ as scheduled
    Spatial message passing
        $\tilde{Q}_t^p(l) \leftarrow \lambda_b \sum_{q \neq p} w_q^p Q_t^q(\bar{l})$
    Temporal message passing from next frame
        $\tilde{Q}_t^p(l) \leftarrow \tilde{Q}_t^p(l) + \lambda_c \sum_{(p,q) \in \mathcal{E}_{t1}} Q_{t+1}^q(\bar{l})$
        $\tilde{Q}_t^p(1) \leftarrow \tilde{Q}_t^p(1) + \sum_{(p,q) \in \mathcal{E}_{t1}} C_{t1}^p Q_{t+1}^q(1)$
        $\tilde{Q}_t^p(0) \leftarrow \tilde{Q}_t^p(0) + \sum_{(p,q) \in \mathcal{E}_{t2}} C_{t2}^p Q_{t+1}^q(0)$
    Temporal message passing from previous frame
        $\tilde{Q}_t^p(l) \leftarrow \tilde{Q}_t^p(l) + \lambda_c \sum_{(q,p) \in \mathcal{E}_{t-1,1}} Q_{t-1}^q(\bar{l})$
        $\tilde{Q}_t^p(1) \leftarrow \tilde{Q}_t^p(1) + \sum_{(q,p) \in \mathcal{E}_{t-1,1}} C_{t-1,1}^q Q_{t-1}^q(1)$
        $\tilde{Q}_t^p(0) \leftarrow \tilde{Q}_t^p(0) + \sum_{(q,p) \in \mathcal{E}_{t-1,2}} C_{t-1,2}^q Q_{t-1}^q(0)$
    Exp and normalize
        $Q_t^p(l) \leftarrow \frac{\exp\{-\tilde{Q}_t^p(l)\}}{\exp\{-\tilde{Q}_t^p(0)\}+\exp\{-\tilde{Q}_t^p(1)\}}$
    Damping
        $Q \leftarrow \mu Q + (1-\mu) Q^{\text{prev}}$
    Median filter $Q$ when $\lambda_c$ changes
end while
```

Temporal Messages. Temporal connectivity is more sparse than the non-local spatial model. Each pixel p has two temporal neighbors q at the next frame, determined by the motion of the foreground and the background layers. Its update depends on $\{Q_{t+1}^q : q = p + (u_{tk}^p, v_{tk}^p), k = 1, 2\}$. As real motion is subpixel, we use bilinear interpolation to compute these messages from the four nearest neighbors. Because marginals are positive real numbers, this is straightforward with complexity linear in the frame size.

A pixel, p, may have several temporal neighbors, q, at the previous frame, so that its update depends on marginals $\{Q_{t-1}^q : p = q + (u_{t-1,k}^q, v_{t-1,k}^q), k = 1, 2\}$. We locate these neighbors by inverse warping of the flow field, and complexity remains linear in the number of pixels.

Convergence and Local Optima. To implement spatial message passing via high-dimensional filtering, we must update the node marginals within a frame simultaneously and in parallel [15]. While mean field methods are guaranteed to converge when marginals are updated sequentially [9], they may oscillate with parallel updates as demonstrated in Figure 5. We suspect this is a greater problem for our flow model, where likelihoods are more ambiguous than for

$$Q_t^p(g_t^p) = \frac{1}{Z_t^p} \exp\Bigg\{ -\lambda_b \sum_{q \neq p} w_q^p \mathbb{E}_Q[\delta(g_t^p \neq g_t^q)|g_t^p] - \lambda_c \sum_{(p,q)\in\mathcal{E}_{tk}} \mathbb{E}_Q[\delta(g_t^p \neq g_{t+1}^q)|g_t^p] \tag{11}$$

$$- \sum_{k=1}^{2} \sum_{(p,q)\in\mathcal{E}_{tk}} \mathbb{E}_Q[\phi_{\text{data}}^k(g_t^p, g_{t+1}^q)|g_t^p] - \lambda_c \sum_{(q,p)\in\mathcal{E}_{t-1,k}} \mathbb{E}_Q[\delta(g_{t-1}^q \neq g_t^p)|g_t^p] - \sum_{k=1}^{2} \sum_{(q,p)\in\mathcal{E}_{t-1,k}} \mathbb{E}_Q[\phi_{\text{data}}^k(g_{t-1}^q, g_t^p)|g_t^p] \Bigg\}$$

semantic image segmentation tasks.

We use several approaches to reliably find better local optima. First, we mix the distributions at the current and the previous iterations, similar to damped BP [11]:

$$Q^{\text{new}} = \mu Q^{\text{curr}} + (1-\mu) Q^{\text{prev}} \tag{14}$$

where μ is a stepsize parameter. Second, we start with a small temporal weight λ_c and gradually increase its strength. With strong temporal dependence, it is difficult to deviate from our (temporally consistent) initialization; a weak temporal term allows the algorithm to escape local optima via likelihood cues. We use a piecewise adjustment scheme, in which λ_c is fixed for several iterations before jumping to a series of larger value.

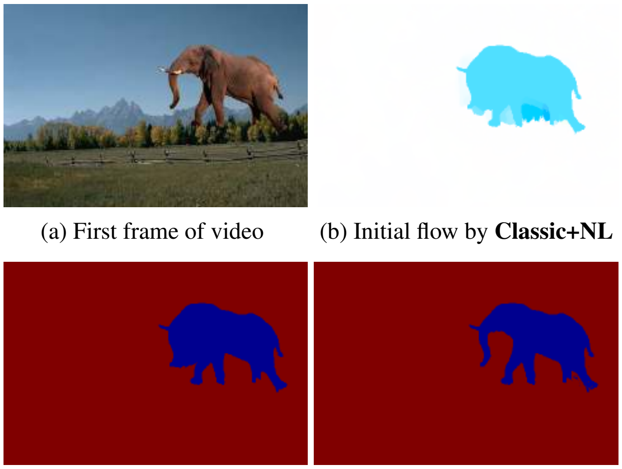
Third, we perform median filtering to the approximate distribution Q whenever there is a change in temporal weight λ_c. This median filtering step helps reduce speckles caused by outliers in the data matching term, and results in better local optima as measured by the K-L divergence between the approximate and true distributions.

Flow Updates. We interleave mean field updates to the layer support distributions with refinement of the foreground and background flow fields. Gradient-based optimization is similar to single-layer affine-biased flow estimation, except that likelihoods are weighted by the inferred layer supports. To avoid local optima, we initialize via a FlowFusion [20] step which combines the current flow estimate and the affine flow field of each layer.

4. Experimental Results

Parameter settings. We hand-tune the parameters using the Middlebury training set: $\sigma_I = 8$, $\sigma_s = 20$, $\lambda_b = 20$, $\lambda_{\text{aff}} = 0.001$, $\sigma_{s2} = 5$, and $\mu = 0.6$. Following [26], we first compute an initial flow estimate using **Classic+NL** [25], and cluster the flow vectors into 2 groups. The method proposed here is applied to four-frame video sequences.

Graph cuts vs mean field. We use the synthetic examples from [27] to compare the effect of using graph cuts for local models and a mean field approximation for fully connected models. As shown in Figure 3, the lack of local evidence and the poor initialization causes the local model to get stuck. The fully connected model recovers from the poor initialization.


(a) First frame of video (b) Initial flow by **Classic+NL**

(c) Local model optimized by graph cuts (d) Fully connected model optimized by mean field

Figure 3. Motion segmentation results on a synthetic sequence. The textureless sky region surrounded by the trunk of the elephant makes it hard for the local model to infer the layer ownership. The fully connected model can more accurately infer the layer ownership by using global information.

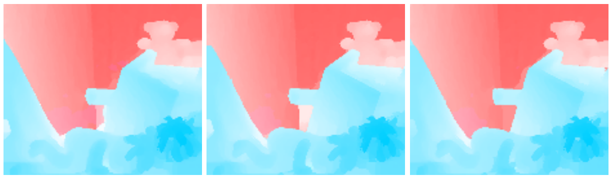

(a) **Classic+NL** (b) **w/o** FlowFusion (c) **w/** FlowFusion

Figure 4. **FlowFusion helps reach better local minima.** .

Effect of FlowFusion. We perform a FlowFusion step to obtain stronger local minima, with some increase in computational cost. As shown in Tables 1 and 2, FlowFusion improves the flow estimates on some sequences. We evaluate the K-L divergence between the approximate distribution Q and the true distribution P using the 12 Middlebury test sequences. The FlowFusion step reduces the K-L divergence on every sequence. The average logarithmic decrease is $2 * 10^5$. We evaluate our algorithm with the FlowFusion step (**FC-2Layers-FF**) and without it (**FC-2Layers**).

Convergence of the mean field algorithm. Figure 5 shows that using the damping scheme helps reach a better local minimum. Figure 6 shows that adjusting the weight on the temporal term and applying median filtering respectively helps the mean field algorithm converge. The algorithm can freely explore the solution space when the temporal weight is small and then converge to a better local minimum. Visu-

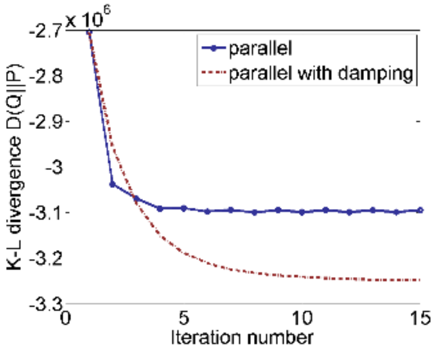
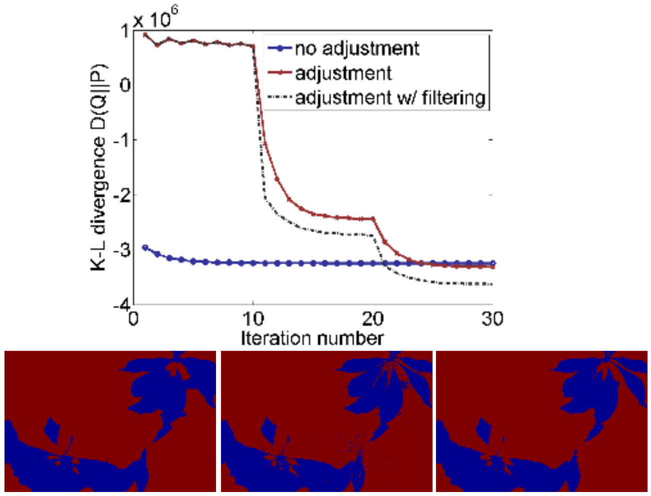


Figure 5. **Damping helps convergence.** The parallel mean field algorithm (blue circle) fails to converge for the fully connected layered model, while damping (red dot) helps the algorithm to converge to a better local minimum.



(a) no adjustment (b) adjustment (c) w/ filtering

Figure 6. **Adjustment and median filtering helps convergence.** They also lead to fewer speckles; see the image in Figure 7. (***better viewed on a computer screen***)

ally, the segmentation has fewer speckles with the adjusting and filtering.

4.1. Benchmark Sequences

Middlebury optical flow. The proposed method, **FC-2Layers-FF**, is ranked 11th on EPE and 7th on AAE in the public table at the time of writing (April 2013). Without the FlowFusion step, the algorithm still obtains reasonable results, as shown in Tables 1 and 2. We perform a bootstrap statistical significance test of the flow estimation results on the Middlebury training and test set for the algorithm with and without the FlowFusion step. The P-Values are 0.9602 and 0.8954, suggesting that the two have similar performance. For practical purposes, we can drop the computationally expensive FlowFusion step and still obtain acceptable results. Our dense two-layer **FC-2Layers-FF** model does not outperform the sparse multi-layer **Layers++** method; we expect future multi-layer formulations to further improve the performance of fully-connected models of layer support.

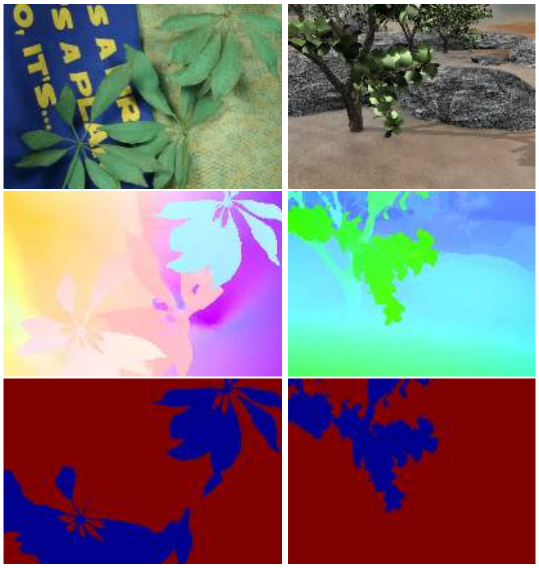


Figure 7. Example results on Middlebury. Top to bottom: First frame; Estimated flow; Segmentation.

MIT layered segmentation. Figures 1 and 9 show some results on the MIT dataset. **FC-2Layers-FF** correctly segments the hand from the background. Compared with a local model, the fully-connected model can recover the background holes between the fingers. **FC-2Layers-FF** also recovers the fine structure of the bicycle in "Car3" (Figure 9). However, there are also failure cases that reveal the limitation of the fully-connected layered model. The reflections on the car and the woman's forehead are misleading color cues for the segmentation. As discussed in [15], long-range connections may propagate wrong information for small regions with different appearance from that of their true layer.

Historical sequences. We also apply **FC-2Layers-FF** to several other sequences, as shown in Figure 8. **FC-2Layers-FF** correctly segments the tree from the background in "flowergarden" and the person from the background in "8_org".

MPI Sintel. We apply **FC-2Layers-FF** to the MPI Sintel dataset [6] using the same parameters tuned on the Middlebury training set. As summarized in Table 3, **FC-2Layers-FF** performs better than **MDP-Flow2** on the more challenging final set. In the unmatched (occlusion) regions, **FC-2Layers-FF** is better on both sets than **MDP-Flow2**. As shown in Figure 10, **FC-2Layers-FF** captures the major occlusions in the scene and the segmentation is consistent with the scene structure. The estimated flow fields are visually close to the ground truth. Note that because the head in "shaman_2" has very different motion from the body in the four frames we used, it is reasonable that **FC-2Layers-FF** separates the head from the body.

Table 1. Average end-point error (EPE) on the Middlebury *training* set. The proposed **FC-2Layers** and **FC-2Layers-FF** methods improve over the single layered **Classic+NL**, while obtaining performance close to a multi-layered formulation.

	Avg.	Venus	Dimetrodon	Hydrangea	RubberWhale	Grove2	Grove3	Urban2	Urban3
Classic+NL	0.221	0.238	0.131	0.152	0.073	0.103	0.468	0.220	0.384
Layers++	0.195	0.211	0.150	0.161	0.067	0.086	0.331	0.210	0.345
FC-2Layers	0.207	0.227	0.145	0.160	0.072	0.096	0.366	0.195	0.395
FC-2Layers-FF	0.205	0.228	0.143	0.155	0.072	0.094	0.362	0.199	0.391
Fast version	0.212	0.227	0.139	0.159	0.077	0.095	0.383	0.214	0.405

Table 2. Average end-point error (EPE) on the Middlebury optical flow benchmark *test* set. The two-layer formulation of the fully-connected layered model achieves performance comparable to a multi-layer local model (**Layers++**).

Fast version uses a fast but less accurate version to compute the initial flow field, which results in slight loss in performance.

		Rank	**Avg.**	Army	Mequon	Schefflera	Wooden	Grove	Urban	Yosemite	Teddy
EPE	**Layers++**	11.5	0.27	0.08	0.19	0.20	0.13	0.48	0.47	0.15	0.46
	FC-2Layers	16.9	0.30	0.08	0.21	0.21	0.15	0.58	0.51	0.16	0.48
	FC-2Layers-FF	13.8	0.28	0.08	0.21	0.20	0.15	0.53	0.49	0.16	0.44

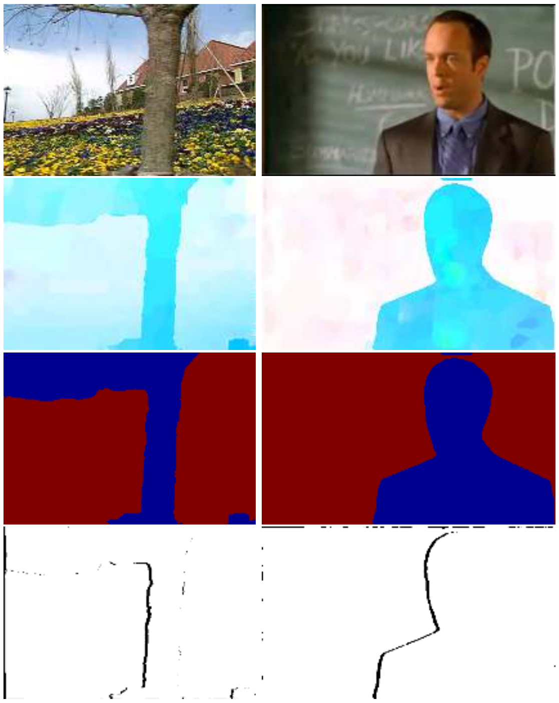


Figure 8. Motion estimation and segmentation results on "flowergarden" and "8_org" [3]. Top to bottom: First frame; Estimated flow; Segmentation; Occlusions.

Table 3. Average end-point error (EPE) on the MPI Sintel *test* set.

	Final		Clean	
	Overall	Unmatched	Overall	Unmatched
FC-2Layers-FF	**8.137**	**39.723**	6.781	**37.144**
MDP-Flow2 [32]	8.445	43.430	**5.837**	38.158
Classic+NL [25]	9.153	44.509	7.961	42.079

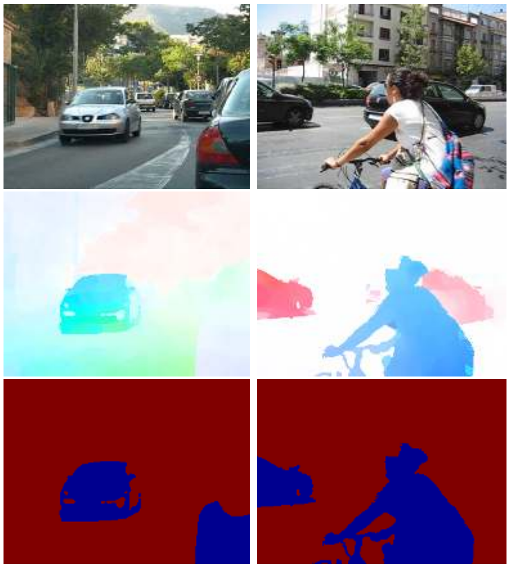
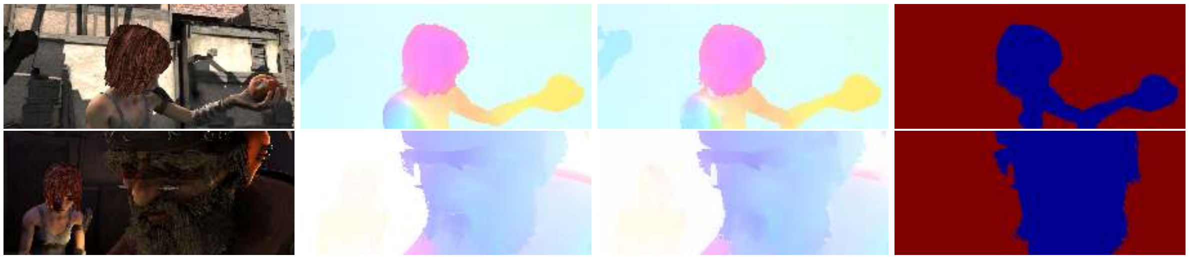


Figure 9. Motion estimation and segmentation results on the MIT benchmark. From top to bottom: First frame of video; Flow estimate; Segmentation. The fully connected layered model can recover the fine structures in the scene, such as the background holes in "Car3" (right column).

Computational time. The computational time for the 4-frame 640 × 480 "Urban" sequence is approximately 40 minutes on a 4GHz Linux desktop computer. The core algorithm for mean field inference takes about 5 minutes. The remaining time is largely spent on computing the initial flow fields with **Classic+NL** [25] in MATLAB. We have developed a fast version of **Classic+NL** by using preconditioned conjugate gradient that reduces the total computational time to about 10 minutes, with slight drop in performance, as shown in Table 1. Further speedup is achievable by using C++ and a GPU flow implementation. Note that the speed is already much faster than previous locally layered models, which take 5 hours to process 2 frames [26] or more than 10 hours for 4 frames [27].

5. Conclusion

We have formulated a fully-connected layered model that captures long-range correlations in natural scenes for joint motion segmentation and estimation. Building on recent successes in static image segmentation, we develop a variational



(a) First image (b) True optical flow (c) Estimated optical flow (d) Segmentation

Figure 10. Results on two sequences of the MPI-Sintel training dataset, “alley_1” (top row) and “shaman_2” (bottom row).

EM algorithm based on high-dimensional filtering. Inference for our fully-connected model is more efficient than algorithms previously used for local models, and fully-connected prior models are particularly effective at recovering fine scene structures and their motion. The proposed algorithm achieves competitive results on the Middlebury and MPI Sintel optical flow benchmark and produces reliable results on a variety of other sequences. Our work extends previous work on fully-connected models for joint motion segmentation and estimation, and also suggests that layered models can be a rich and flexible representation for natural scenes.

Acknowledgement This work has been partially supported by Nvidia, Google, and the Intel Science and Technology Center for Visual Computing.

References

[1] A. Adams, J. Baek, and M. A. Davis. Fast high-dimensional filtering using the permutohedral lattice. *Comput. Graph. Forum*, 29(2):753–762, 2010. 2, 4

[2] X. Bai, J. Wang, and G. Sapiro. Dynamic color flow: A motion-adaptive color model for object segmentation in video. In *ECCV*, pages V:617–630, 2010. 2

[3] X. Bai, J. Wang, D. Simons, and G. Sapiro. Video snapcut: Robust video object cutout using localized classifiers. In *SIGGRAPH*, pages 70:1–70:11, 2009. 6

[4] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *PAMI*, 23(11):1222–1239, Nov 2001. 2

[5] A. Buades, B. Coll, and J. Morel. A non-local algorithm for image denoising. In *CVPR*, pages 2:60–65, 2005. 2

[6] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In *ECCV*, IV, pages 611–625, 2012. 7

[7] A. Criminisi, G. Cross, A. Blake, and V. Kolmogorov. Bilayer segmentation of live video. In *CVPR*, pages 1:53–60, 2006. 2

[8] T. Darrell and A. Pentland. Cooperative robust estimation using layers of support. *PAMI*, 17(5):474–487, 1995. 1

[9] B. Frey and N. Jojic. A comparison of algorithms for inference and learning in probabilistic graphical models. *PAMI*, 27(9):1392–1416, Sept. 2005. 4

[10] G. Gilboa and S. Osher. Nonlocal operators with applications to image processing. *ACM MMS*, 7:1005–1028, 2008. 1

[11] T. Heskes. Stable fixed points of loopy belief propagation are local minima of the bethe free energy. In *NIPS*, pages 343–350, 2002. 5

[12] A. Jepson and M. J. Black. Mixture models for optical flow computation. In *CVPR*, pages 760–761, 1993. 1, 2

[13] N. Jojic and B. Frey. Learning flexible sprites in video layers. In *CVPR*, pages I:199–206, 2001. 2

[14] P. Kohli, L. Ladicky, and P. H. S. Torr. Robust higher order potentials for enforcing label consistency. *IJCV*, 82(3):302–324, 2009. 2

[15] P. Krähenbühl and V. Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *NIPS*, pages 109–117, 2011. 1, 2, 3, 4, 7

[16] P. Krähenbühl and V. Koltun. Efficient nonlocal regularization for optical flow. In *ECCV*, pages I:356–369, 2012. 1

[17] F. Kschischang, B. Frey, and H.-A. Loeliger. Factor graphs and the sum-product algorithm. *IEEE Trans. IT*, 47(2):498–519, Feb 2001. 2

[18] M. Kumar, P. Torr, and A. Zisserman. Learning layered motion segmentations of video. *IJCV*, 76(3):301–319, March 2008. 2

[19] X. Lan, S. Roth, D. P. Huttenlocher, and M. J. Black. Efficient belief propagation with learned higher-order markov random fields. In *ECCV*, pages 269–282, 2006. 2

[20] V. Lempitsky, S. Roth, and C. Rother. FusionFlow: Discrete-continuous optimization for optical flow estimation. In *CVPR*, pages 1–8, 2008. 5

[21] S. Nowozin and C. H. Lampert. Global connectivity potentials for random field models. In *CVPR*, pages 818–825, 2009. 1

[22] P. Ochs and T. Brox. Higher order motion models and spectral clustering. In *CVPR*, pages 614–621, 2012. 2

[23] S. Paris and F. Durand. A fast approximation of the bilateral filter using a signal processing approach. *IJCV*, 81(1):24–52, Jan. 2009. 4

[24] S. Roth and M. J. Black. Fields of experts. *IJCV*, 82(2):205–229, Apr. 2009. 2

[25] D. Sun, S. Roth, and M. J. Black. Secrets of optical flow estimation and their principles. In *CVPR*, pages 2432–2439, 2010. 5, 7

[26] D. Sun, E. B. Sudderth, and M. J. Black. Layered image motion with explicit occlusions, temporal consistency, and depth ordering. In *NIPS*, pages 2226–2234, 2010. 2, 3, 5, 7

[27] D. Sun, E. B. Sudderth, and M. J. Black. Layered segmentation and optical flow estimation over time. In *CVPR*, pages 1768–1775, 2012. 1, 2, 3, 5, 7

[28] R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. F. Tappen, and C. Rother. A comparative study of energy minimization methods for markov random fields with smoothness-based priors. *PAMI*, 30(6):1068–1080, 2008. 2

[29] M. Unger, M. Werlberger, T. Pock, and H. Bischof. Joint motion estimation and segmentation of complex scenes with label costs and occlusion modeling. In *CVPR*, pages 1878–1885, 2012. 2

[30] J. Y. A. Wang and E. H. Adelson. Representing moving images with layers. *IEEE Trans. IP*, 3(5):625–638, Sept. 1994. 1, 2

[31] Y. Weiss and E. Adelson. A unified mixture framework for motion segmentation: Incorporating spatial coherence and estimating the number of models. In *CVPR*, pages 321–326, 1996. 1, 2

[32] L. Xu, J. Jia, and Y. Matsushita. Motion detail preserving optical flow estimation. *PAMI*, 34(9):1744–1757, 2012. 7

[33] Y. Zhang and T. Chen. Efficient inference for fully-connected crfs with stationarity. In *CVPR*, pages 582–589, 2012. 2