A Generative Model of Dense Optical Flow in Layers

Anitha Kannan[1], Brendan Frey[1], and Nebojsa Jojic[2]

[1] Probabilistic and Statistical Inference Group ,
University of Toronto, Canada
www.psi.utoronto.ca
{frey, anitha}@psi.utoronto.ca
[2] Microsoft Research, Redmond, USA
jojic@microsoft.com

Abstract. We introduce a generative model of dense flow fields within a layered representation of 3-dimensional scenes. Using probabilistic inference and learning techniques (namely, variational methods), we solve the inverse problem and locally segment the foreground from the background, estimate the nonuniform motion of each, and fill in the disocclusions. To illustrate the usefulness of both the representation and the estimation algorithm, we show results on stabilization and frame interpolation that are obtained by generating from the trained models.

1 Introduction

Traditional methods (c.f. [2]) for estimating optical flow assume brightness constancy across frames and a spatially smooth image motion. These methods can be at a disadvantage in a scene with motion discontinuties which arise when multiple objects are move at different velocities. Motion boundaries carry useful information about the objects in the scene, and in fact, when motion boundaries are estimated correctly, a more reliable flow field can be obtained.

We take a generative model approach for estimating dense flow fields of objects within a layered framework. The model assumes that the scene is a composition of objects from multiple layers (for clarity, we present the case with 2 layers- foreground and background), with objects in each layer undergoing a non-uniform motion. The work we present here assumes that the motion is translational, though it is very straight forward to extend to the case of other transformations. The estimation of the flow field and the appearances and alpha masks are done iteratively, using Expectation Maximization, with guaranteed convergence to locally optimal solution.

2 Related Work

An approach to multiple motion problem views layers as components in a mixture model set-up and provides a soft assignment of pixels to layers (or components) [1, 5]. However, these methods do not model the motion boundaries. The approach taken by Wang & Adelson [10] computes motion vectors at each pixel separately [8] and then regularize the motion fields.

In [6], Jojic & Frey take a generative model approach to jointly infer the uniform motion vector and the appearances and the alpha maps for the objects in different layers. The work of Tao et.al. [9] explicitly models depth ordering of layers and occlusion, along with parameterized motion for the foreground.

Black and Fleet [3] modelled the motion discontinuities directly. In their work, the foreground and background are separated by a straight edge within a single, fixed window in the image sequence. The image sequence within the window is modelled by a generative model that predicts the image at time t from the image at time $t-1$, using unknown state variables that describe the location of the edge and the motions of the foreground and background. They use particle filtering to infer the location of the edge, a motion vector for the foreground and a motion vector for the background at each time step.

In contrast, our model [7] explains the entire image and allows the boundary between background and foreground to have any shape. In fact, the model uses a real-valued map to separate foreground from background, so it is capable of modelling semi-transparent patches of objects. The model operates across multiple frames, allowing local patches that are temporarily occluded to be used to fill in "disocclusions" in later frames.

Our work is closely related to and generalizes the work of [6]. Instead of assuming that object within a layer is moving with a uniform motion, we infer

a dense flow field, a motion vector for each pixel, and at the same time learning the segmentation and appearances of objects in different layers.

We present an expectation maximization algorithm for inferring background and foreground motion vectors at every pixel at each time step and at the same time estimating the boundary between the foreground and background.

3 Generative Model

Fig. 1 illustrates our generative model. For each image in the video sequence, our technique estimates an appearance map for the foreground, a transparency map for the foreground, and an appearance map for the background. These maps are estimated from nearby video frames in such a way that every $K \times K$ patch in nearby video frames can be explained by combining a patch from the foreground map with a patch from the background map, using a patch from the transparency map. Here, we show how a fixed patch in 3 video frames is explained by composing patches from the estimated maps.

4 Model of partially occluded patches in motion

Each video frame in turn is considered as a "reference frame". The appearance maps and the transparency map together with the motion fields for foreground and background are estimated using nearby frames. More generally, the maps and motion fields are coupled by a dynamic model. For example, if the maps and motion fields are described by a linear dynamic system, the inference and estimation algorithms presented in later sections are extended to include a Kalman filtering step. For clarity, we focus on estimating the maps and motion fields for a single reference frame.

Let I^t be an $M \times N$ image at time t with pixel intensity $I^t(\mathbf{z})$ at coordinate $\mathbf{z}$. For notational ease, $\mathbf{z}$ is used exclusively as the coordinate in the observed images and $\mathbf{x}$ in the foreground map and the transparency map and $\mathbf{y}$ as the coordinate in the background model.

ϕ is the foreground appearance map and $\phi(\mathbf{x})$ is the foreground appearance at $\mathbf{x}$. α is the transparency map and $\alpha(\mathbf{x})$ is the transparency value in $[0, 1]$ at $\mathbf{x}$. $\alpha(\mathbf{x}) = 1$ indicates the point belongs to the foreground, whereas $\alpha(\mathbf{x}) = 0$ indicates the point belongs to the background. Intermediate values correspond to varying degrees of transparency. β is the background and $\beta(\mathbf{y})$ is the background appearance at $\mathbf{y}$.

In this paper, we use a discrete coordinate system for clarity, although we can easily extend to the case of sub-pixel inference and multi-scale search. To determine the maps and the motion fields, we consider $K \times K$ patches which are translated and compared with patches in nearby frames. We can easily extend to the case where the patches undergo a general affine transformation before comparison with patches in nearby frames. Also, For computational efficiency, we consider motions whereby a patch moves by at most D pixels. However, recent

work of Darabiha et.al. [4] has shown that implementation of vision algorithms on FPGAs can lead to real-time performance.

The sequence of images is the only observation to the model. We require to estimate the appearance and transparency maps and the motion fields, all of them depending on each other for reliable estimation. We treat the appearances and the transparency map as the parameters to the model, and encode the uncertainty in the motion vectors due to initial unreliable parameters of the model by treating motion vectors as random variables.

For pixel $I^t(\mathbf{z})$ in the observed image at time t, the foreground and background motion vectors are represented by random variables $\mathbf{U}^t(\mathbf{z})$ and $\mathbf{V}^t(\mathbf{z})$, respectively. The observed patch at $\mathbf{z}$ is composed from a foreground patch centered at $\mathbf{z}+\mathbf{U}^t(\mathbf{z})$, and a background patch centered at $\mathbf{z}+\mathbf{V}^t(\mathbf{z})$, through the transparency map centered at $\mathbf{z}+\mathbf{U}^t(\mathbf{z})$. Since the motion is limited to D pixels, each of these vectors can take on roughly $(2D)^2$ values.

Each $M \times N$ observed frame is decomposed into an $(M-K+1)\times(N-K+1)$ grid of $K \times K$ overlapping patches. $\mathcal{P}(\mathbf{z})$ denotes the set of coordinates centered at $\mathbf{z}$.

$$\mathcal{P}(\mathbf{z}) = \{\mathbf{w} : |\mathbf{w}-\mathbf{z}| \leq K\}, \tag{1}$$

and $I(\mathcal{P}(\mathbf{z}))$ denotes the set of observed pixel intensities in the patch centered at $\mathbf{z}$.

Given the foreground appearance map, the transparency map and the background appearance map, the patch appearances are assumed to be independent. While this is clearly not true for any sensible interpretation of the maps, this assumption simplifies the model. For patch $\mathcal{P}(\mathbf{z})$ at time t, the observation likelihood for the corresponding motion vectors is

$$P(I^t(\mathcal{P}(\mathbf{z})) \;\; |\mathbf{U}^t(\mathbf{z})=\mathbf{u}, \mathbf{V}^t(\mathbf{z})=\mathbf{v}) \propto \exp\Big[-\sum_{\mathbf{w}\in\mathcal{P}(\mathbf{z})} \frac{(\alpha(\mathbf{w}+\mathbf{u})\phi(\mathbf{w}+\mathbf{u})+\bar{\alpha}(\mathbf{w}+\mathbf{u})\beta(\mathbf{w}+\mathbf{v})-I^t(\mathbf{w}))^2}{2\sigma^2}\Big] \tag{2}$$

where σ^2 is the variance of the sensor noise, and $\bar{\alpha}(\mathbf{x}) = 1-\alpha(\mathbf{x})$ is the inverse transparency map. Under this likelihood function, each observed pixel is equal to a composition of a foreground pixel and a background pixel, plus Gaussian sensor noise. σ^2 may be estimated from the frames or set to a small value.

In this work, we assume the motion vectors are independent and uniform *a priori*. Smaller motions can be favored using, *e.g.*, a Gaussian prior on displacement. The foreground and background motion fields can be separately smoothed using independent random process priors.

The joint distribution over the motion fields in all nearby frames $\mathcal{U}$ and $\mathcal{V}$ and the observed patches in all nearby frames $\mathcal{I}$ is

$$\begin{aligned} P(\mathcal{U},\mathcal{V},\mathcal{I}) &\propto \prod_t \prod_{\mathbf{z}} P(I^t(\mathcal{P}(\mathbf{z}))|\mathbf{U}^t(\mathbf{z})=\mathbf{u},\mathbf{V}^t(\mathbf{z})=\mathbf{v}) \\ &= \prod_t \prod_{\mathbf{z}} \exp\Big[-\sum_{\mathbf{w}\in\mathcal{P}(\mathbf{z})} \frac{(\alpha(\mathbf{w}+\mathbf{u})\phi(\mathbf{w}+\mathbf{u})+\bar{\alpha}(\mathbf{w}+\mathbf{u})\beta(\mathbf{w}+\mathbf{v})-I^t(\mathbf{w}))^2}{2\sigma^2}\Big] \\ &= \exp\Big[-\sum_t \sum_{\mathbf{z}} \sum_{\mathbf{w}\in\mathcal{P}(\mathbf{z})} \frac{(\alpha(\mathbf{w}+\mathbf{u})\phi(\mathbf{w}+\mathbf{u})+\bar{\alpha}(\mathbf{w}+\mathbf{u})\beta(\mathbf{w}+\mathbf{v})-I^t(\mathbf{w}))^2}{2\sigma^2}\Big]. \end{aligned} \tag{3}$$

Given a video sequence, the computational task is to jointly infer the posterior distribution over the motion fields $P(\mathbf{U}^t(\mathbf{z}),\mathbf{V}^t(\mathbf{z})|I^t)$ of nearby frames, and estimate the model parameters ϕ, α and β. For this, we use the expectation maximization algorithm, wherein we alternate between inferring the distribution over plausible motion fields in the expectation step, and estimating the maps in the maximization step. The resulting procedure guarantees convergence to local optimal solution.

5 Motion analysis using the EM algorithm

Initially, the foreground appearance map ϕ and the background appearance map β are unknown. We set them to the average value of the nearby frames and set the transparency map values to 0.5. Starting from the initial maps, the estimation algorithm alternates between inferring the distribution over motion fields in E-step and estimating the maps in the M-step as described below.

E-Step

In the E-Step, for each image I^t, the posterior distribution $P(\mathbf{U}^t(\mathbf{z}),\mathbf{V}^t(\mathbf{z})|I^t)$ over the the foreground motion vector $\mathbf{U}^t(\mathbf{z})$ and the background motion vector $\mathbf{V}^t(\mathbf{z})$ is computed for each coordinate $\mathbf{z}$ in the observed image.

This posterior is computed by examining all possible ways in which the patch centered at $\mathbf{z}$ in I^t can be composed by displacing patches from the foreground and background maps. The posterior distribution is

$$\begin{aligned} P(\mathbf{U}^t(\mathbf{z})=\mathbf{u},\mathbf{V}^t(\mathbf{z})=\mathbf{v}|I^t) = \rho \exp\Big[-\sum_{\mathbf{w}\in\mathcal{P}(\mathbf{z})} \\ \frac{(\alpha(\mathbf{w}+\mathbf{u})\phi(\mathbf{w}+\mathbf{u})+\bar{\alpha}(\mathbf{w}+\mathbf{u})\beta(\mathbf{w}+\mathbf{v})-I^t(\mathbf{w}))^2}{2\sigma^2}\Big], \end{aligned} \tag{4}$$

where ρ ensures that $\sum_{\mathbf{u}}\sum_{\mathbf{v}} P(\mathbf{U}^t(\mathbf{z})=\mathbf{u},\mathbf{V}^t(\mathbf{z})=\mathbf{v}|I^t)=1$.

M-Step

The set of all coordinates whose $K \times K$ patches can "reach" coordinate $\mathbf{x}$ when moved by at most D is

$$\mathcal{R}(\mathbf{x}) = \{\mathbf{z} : |\mathbf{x} - \mathbf{z}| \leq (K-1)/2 + D\}. \tag{5}$$

The set of all motion vectors for the patch at $\mathbf{z}$ that cause a pixel in the patch to be mapped to $\mathbf{x}$ is

$$\mathcal{M}(\mathbf{x}, \mathbf{z}) = \{\mathbf{u} : |(\mathbf{x} - \mathbf{z}) - \mathbf{u}| \leq (K-1)/2; |\mathbf{u}| \leq D\}. \tag{6}$$

Define,

$$\Big\langle \aleph \Big\rangle = \sum_t \sum_{\mathbf{z} \in \mathcal{R}(\mathbf{x})} \sum_{\mathbf{u} \in \mathcal{M}(\mathbf{x},\mathbf{z})} \sum_{\mathbf{v} \in \mathcal{M}(\mathbf{x},\mathbf{z})} P(\mathbf{U}^t(\mathbf{z}) = \mathbf{u}, \mathbf{V}^t(\mathbf{z}) = \mathbf{v} | I^t) \aleph$$

After computing $P(\mathbf{U}^t(\mathbf{z}) = \mathbf{u}, \mathbf{V}^t(\mathbf{z}) = \mathbf{v}|I^t)$ for all t in the E-Step, the foreground appearance is modified:

$$\phi(\mathbf{x}) \leftarrow \left(\Big\langle \alpha(\mathbf{x})^2 \Big\rangle \right)^{-1} \cdot \left(\Big\langle \alpha(\mathbf{x})(I^t(\mathbf{x}-\mathbf{u}) - \bar{\alpha}(\mathbf{x})\beta(\mathbf{x} - \mathbf{u}+\mathbf{v})) \Big\rangle \right) \tag{7}$$

Next, the transparency map is modified, using the foreground appearance map modified above:

$$\begin{aligned} \alpha(\mathbf{x}) \leftarrow & \left(\Big\langle \Big(\phi(\mathbf{x}) - \beta(\mathbf{x}-\mathbf{u}+\mathbf{v}) \Big)^2 \Big\rangle \right)^{-1} \\ & \cdot \left(\Big\langle (\phi(\mathbf{x}) - \beta(\mathbf{x}-\mathbf{u}+\mathbf{v}))(I^t(\mathbf{x}-\mathbf{u}) - \beta(\mathbf{x}-\mathbf{u}+\mathbf{v})) \Big\rangle \right) \end{aligned} \tag{8}$$

In fact, before modifying the transparency map, an E-Step can be used to recompute the posterior, which changes once the foreground appearance is modified. Although this step is not required to guarantee convergence, we find it speeds up convergence. Next, the background appearance is modified:

$$\begin{aligned} \beta(\mathbf{y}) \leftarrow & \left(\Big\langle \bar{\alpha}(\mathbf{y}-\mathbf{v}+\mathbf{u})^2 \Big\rangle \right)^{-1} \\ & \cdot \left(\Big\langle \bar{\alpha}(\mathbf{y}-\mathbf{v}+\mathbf{u}) \Big(I^t(\mathbf{y}-\mathbf{v}) - \alpha(\mathbf{y}-\mathbf{v}+\mathbf{u})\phi(\mathbf{y} - \mathbf{v}+\mathbf{u}) \Big) \Big\rangle \right). \end{aligned} \tag{9}$$

Again, before modifying the background appearance, an E-Step can be applied to update the posterior.

6 Experiments

6.1 Sequence 1

We present results on a sequence of 5 frames, each of size 60×60 in which a person is moving in front of a cluttered background. Fig. 2d & e shows the sequence. We used 7x7 overlapping patches. For computational reasons, we restricted the search space for the foreground motion to be 7 pixels for horizontal shifts, and 2 pixels for vertical shifts. Similarly, for background motion, the search space is restricted to 1 pixel in either directions. Thus, the posterior over motion field involved a distribution over $75 \times 9 = 675$ values.

On convergence, we can reliably use MAP estimate of the posterior distribution for each pixel as the flow for that pixel since the posterior gets to be peaky. Fig. 2a-b shows the flow fields for both foreground and the background motion for each frame with respect to the reference frame. For ease of viewing, the flow fields for foreground motion is masked using its transparency mask.

Fig. 2c shows the learned appearance maps for the foreground and the background, and the transparency map. We find that the background accounts for pixels that were occluded in some frames. The transparency map is reasonable given that we used only 5 frames to train the model.

We can use the generative model to generate new data, and in particular do frame interpolation. We used the trained model to obtain frames between the reference frame and the first frame in the sequence (Fig. 3).

6.2 Sequence 2: Flower garden sequence

The flower garden sequence has a fast moving tree in front of slow moving complex background, recorded using a camera mounted on a moving vehicle. We used a small subset of the sequence consisting of 7 frames. The first and last frames of the sequence we used is in Fig. 4 b & c. We used 15x15 fully overlapping patches. The search space for foreground motion is restricted to lie between -10 and 10 for horizontal motion, and -2 to 2 for vertical motion. For the background motion, the search is between -2 and 2 for horizontal shifts and -1 and 1 for vertical shifts. Thus, the posterior distribution assigns probability for each of the 1575 flow combinations.

Fig. 4a shows the foreground (masked by mask) and background appearances and the transparency map. The portion of garden closer to observer is assigned to foreground, a possible reason being that the search space for background is very much restricted (within 2 pixel deviation), and the garden closer to the observer moves faster than that. Hence, it uses foreground to explain its motion better.

6.3 Sequence-3

We considered a sequence of 9 frames in which a person turns her hand within plane. Fig. 5a shows five frames from the sequence. We fixed the background to

be stationary, and allowed foreground displacement to be between -9 and 9 in both directions. The estimated flow fields for the frames are shown in Fig. 5b-c. The flow fields elucidate the non-uniform motion that the foreground object undergoes. In Fig. 6, we present the parameters learned using the estimation. The model has done a reasonable job of learning the transparency mask even though the data is limited and some of the pixels in the foreground object are stationary. Fig. 7 shows the interpolated sequence of 10 frames obtained by generating from the model.

7 Summary and Conclusions

We presented a layered generative model for inferring dense flow fields. Using unsupervised learning and variational method for inference, we solve the problem of estimating the appearances of foreground and background objects by segmenting local patches and inferring its local non uniform motion. This enables filling in of disocclusions. Being a generative model, learned model can be used to generate data and thus readily available for tasks such as frame interpolation.

References

1. Ayer, S. & Sawhney, H Layered representation of motion video using robust maximum-likelihood estimation of mixture models and MDL encoding In *Proc. IEEE International Conference on Computer Vision*, 1995
2. Barron, J.L., Fleet, D.J., & Beauchemin Performance of optical flow techniques In *International Journal of Computer Vision*, 12(1):4377, 1994.
3. Black, M.J. & Fleet, D.J. Probabilistic detection and tracking of motion discontinuities In *International Journal of Computer Vision*, 36(3):171-193, 2000
4. Darabiha A, Rose JR & MacLean WJ Video Rate Stereo Depth Measurement on Programmable Hardware. In *Proc. of IEEE Conference on CVPR* , 2003
5. Jepson, A. & Black, M.J. Mixture models for optical flow computation In *In Partitioning Data Sets: With Applications to Psychology, Vision and Target Tracking, Ingmer Cox, Pierre Hansen, and Bela Julesz (Eds.), AMS Pub.: Providence, pp. 271 286. RI, DIMACS Workshop*, 1993
6. Jojic, N. & Frey, B.J. Learning flexible sprites in video layers In *Proc. of IEEE Conference on CVPR* ,1999
7. Jojic, N. Frey, B.J. & Kannan,A. A Generative model of dense optical flow in Layers *University of Toronto Technical Report* PSI-2001-11, 2001
8. Lucas, B.D. & Kanade, T., An Iterative Image Registration Technique with an Application to Stereo Vision In *Proceedings of the 7th International Joint Conference on Artificial Intelligence*, 1981
9. Zhou,Y. & Tao,H. A Background Layer Model for Object Tracking through Occlusion In *Proc. IEEE International Conf. on Computer Vision, ICCV*, 2003
10. Wang, J.Y.A. & Adelson, E.H Layered Representation for Motion Analysis In *IEEE Conference on Computer Vision and Pattern Recognition*,1993
11. Weiss, Y. & Adelson, E.H A unified mixture framework for motion segmentation: Incorporating spatial coherence and estimating the number of models In *Proc. IEEE Computer Vision and Pattern Recognition*, 1996.

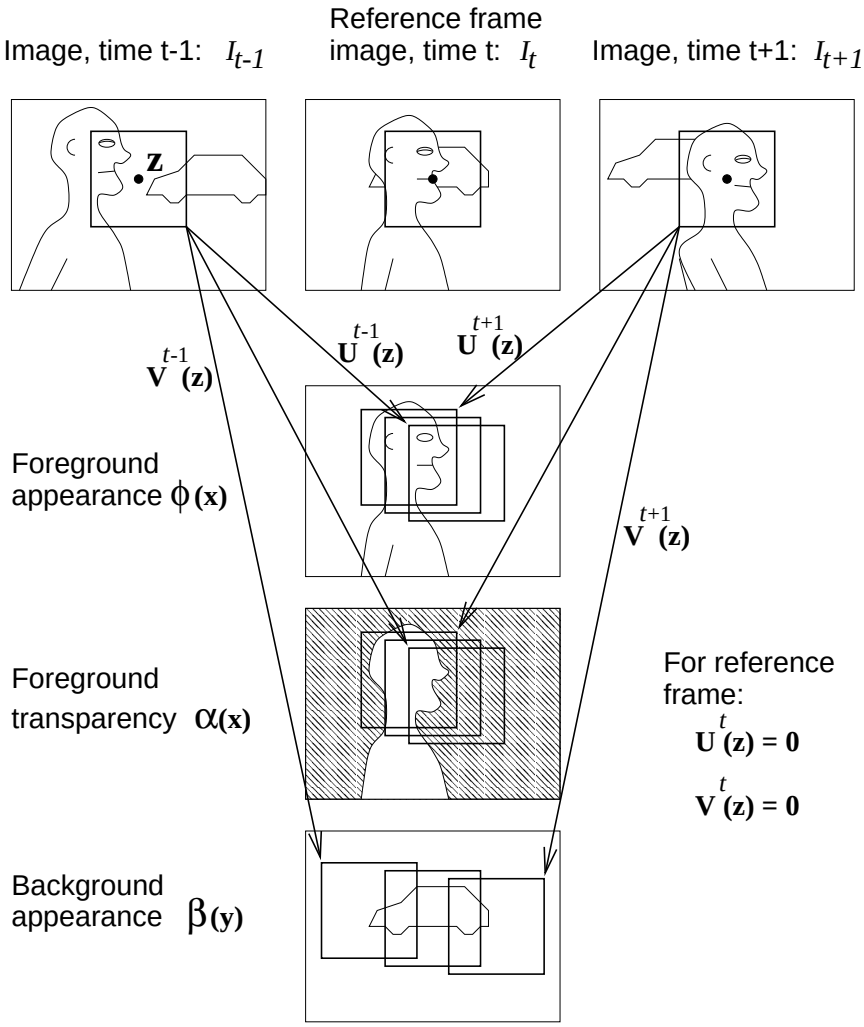

Fig. 1. Nearby video frames are modelled with respect to a reference frame. The appearance of every $K \times K$ patch in nearby frames is modelled using a patch from a foreground transparency map to combine a patch from a foreground appearance map with a patch from a background appearance map. Here, we show how a patch centered at $\mathbf{z}$ in 3 frames is explained by composing displaced patches from the maps. At time $t-1$, the foreground patch and transparency patch displaced by the motion vector $\mathbf{U}^{t-1}(\mathbf{z})$, while the background patch is displaced by the motion vector $\mathbf{V}^{t-1}(\mathbf{z})$. The maps are estimated using the EM algorithm.

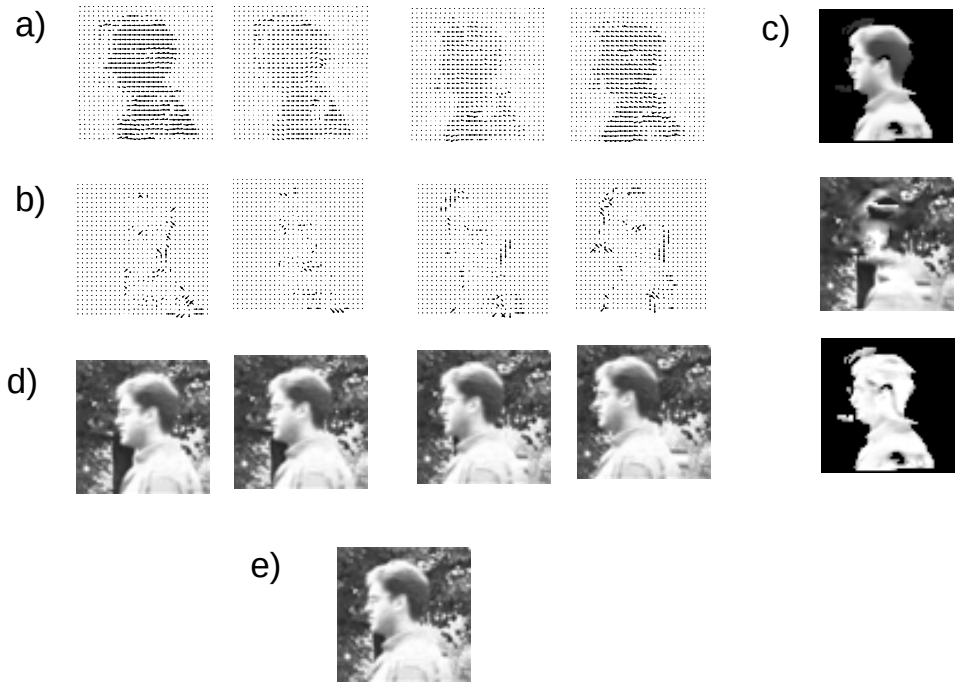

Fig. 2. a) Inferred flow fields for the foreground motion b) Inferred flow fields for the background motion c)the learned parameters of the model - appearance maps and the transparency map. These are inferred using the sequence of 5 frames in d) with e)as the reference frame.

Fig. 3. Five frames obtained from generating from the model.

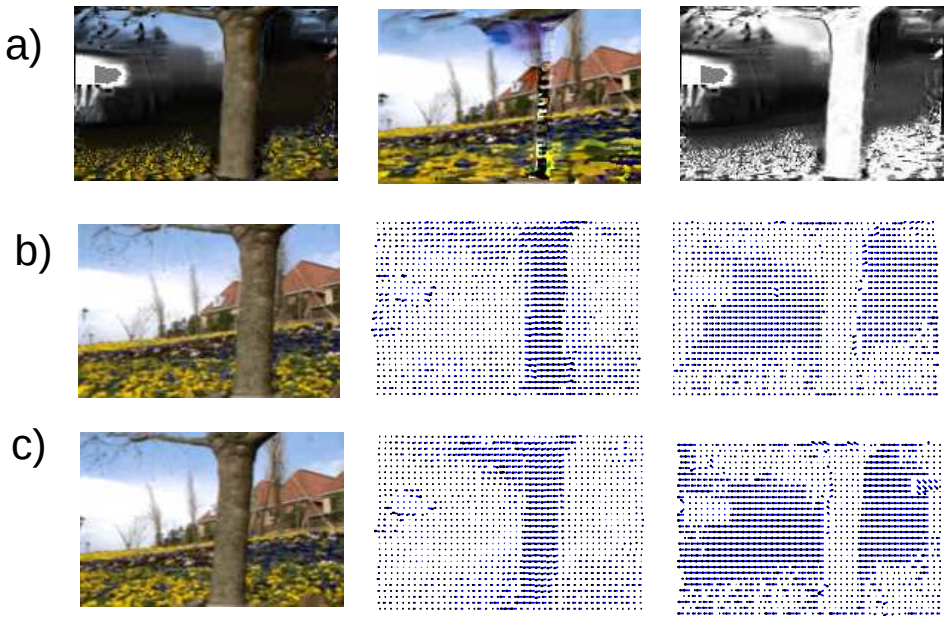

Fig. 4. a) appearance and transparency maps. b) the first frame and its foreground and background motion fields c) last frame and its motion fields.

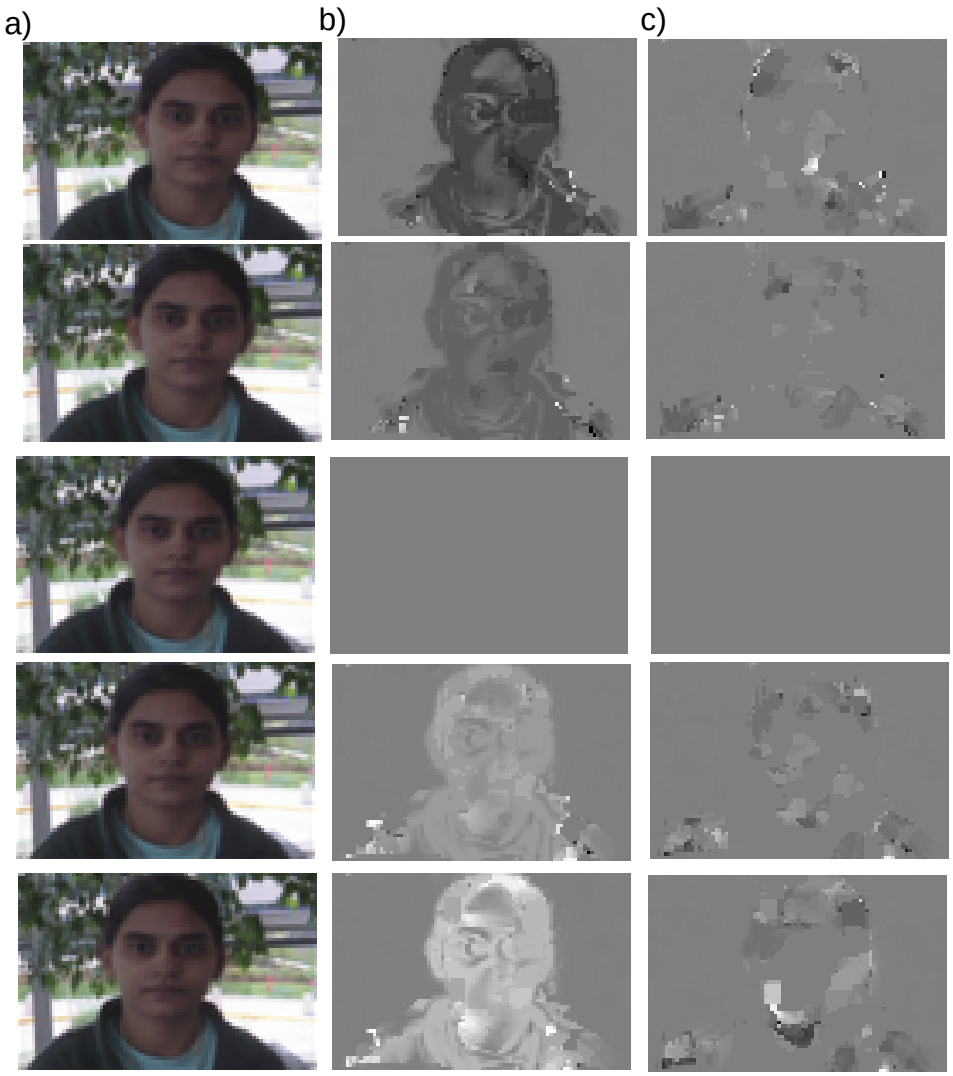

Fig. 5. a) Five (alternate) frames from the sequence, with the reference frame in the middle b) horizontal component and c) vertical component of foreground motion. Pixels for motion are scaled such that white is 9 pixels shift to right (horizontal motion) or up(vertical motion), and black is 9 pixels shift in the opposite direction, with gray representing no motion. We have masked the flow fields with appropriate transparency map for the frame for clarity.

Fig. 6. The parameters learned using a sequence of 9 frames of a person turning her head, as shown in Fig. 7

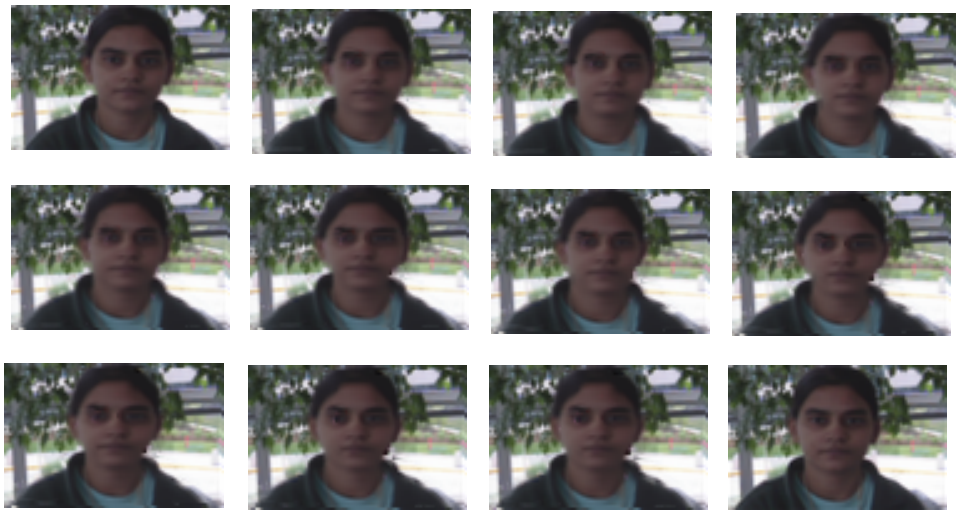

Fig. 7. Ten frames generated from the model between the reference frame(shown here as the first frame) and the last frame of the original sequence (shown here as the last frame).