Coherent Motion Segmentation in Moving Camera Videos using Optical Flow Orientations - Supplementary material

Manjunath Narayana
narayana@cs.umass.edu

Allen Hanson
hanson@cs.umass.edu

Erik Learned-Miller
elm@cs.umass.edu

University of Massachusetts, Amherst

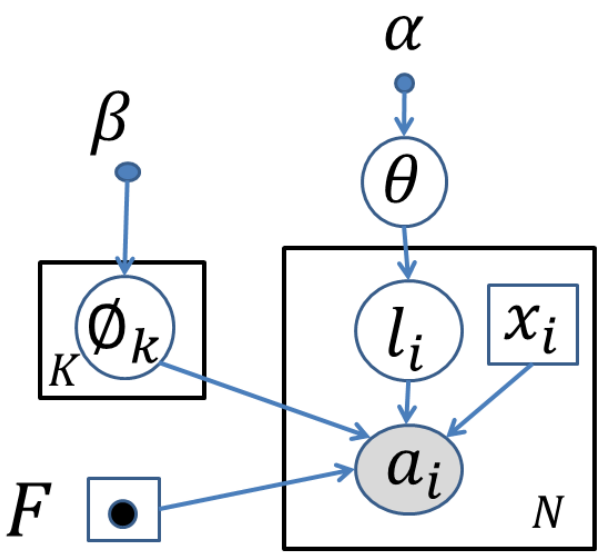

Figure 1. A mixture model for segmentation based on optical flow orientations.

1. Introduction

In this supplementary material, we present some additional details and results for our paper. The sampling procedure used in our graphical model is in Section 2. The entire set of 46 orientation fields used as the "library" FOFs are presented in Section 3. Comparison of our segmentation results to various other methods is presented in Section 4. In Section 5, we compare our model to other models with optical flow orientations as the input. In addition to this document, the supplementary material includes sample video results for all 37 videos that we tested. Video results are discussed in Section 6.

2. The model

As explained in the paper, Figure 1 represents our probabilistic segmentation model. Our sampling scheme is given in Algorithm 1. The algorithm is similar to the Gibbs sampling for Dirichlet processes with non-conjugate priors as described by Neal (algorithm 8 in [3]). The algorithm adds additional auxiliary Φ parameters at each iteration. We begin with $K = 1$ component and add one new auxiliary component$(M = 1)$ to the model at each iteration.

Algorithm 1 Sampling procedure in our model

Step 0 : Initialize $\Phi_1 = c_1$ where c_1 is sampled from a uniform distribution over the set of "library" camera motion parameters.
for n iterations **do**
  Let K be the current number of motion components.
  Sample M new motion parameters from β.
  for $i = 1$ to N pixels **do**
    Sample label l_i with the following probability:

$$P(l_i = c|c_{-i}, a_i, \Phi_1, \Phi_2, ...\Phi_{K+M}) = \begin{cases} b\frac{N_{-i,c}}{N-1+\alpha}P(a_i, \Phi_c) & \text{for } 1 \leq c \leq K \\ b\frac{\frac{\alpha}{M}}{N-1+\alpha}P(a_i, \Phi_c) & \text{for } K < c \leq K+M \end{cases}, \quad (1)$$

    where c_{-i} represents c_j for all $j \neq i$, $N_{-i,c}$ represents the number of labels l_j that have value c and $j \neq i$, and b is an appropriate normalization constant.
  end for
  for all $c \in \{c_1, c_2, ...c_N\}$ **do**
    Draw a new value $\Phi_c|l_i$ such that $l_i = c$.
  end for
end for

3. Flow orientation fields

As explained in the paper, we use a discrete number of orientation fields or FOFs' which try to explain the observed orientations in the current frame. In the paper, we presented a subset of FOF's used. The complete set of orientations

is given here in Figure 2. The motion parameter tuple $t = (t_x, t_y, t_z)$ responsible for each FOF is listed within each image.

4. FOF versus flow vector-based segmentations

In the paper, we compared our segmentation results to spectral clustering results of Ochs and Brox [4], which represented the best method among many that we experimented with. K-means, Dirichlet process Gaussian mixture model, and the spectral clustering method of Elqursh and Elgammal [1], were among the other methods we used. Sample results from all methods are given in Figure 4. The first four rows are the input image, a visualization of the optical flow vectors [5], the optical flow magnitudes, and optical flow orientations respectively. The optical flow vectors appear to have similar values for all background pixels when the background is relatively simple and at roughly uniform depth from the camera. When background objects are at varying depths (columns 3 and 4), their flow vectors are not uniform. The magnitudes of the vectors (row 3) depend heavily on object depth. Optical flow orientations (row 4), arguably, are the most reliable indicators of independent motion. For orientations, it may be noted that the color blue represents 0 degrees and red represents 360 degrees. Hence they should be considered as equivalent (for instance, in column 3).

As a baseline method, we first used K-means with the flow vectors as input. Rows 5, 6, and 7 show the results of K-means clustering for $K = 2$, $K = 3$, and $K = 5$, respectively. The results are highly sensitive to the value of K. For videos with simple backgrounds (columns 1, 2, and 5), small values of K work well. For complex background videos (columns 3 and 4), there is no value of K that yields good results. This can be seen in row 8 where human judgement was used to pick the best results from many different K values.

Since requiring knowledge of K beforehand severely limits the use of K-means to general video settings, we next show a non-parametric mixture model with optical flow vectors as the features. We use the accelerated variational Dirichlet process Gaussian mixture model (DPGMM) implementation of Kurihara *et al*. [2]. Results from DPGMM are shown in row 9. Although DPGMM is non-parametric and can adapt to the complexity in the data, the use of optical flow vectors as the features causes the method to oversegment the background.

Spectral clustering has been shown to be useful for motion segmentation by clustering tracked keypoints based on their long-term trajectories. Elqursh and Elgammal [1] find a low-dimensional embedding for trajectories from 5 consecutive frames. However, their method tends to separate the background into several clusters. In order to apply their method for background subtraction, they assume that the background is a Gaussian mixture model of 5 components. Row 10 shows keypoints with the colors representing cluster memberships for their algorithm with a mixture model of 5 components[1]. It is not clear whether using a mixture of 5 components would work for across all videos. In some videos, the foreground object keypoints clearly form a separate cluster (column 1 and 5), but in others, they are grouped with the background.

Finally, we compare to the spectral clustering of Ochs and Brox [4] which represents the state of the art for segmentation of trajectories. This method considers interaction between triplets of keypoints instead of simply using pairwise distances for the clustering. Further, they use some post-processing to merge regions with similar motion properties. The results from their implementation[2] that includes a merging step is given in row 11. Again, the results show keypoints with the colors representing their clusters. The method works well when the background is fairly at a uniform distance from the camera. Complex backgrounds, however, still suffer from over-segmentation. Both the above spectral methods result in labels for sparse keypoints. For a dense labeling of all pixels, additional processing is required. In contrast, our method directly returns a dense labeling of the image.

Our results in the last row show the efficacy of our method across different scenarios. Note that these results are from the orientation-based segmentation alone, without any use of color or prior information. Foreground objects are broken down into smaller segments depending on their motion, but all of the background is correctly identified as one segment. Our method is prone to failure when the object's motion happens to be in a direction that is consistent with the orientation field due to the camera's motion (column 5). The object will go undetected until it or the camera change their motion direction. In the above video, the object is detected after 5 frames.

5. Our model versus other models using flow orientations

Section 4 shows the advantages of using flow orientations compared to flow vectors. In this section, we compare other models to ours while using the same feature representation. Using flow orientations as the common feature representation, our segmentation model is directly compared to K-means and DPGMM in Figure 5. Once again, results from K-means (rows 3, 4, and 5) are sensitive to the choice of K. When the correct value of K is provided by human judgement, the resulting segmentations are reasonable (row 6), but still fall short of our FOF results in the last

[1] Our own implementation of [1]

[2] We set the number of tracked frames to 3 in order to keep the comparison to our method fair

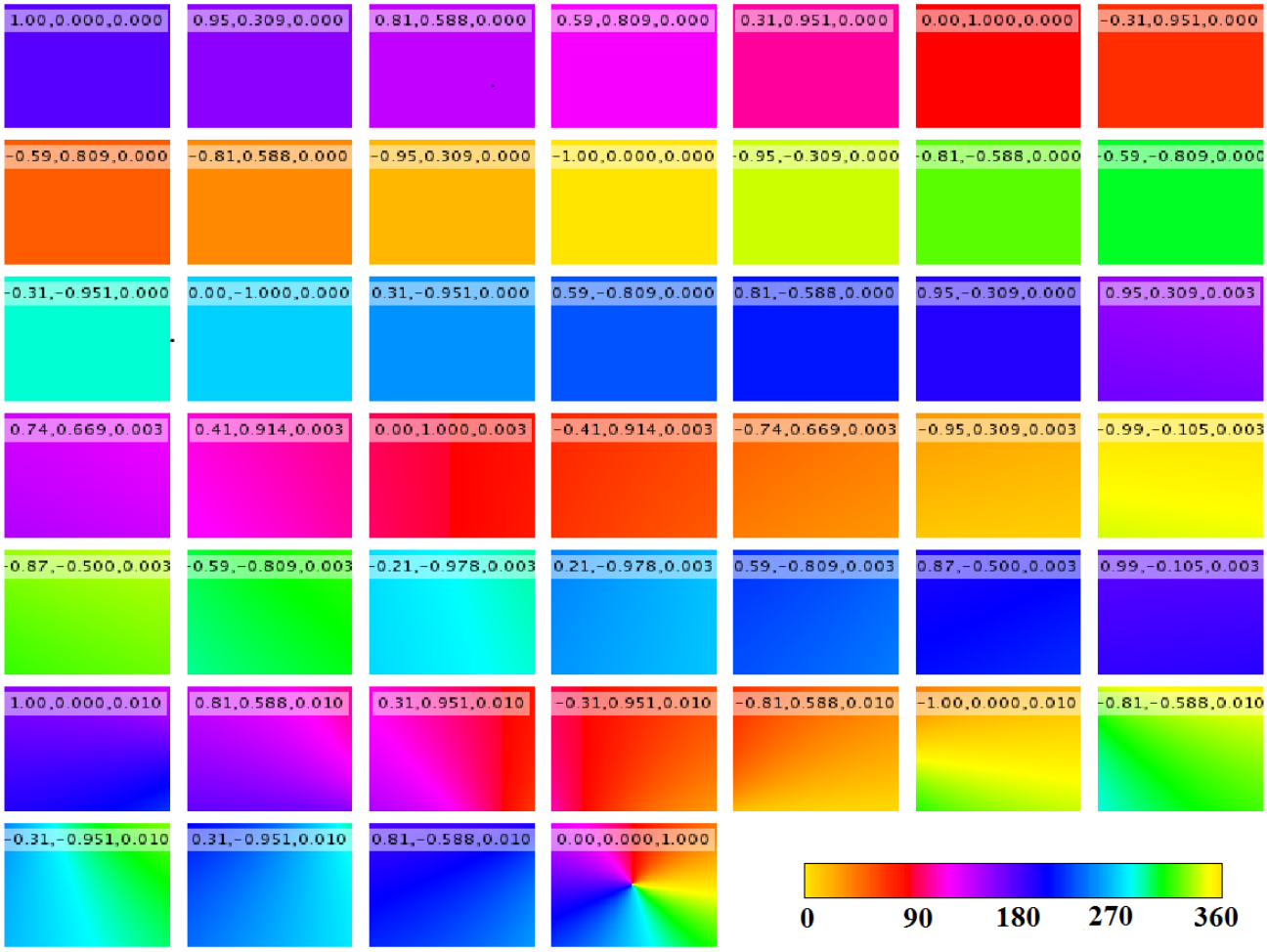

Figure 2. The complete set of orientation fields used in our model. The motion parameters responsible for each field are given within each image. The color bar on the bottom right of the figure shows the mapping from angles in degrees to color values in the images.

row. For complex backgrounds (columns 3 and 4), K-means with flow orientations results in a better segmentation than K-means with flow vectors in Figure 4. DPGMM automatically determines the number of components, but tends to break the background into smaller sections. Our segmentation method, by explicitly modeling the spatial dependency between pixels through the "library" FOFs', results in a better segmentation compared to both human-assisted K-means and DPGMM.

6. Video results

Sample video results from all our test videos are available in the supplementary submission. We show the first 20 frames in each video[3]. The same set of parameters (as specified in the paper) have been used for all 37 videos. The videos are compressed AVI files and have been tested on Windows 7 Media player and IrfanView version 4.35. The videos are organized into three folders - Hopkins, SegTrack, and complexBg. Each video shows the input image, the optical flow orientations, the FOF-based segmentation, and FOF-based segmentation with color and prior information. The results show our system capable of handling situations when there is no independently moving object (Hopkins-marple6 and 11), detecting objects that are initially at rest but begin moving later (Hopkins-marple9), and recovering objects that are initially not detectable using orientations (Hopkins-cars1). Videos with no camera motion are handled well by using a "zero-motion" FOF (SegTrack-birdfall). This video shows the noisy optical flow orientations which are actually ignored when computing the segmentation. Most videos show that using only orientations can cause occasional errors, but the use of color and prior information helps recover from them.

[3]Some videos in the data sets contain fewer than 20 frames. All frames are reported for these videos

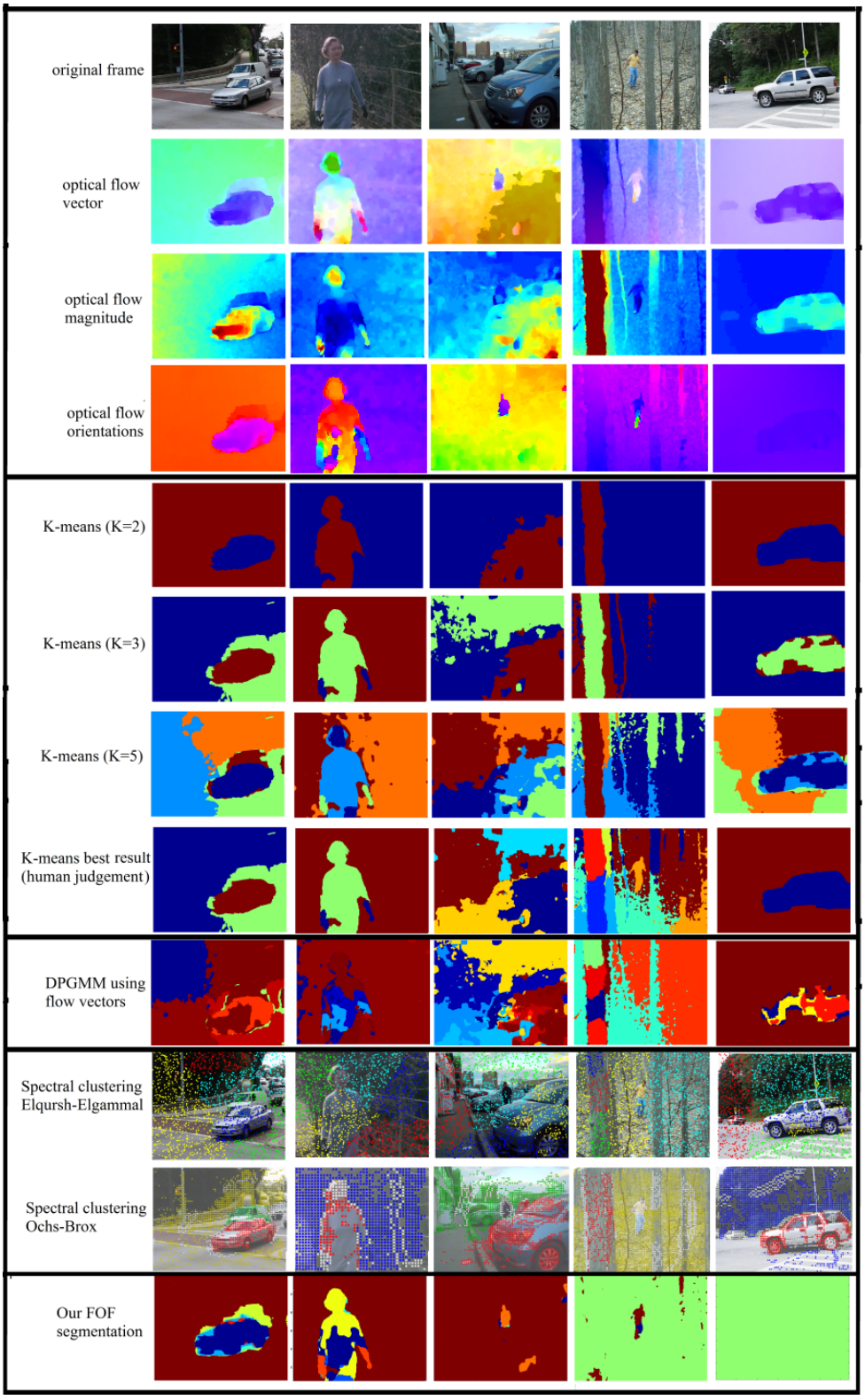

Figure 3. Comparison of FOF segmentation to several other optical flow vector-based methods.

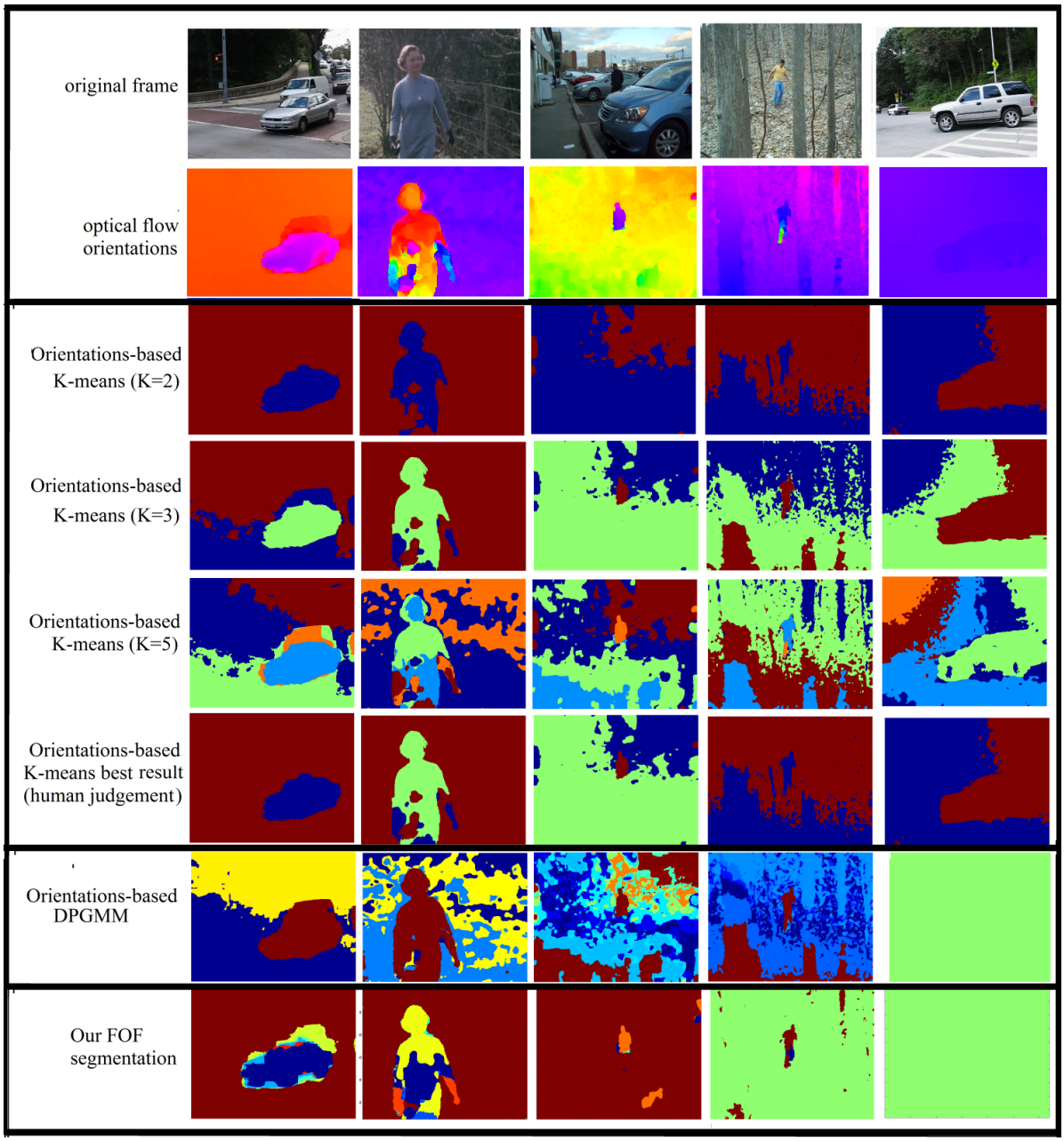

Figure 4. Comparison of our FOF model to other methods using orientation fields.

Some difficult cases for our algorithm are when the object moves very slowly (Hopkins-cars9) and when optical flow is not reliable (SegTrack-cheetah). In presence of camera rotation (complexBg videos, Hopkins-cars7), the orientation-based segmentation is more prone to making errors. In these videos, color and prior information are very useful in correcting such errors.

References

[1] A. Elqursh and A. M. Elgammal. Online moving camera background subtraction. In *ECCV*, 2012. 2

[2] K. Kurihara, M. Welling, and N. Vlassis. Accelerated variational dirichlet process mixtures. In *NIPS*, 2006. 2

[3] R. M. Neal. Markov chain sampling methods for dirichlet process mixture models. *Journal of Computational and Graphical Statistics*, 9(2):249–265, 2000. 1

[4] P. Ochs and T. Brox. Higher order motion models and spectral clustering. In *CVPR*, 2012. 2

[5] D. Sun, S. Roth, and M. J. Black. Secrets of optical flow estimation and their principles. In *CVPR*, 2010. 2