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

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

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

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

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

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 λ_c as scheduled
  Spatial message passing
    $\tilde{Q}_t^p(l)\leftarrow\lambda_b\sum_{q\neq p}w_q^pQ_t^q(\bar{l})$
  Temporal message passing from next frame
    $\tilde{Q}_t^p(l)\leftarrow\tilde{Q}_t^p(l)+\lambda_c\sum_{(p,q)\in\mathcal{E}_{t1}}Q_{t+1}^q(\bar{l})$
    $\tilde{Q}_t^p(1)\leftarrow\tilde{Q}_t^p(1)+\sum_{(p,q)\in\mathcal{E}_{t1}}C_{t1}^pQ_{t+1}^q(1)$
    $\tilde{Q}_t^p(0)\leftarrow\tilde{Q}_t^p(0)+\sum_{(p,q)\in\mathcal{E}_{t2}}C_{t2}^pQ_{t+1}^q(0)$
  Temporal message passing from previous frame
    $\tilde{Q}_t^p(l)\leftarrow\tilde{Q}_t^p(l)+\lambda_c\sum_{(q,p)\in\mathcal{E}_{t-1,1}}Q_{t-1}^q(\bar{l})$
    $\tilde{Q}_t^p(1)\leftarrow\tilde{Q}_t^p(1)+\sum_{(q,p)\in\mathcal{E}_{t-1,1}}C_{t-1,1}^qQ_{t-1}^q(1)$
    $\tilde{Q}_t^p(0)\leftarrow\tilde{Q}_t^p(0)+\sum_{(q,p)\in\mathcal{E}_{t-1,2}}C_{t-1,2}^qQ_{t-1}^q(0)$
  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 λ_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