Towards Unsupervised Semantic Segmentation of Street Scenes From Motion Cues

Hajar Sadeghi Sokeh
Research School of Computer Science
The Australian National University
Canberra, ACT 0200
hajar.sadeghi@anu.edu.au

Stephen Gould
Research School of Computer Science
The Australian National University
Canberra, ACT 0200
stephen.gould@anu.edu.au

ABSTRACT

Motion provides a rich source of information about the world. It can be used as an important cue to analyse the behaviour of objects in a scene and consequently identify interesting locations within it. In this paper, given an unannotated video sequence of a dynamic scene from fixed viewpoint, we first present a set of useful motion features that can be efficiently extracted at each pixel by optical flow. Using these features, we then develop an algorithm that can extract motion topic models and identify semantically significant regions and landmarks in a complex scene from a short video sequence. For example, by watching a street scene our algorithm can extract meaningful regions such as roads and important landmarks such as parking spots. Our method is robust to complicating factors such as shadows and occlusions.

Categories and Subject Descriptors

I.2.10 [**ARTIFICIAL INTELLIGENCE**]: Vision and Scene Understanding; I.4.8 [**IMAGE PROCESSING AND COMPUTER VISION**]: Scene Analysis

General Terms

Algorithms, Experimentation

Keywords

Semantic segmentation, motion cues, scene understanding

1. INTRODUCTION

Semantic scene segmentation is a fundamental task with a large number of applications in video processing and artificial intelligence. This task identifies meaningful regions behind the content of a scene and is often referred to scene labelling. In an unsupervised algorithm, the first task is to identify robust features to distinguish different areas of interest in the scene. There are many different features for this goal such as texture, colour, motion and so on. In this paper, we focus on motion and colour. Moreover, we treat each pixel independently, that is, without any detecting or tracking of moving blobs across frames.

One of the issues confronting the use of motion features is perspective distortion. When an object is moving in a scene, the perception of its size and speed in the 2D image changes relative to its distance from a stationary camera. To overcome this problem and achieve camera-free features, we employ a simple self-calibration method which does not require Euclidean information. Specifically, we consider a single object of constant height over at least two frames of the video and use the perceived size change to calibrate for distance.

Generally, the problem that we tackle in this paper is automatic segmentation of urban scenes into major semantic categories of interest. For example, our method discovers categories such as side-walk, road, cross road, parking place and some traffic signs that are not easy to identify in crowded areas or in low resolution videos.

Our approach is composed of the four modules. The first module performs motion detection using optical flow. It also removes all shadows from moving objects. The second module extracts robust features for segmentation. The third module removes perspective distortion from the features and finally the fourth module clusters the scene based on the features. We evaluate our method on two different metrics and show increased precision of our results in comparison to a baseline method.

2. RELATED WORK

Dynamic scene understanding has been an active area of research for many years. Traditionally, many of the existing methods exploit only colour and texture information from a single view of the scene. Cao et al. [3] used an appropriate distance measure in the composite feature space of colour and texture. Seetha et al. [13] proposed an unsupervised colour texture segmentation using Expectation Maximisation. However in many applications, texture and colour are not enough. For example, roads and side walks have locally similar appearance but very different semantics. Working with video sequences gives us another feature to exploit: Motion, which can help us draw some information from the behaviour of moving objects in the scene and distinguish between regions with locally similar appearance. Some works are based on object tracking [18, 12] which need more computation for tracking and some others apply probabilistic models [9] for clustering the scene.

IVCNZ '12, November 26 - 28 2012, Dunedin, New Zealand

Wang et al. [17] suggested a new generative model of topic modelling called Spatial Latent Dirichlet Allocation (LDA) to encode spatial structure among visual words and group them to different segments. They defined the whole image as a document and each local patch as a word and categorised the close visual words in feature space into the same document. In this work we also use Latent Dirichlet Allocation for clustering, but instead of spatial relations among visual words, we consider motion features in temporal structure of the video to find the topic associated to each pixel as a document.

One of the issues we face in comparing some features is perspective distortion. Using camera calibration methods [14] helps us remove this distortion, for example to normalise the height of objects. A common drawback of these methods is that they require known camera parameters or have at least two images of each frame. Albeit there is some research which just work on one image as described in [10] to find three orthogonal vanishing points. However, these are not always available. Criminisi [5] calibrated the camera by finding vanishing line and vertical vanishing point which finding vertical vanishing point is not feasible in images without some converging vertical lines. Video usually gives us some other useful information about the scene. We extract this information by tracking one person at least in two different frames. Then by using camera calibration formulas, we can remove perspective distortion in the extracted features.

3. MOTION DETECTION

In this module, we independently detect all moving objects by measuring the velocity field of pixels in the frames. This is accomplished using an optical flow algorithm. We also apply frame differencing and shadow removal to obtain clean boundaries of moving objects. All of these steps are explained below.

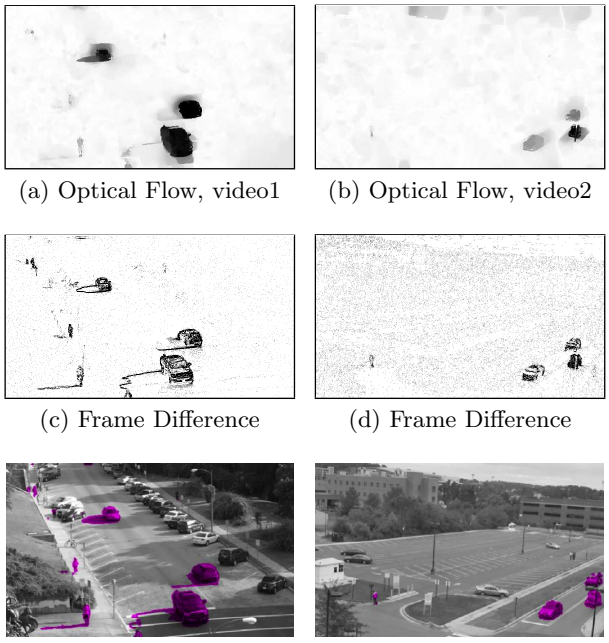
3.1 Optical Flow

Almost 30 years ago, Horn and Schunck [8] published their seminal paper on optical flow calculation and its techniques. In our work optical flow is used to estimate magnitude and direction of movement in each pixel, which defines two features for segmentation. We use the optical flow method proposed by Sun et al. [16], which de-noises the flow using median filtering to improve accuracy.

3.2 Frame Difference

Frame differencing is a simple way to detect changes between subsequent frames. The output of this method is inherently noisy due to similarity in intensity inside moving blobs in two consecutive frames. In contrast to optical flow, however, frame differencing results in clean object boundaries. In this work, we combine both methods to obtain clean boundary and exact velocity information for each moving object.

Initially we apply both algorithms on successive frames. As shown in Fig. 1, inside each moving blob, the result of frame difference is sparse. We complete the missing pixels within detected blobs by interpolating from optical flow information. Specifically we connect adjacent pixels if they have the same optical flow. At the end, all detected blobs are filled with the velocity calculated by the optical flow algorithm.



(a) Optical Flow, video1 (b) Optical Flow, video2

(c) Frame Difference (d) Frame Difference

(e) Combined method (f) Combined method

Figure 1: **Best viewed in colour.** Difference between optical flow, frame difference and combined method for both videos. (a) and (b) are the output of optical flow, (c) and (d) are the frame difference of two subsequence frames and finally(e) and (f) show the detected blobs by combining both methods.

3.3 Shadow Removal

Dynamic shadows generated as a result of bright point-like light sources, e.g., the sun in outdoor scenes, are a major problem confronting motion detection algorithms. Zhang et al. [19] introduced a new feature called Ratio Edge to find shadows. Ratio Edge represent the quantity of the texture in one neighbouring region. We simplify their method by having the background.

One simple way to find the background is averaging frames in the video. However, in crowded areas like our scenes, the time available for viewing background intensity values is small, so the result is not very satisfactory as is shown in Fig. 2. A better way is to average just the no-motion pixels per frame.

To calculate the Ratio Edge, we define the neighbouring region of pixel (x, y) as:

$$\theta(x,y) = \Big\{(x+i, y+j) \mid 0 < i^2 + j^2 \leq r^2 \text{ and } f(x+i, y+j) \neq 0\Big\} \quad (1)$$

in which r is the radius of the neighbouring region and $f(x, y)$ is the intensity value at pixel (x, y). The ratio edge of this pixel is then defined as:

$$R(x,y) = \sum_{(i,j)\in\theta(x,y)} \frac{f(x,y)}{f(i,j)} \quad (2)$$

Under certain conditions [19], there is a noticeable differ-

(a) Averaging method

(b) Our method

(c) Difference between two methods

Figure 2: Comparison between averaging method and our method to find a soft background. (a) shows the output of averaging over frames. (b) is the background found by our method. (c) is the difference between two methods. (image is inverted to be better viewed)

(a) Original frame

(b) Un-shadow frame

Figure 3: **Best viewed in colour.** Shadow removal results. (a) before and (b) after shadow removal.

ence between the ratio edge of the pixels under shadow and without it. By automatically specifying a threshold T_{ratioE}, we can differentiate pixels in shadows from others. The shadows are then detected by a simple rule:

$$\text{if } (R(x,y)_{background} - R(x,y)_{currentFrame}) < T_{ratioE}$$

$$\text{then } (x,y) \in Shadow \tag{3}$$

The results of this shadow removal algorithm for an example scene are shown in Fig. 3.

4. FEATURE EXTRACTION

Using irrelevant attributes usually adds noise to our data and also increases the memory usage, computation time and overall system resources. In our method, the attributes used to build the model are based on the output of motion detection and are listed below:

- The size of objects moving over each pixel
- The magnitude of optical flows on each pixel
- The direction of optical flows on each pixel

In this work we use three major motion features each of which is strong enough to distinguish interested regions. For example, vehicles moving on the road are usually faster than

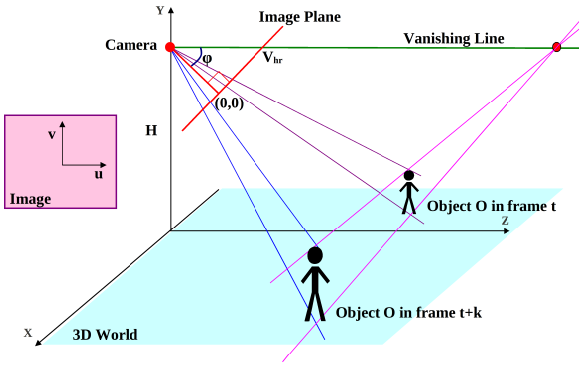


Figure 4: The pinhole camera and vanishing line in two different frames

people. So in these pixels we have flows with a relatively high magnitude. Furthermore the size of people commonly is smaller than that of vehicles. Hence, we can use these features to differentiate between side-walk and road. By considering both of them, we could determine the cross road area where both people and vehicles can move on these pixels but usually with different directions. Some semantic regions like approaches to stop signs and traffic lights also can be recognised by the changing speed of objects on the road.

We also use colour information as a low-level feature which can be helpful in homogeneous colour regions where perhaps motion is absent. We tested two different colour spaces, RGB and Lab, and compared the results (see Sec. 6).

The features used in this work are highly affected by perspective, so we employ simple calibration to overcome this problem. There is a direct relation between the geometry of the scene and camera calibration. In this paper we use motion: by tracking objects in two different frames we can estimate the relative place of moving objects in 3D world.

From the pinhole camera model [7], some mathematical relations can be defined in homogeneous coordinates (see Fig. 4). Let (u, v) be the image coordinates of the world coordinates (X, Y, Z), then by projection, we have:

$$\begin{bmatrix} \lambda u \\ \lambda v \\ \lambda \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f.\cos\phi & f.\sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} . \begin{bmatrix} X \\ Y - H \\ Z \end{bmatrix} \tag{4}$$

where f is the focal length of the camera, H is the height of the camera place from the ground and ϕ is the angle between focal length and horizon. We assume that the principal point of the camera is located at the image centre.

Considering an object resting on the ground plane we can use Eq. 4 to arrive at:

$$(f + v_T.\tan\phi).(Y - H).(v_B - f.\tan\phi) =$$

$$-(f + v_B.\tan\phi).H.(v_T - f.\tan\phi) \tag{5}$$

where v_T and v_B are the head and feet points for an object in the image plane.

However even by tracking many people in different frames, we can not solve Eq. 5. Regarding the fact that the real height of an object appearing in two frames remains the same, we can find the vanishing line (see Fig. 4). Then we

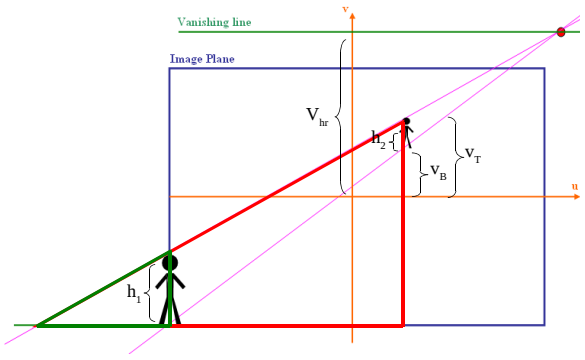


Figure 5: Removing perspective distortion using considering a reference point as the left-bottom point in the image plane

have:

$$\tan\phi = \frac{V_{hr}}{f} \tag{6}$$

in which V_{hr} is the vertical distance of vanishing line to the principal point. The vanishing line is located by using top and bottom points of one object in two different frames as shown in Fig. 4.

To remove perspective distortion, we map each moving blob with v_T and v_B to a reference point which is chosen to be the left-bottom point in the frame. The height of the image plane is indicated by r. These parameters and their relations are shown in Fig. 5 and after some algebraic manipulation and using Eqs. 5 and 6, we obtain:

$$\frac{h_1}{h_2} = \frac{(V_{hr}+\frac{r}{2})(v_T+\frac{r}{2})}{(V_{hr}+\frac{r}{2})(v_T-v_B)+(V_{hr}-v_T)(v_B+\frac{r}{2})} \tag{7}$$

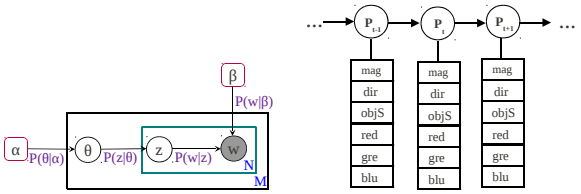
By finding the ratio between the heights, $\frac{h_1}{h_2}$, we can map the object size in each pixel to a reference point and remove the perspective problem.

5. SEGMENTATION

Latent Dirichlet Allocation (LDA) [2] is a probabilistic model for automatically clustering collections of discrete data. This model is a powerful machine learning algorithm which was first presented in the context of document analysis to group words into topics and associate a probability distribution with each document over topics. In recent years, some other applications have been found for LDA in the computer vision field, for example, action recognition [4], classification [6] and image segmentation [15].

In our work, we treat each pixel in the image plane as a document and the whole image as the corpus. Each pixel represents a ray from the constant camera and can therefore provide a model for the ground plane point intersecting that ray. All extracted features per pixel in each frame act as a word. The visual words in our case form descriptors for motion patterns of moving blobs and static colour information. We use three motion features and three colour features, so we will have six-dimensional words for each pixel. We quantise their six-dimensional values into discrete words with the k-means algorithm.

Suppose we have t video frames with M pixels per frame. Each pixel contains words from a vocabulary of size N which should describe the properties of each pixel and what happens in it during the video. We find the hidden topic for



(a) LDA model (b) Words in our model

Figure 6: (a) shows the graphical model representation of LDA. The outer plate represents M documents, while inner plate represents the repeated choice of topics and words within a document. (b) illustrates graphical representation of each word in our model

each pixel and group similar ones with the same topics. Concretely, each pixel d_i is defined as a set of N_i visual words $w_{ij}, j = 1, 2, ..., n$ and $n \leq t$. In fact, each word w_{ij} is a six-dimensional measured value for pixel i in each of j frames, in which each of these dimensions are one of the measured features for that pixel. In addition, there is a latent topic variable z_i associated with each pixel which represents the labels for pixel, $z_i = 1, \ldots, K$. All the words in the corpus will be clustered into K topics which are the total number of latent topics and the number of semantic segments in our video. The graphical model of LDA is shown in Fig. 6. The joint distribution over this model is given by:

$$p(\theta, z, w|\alpha, \beta) = p(\theta|\alpha)\prod_{i=1}^{N} p(z_i|\theta)p(w_i|z_i, \beta) \tag{8}$$

where α and β are the parameters of the Dirichlet prior on the per-pixel label distribution and label visual words distribution, respectively. We are interested in θ which contains the probabilities of each pixel belonging to each of K labels. The selected label for that pixel will be the label with the highest probability.

6. EXPERIMENTAL RESULTS

By analysing the features at each pixel, we segment the scene into different semantic landmarks like cross-roads and stop places. Experiments were carried out using the challenging VIRAT video dataset [1]. We chose two videos of crowded streets and parking lots with about 1300 frames for each video. To make the run time more efficient, we sample every 10th frame. The applied threshold for Ratio Edge, T_{ratioE}, was automatically calculated by k-means.

For each video, we extract different regions by applying an implementation of LDA using Gibbs Sampling [11]. Heuristically in LDA, by having K different topics, good initial values for α and β are $50/K$ and 0.01, respectively. To objectively evaluate the quality of topics discovered by LDA, we measure the performance of segmentation in each region. For evaluation, we suppose that for each cluster, the label is determined based on which label is in the majority in the region defined by the ground truth. Then we calculated F1-measure for each clustering result with different values for K. The F1-measure is a combination of purity and recall which are defined as the fraction of majority of labels in one cluster to the whole number of labels in the same cluster and to the whole number of the same label in the clustering result, respectively. Table 1 shows the values of F1-measure for

Table 1: The values of F1-measure for different values of K

K	8	9	10	11
$F1$-measure	63.5	64.7	64.5	63.8

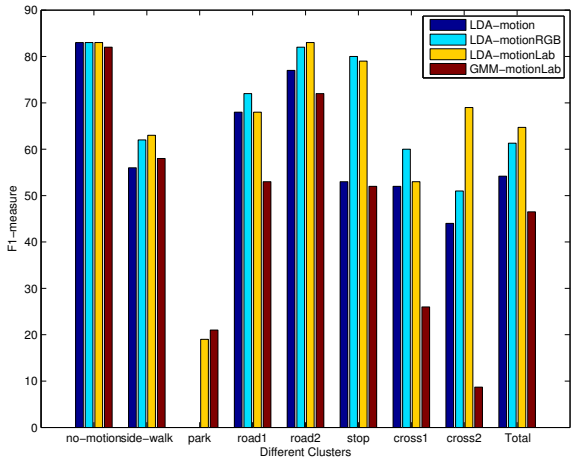


Figure 7: Quantitative comparison between LDA with different groups of features and GMM as a standard method

different values of K. It is notable that there is a maximum for the best value of K which is 9 in the first video.

As explained above, we tried three different groups of features: motion, motion+RGB and motion+Lab. We also compared our method with Gaussian Mixture Model (GMM) with motion and Lab colour features, as a standard method. For evaluating the results, we manually annotated all pixels and made a ground truth for each video which is shown in Fig. 8. A quantitative comparison, with respect to the F1-measure, of all algorithms against the ground truth annotations is shown in Fig. 7. Here, it is clear that LDA clusters better than GMM in most regions even when we use just motion features. Another notable point in this diagram is that the segmentations produced by motion+Lab features agree fairly well with the ground truth segmentations for this set of frames.

Fig. 8 shows the first frame of the video, the annotated labels as ground truth and the segmentation results generated by our algorithm for both videos. The quantitative evaluation based on purity and recall also is displayed in Table 2 and 3 for both videos. These results indicate that some semantic regions like cross roads or stop points on the road are possible to be captured based on motion features. Nevertheless, it is clear that parking places can not be distinguished until at least one vehicle parks in that place. So, the parking points, for example in the second video, are not extracted clearly, because the incidence of parking vehicles in this video is not frequent enough. This also happens for some quiet roads like *road3* and *road8* in the second video. Moreover, in this video some people and vehicles move across vacant spaces in the parking lot as is shown in Fig. 8, making these areas much noisier. Another noticeable point in the stop places on the road is that in both videos this segment is clustered to two different regions. This is due to the stop action for the first moving vehicle on the road causing following cars to slow and stop. If there are many vehicles on the road, after stopping the first one, all of vehicles should

Table 2: Quantitative evaluation on the first video

Clusters	LDA		GMM	
	Purity	Recall	Purity	Recall
no-motion	75	93	74	93
side-walk	53	77	61	56
park place	33	14	33	15
road1	95	53	41	76
road2	95	74	75	69
stop place	84	75	65	44
cross-road1	63	46	26	26
cross-road2	64	75	99	4.5
Total	70	64	62	51.4

Table 3: Quantitative evaluation on the second video

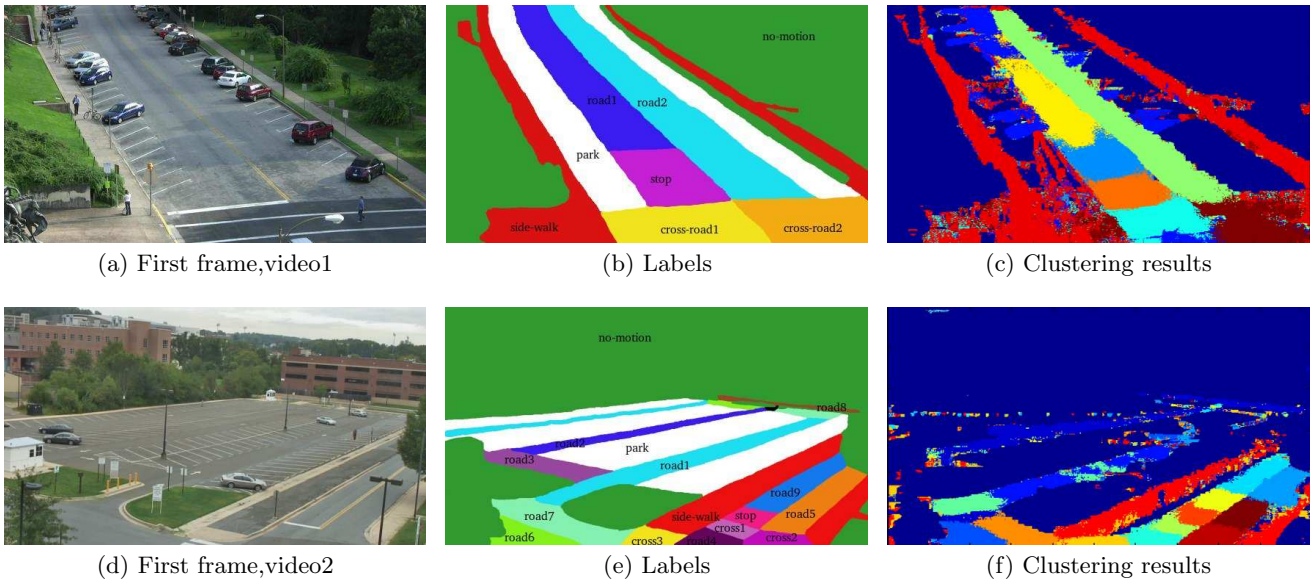
Clusters	LDA		GMM	
	Purity	Recall	Purity	Recall
no-motion	75	96	74	96
road1	92	40	38	51
road2	25	64	19	69
cross road	40	74	40	45
side walk	62	56	46	19
road3	5	3	0.1	0.2
road4	66	80	19	10
road5	59	47	68	52
stop place	66	73	32	60
road6	75	75	1.3	0.3
park place	3	0.3	64	5.5
road7	83	47	88	36
road8	0.2	0.3	34	12
road9	85	50	46	55
Total	71	68.3	65	51.2

stop. This causes another region before stop area which is different from other parts of the road. Finally, we also observe that the two labelled cross roads are different based on the direction of flow.

7. CONCLUSION

We propose an unsupervised clustering of meaningful regions and some semantic landmarks by modelling the pattern movements in the scene. To understand scene motion, we employ optical flow because we need to know exact magnitude and direction at each pixel. Then by defining some distinguishing features like the size of objects and the velocity of flow on each pixel and using LDA, we could determine different parts in the video as side-walk, road, cross-road and so on. We also quantitatively compared the results of our algorithm with GMM as a baseline algorithm which shows that the proposed algorithm works better specially in finding semantic regions like cross road. In addition, by tracking one object in two frames and camera calibration, we removed distortions in our defined motion features caused by perspective.

Our work provides a basis for unsupervised clustering of regions within scenes by observing the motion of objects in the scene. We hope to build on this work to gain a better understanding of activities by linking motion behaviour and location within a scene.



(a) First frame,video1 (b) Labels (c) Clustering results

(d) First frame,video2 (e) Labels (f) Clustering results

Figure 8: Annotated ground truth and our method's results for both videos. (a) and (d) show the first frame of each video, (b) and (e) show ground truth and manually annotated labels, and finally (c) and (f) show the clustering results of our method on both videos.

References

[1] Virat video dataset. `http://www.viratdata.org/`, 2011.

[2] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[3] W. Cao, Y. Yan, and S. Li. Unsupervised color-texture image segmentation based on a new clustering method. *JNIT*, 1(2):96–102, 2010.

[4] N. J. Carlos, W. Hongcheng, and F.-F. Li. Unsupervised learning of human action categories using spatial-temporal words. *IJCV*, 79(3):299–318, 2008.

[5] A. Criminisi, I. D. Reid, and A. Zisserman. Single view metrology. *IJCV*, 40(2):123–148, 2000.

[6] L. Fei-Fei and P. Perona. A bayesian hierarchical model for learning natural scene categories. *CVPR*, pages 524–531, 2005.

[7] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004.

[8] B. K. P. Horn and B. G. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.

[9] T. M. Hospedales, S. Gong, and T. Xiang. A markov clustering topic model for mining behaviour in video. In *ICCV*, pages 1165–1172, 2009.

[10] C. Li and Y. Zhao. Camera self-calibration method by using three orthogonal vanishing points. *AISS: Advances in Information Sciences and Service Sciences*, 3(8):45–52, 2011.

[11] X.-H. Phan and C.-T. Nguyen. Gibbslda++: A c/c++ implementation of latent dirichlet allocation (lda). `http://gibbslda.sourceforge.net/`, 2007.

[12] I. Saleemi, K. Shafique, and M. Shah. Probabilistic modeling of scene dynamics for applications in visual surveillance. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(8):1472–1485, 2009.

[13] J. Seetha, R. Varadharajan, and V. Vaithiyanathan. Unsupervised learning algorithm for color texture segmentation based multiscale image fusion. *EJSR*, 67(4), 2012.

[14] S. N. Sinha and M. Pollefeys. Pan-tilt-zoom camera calibration and high-resolution mosaic generation. *Comput. Vis. Image Underst.*, 103:170–183, 2006.

[15] J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman, and W. T. Freeman. Discovering object categories in image collections. In *Proceedings of the International Conference on Computer Vision*, 2005.

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

[17] X. Wang and E. Grimson. Spatial latent dirichlet allocation. In *NIPS*, 2007.

[18] X. Wang, K. Tieu, and E. Grimson. Learning semantic scene models by trajectory analysis. In *In ECCV (3)*, pages 110–123, 2006.

[19] W. Zhang, X. Fang, X. K. Yang, and Q. M. J. Wu. Moving cast shadows detection using ratio edge. *IEEE Transactions on Multimedia*, 9(6):1202–1214, 2007.