Author/s:
Rao, AS;Gubbi, J;Marusic, S;Stanley, P;Palaniswami, M

Title:
Crowd Density Estimation Based on Optical Flow and Hierarchical Clustering

Date:
2013-01-01

Citation:
Rao, A. S., Gubbi, J., Marusic, S., Stanley, P. & Palaniswami, M. (2013). Crowd Density Estimation Based on Optical Flow and Hierarchical Clustering. 2013 INTERNATIONAL CONFERENCE ON ADVANCES IN COMPUTING, COMMUNICATIONS AND INFORMATICS (ICACCI), pp.494-499. IEEE. https://doi.org/10.1109/ICACCI.2013.6637221.

Persistent Link:
https://hdl.handle.net/11343/313850

Crowd Density Estimation Based on Optical Flow and Hierarchical Clustering

Aravinda S. Rao*, Jayavardhana Gubbi*, Slaven Marusic*, Paul Stanley† and Marimuthu Palaniswami*
* Department of Electrical and Electronic Engineering, The University of Melbourne,
Parkville Campus, VIC - 3010, Australia.
aravinda@student.unimelb.edu.au, {jgl, slaven, palani}@unimelb.edu.au
† ARUP, Melbourne, VIC - 3000, Australia.
Paul.Stanley@arup.com.au

***Abstract*—Crowd density estimation has gained much attention from researchers recently due to availability of low cost cameras and communication bandwidth. In video surveillance applications, counting people and creating a temporal profile is of high interest. Surveillance systems face difficulties in detecting motion from the scene due to varying environmental conditions and occlusion. Instead of detecting and tracking individual person, density estimation is an approximate method to count people. The approximation is often more accurate than individual tracking in occluded scenarios. In this work, a new technique to estimate crowd density is proposed. A block-based dense optical flow with spatial and temporal filtering is used to obtain velocities in order to infer the locations of objects in crowded scenarios. Furthermore, a hierarchical clustering is employed to cluster the objects based on Euclidean distance metric. The Cophenetic correlation coefficient for the clusters highlighted the fact that our preprocessing and localizing of object movements form hierarchical clusters that are structured well with reasonable accuracy without temporal post-processing.**

I. Introduction

Crowd density estimation has recently gained significant attention from researchers in the field of computer vision. Growing population and urbanization has mobilized the day-to-day activities and consequently, people endeavor to participate more often in public events. More often than not, this lead to scenarios such as stampede, people crushing and unruly crowd behavior at large events. Monitoring of such activities leads to adopting mitigating strategies thereby avoiding crowd related disasters. Computer vision techniques can be used for such purposes due to wide availability of cameras. People detection and tracking have been widely adopted for monitoring abnormal activities. People counting has also been an integral part of surveillance and security. Estimating people at public places such as airports, stadia, shopping malls, transport facilities etc., are some of the applications of people counting. In the event of emergency such as fire, stampede, eruption of violence etc., we require an estimate of people currently in different spaces of the arena. Albeit people detection and tracking systems exist, counting people by detecting individuals and tracking is not necessarily required for this type of application and often fails in occluded scenarios. The primary objective of resolving the mishaps in the buildings and at public spaces are to have an estimate of number people rather than to have an exact figure.

Surveillance systems face difficulties in detecting motion from the scene due to varying environmental conditions. Nonuniform illumination, cast shadows, nonrigid human movements and occlusions are some of the major challenges [1]. In the first stage, to eliminate noise and extract moving objects either background subtraction method or motion estimation based on optical flow is practised. Background subtraction considers a background model of the scene for a particular view and subtracts this model from incoming video frames to extract the foreground pixels. Mixture of Gaussian (MOG) [2], [3] based background modeling is widely adopted. There exist several modified versions of MoG. On the other hand, optical flow uses motion estimation based on apparent motion of objects instead of explicitly modeling the background. Horn-Shunck [4] and Lucas-Kanade [5] are the two major versions of optical flow.

In this paper, Horn-Shunck [4] optical flow approach is used for motion detection and estimation as a part of preprocessing stage. The calculated dense optical flow of the frame is divided into blocks for block-based relative flow analysis. Later, the flow analysis in the region of frame is performed, where in density need to be estimated. Further, hierarchical clustering is used to cluster the motion information with each cluster representing the estimate of presence of an object in a given area. The objective is to estimate the density of crowd and contributions in this paper are:

- Use of block-based optical flow approach for density estimation as opposed to background subtraction where foreground pixels are chiefly used.
- Density estimation is based on unsupervised approach - to provide an estimate of number of people in a given area using optical flow and clustering, without using any supervised training approach.

The paper is organized as follows: Section II provides an overview of existing methods and their approaches. Section III provides necessary details of our proposed approach, Section IV provides the results of the proposed approach and finally, our work is concluded in Section V.

II. Related Work

Most of the crowd density estimation use either the texture features on local and global level or extract foreground pixels using motion information. The extracted features or the foreground pixels are then mapped to the density in a given area or region. Rahmalan *et al.* [6] used texture features—Gray Level Dependence Matrix (GLDM), Minkowski Fractal Dimension

(MFD) and a new technique termed as Translation Invariant Orthonormal Chebyshev Moments (TIOCM)—to classify the crowd density into five classes (very low, low, moderate, high and very high) using Self Organizing Maps(SOM). Wu *et al.* [7] utilized texture features (GLDM) at local and global level, scale normalized for perspective correction and Support Vector Machine (SVM) for abnormal crowd density detection. Aijun *et al.* [8] used Kanade-Lucas-Tomasi (KLT) [5] tracking features to estimate the crowd density in public venues. The KLT features were tracked, re-spawned, conditioned and clustered to identify individual objects. Ma and Bai [9] used four texture features (contrast, homogeneity, energy and entropy) and ν-Support Vector Regression (ν-SVR) for high crowd density estimation. The four statistical measure provide a 16-feature vector. However, this approach requires to tune the ν parameter for video sequences during training. Ma *et al.* [10] used texture features for crowd density estimation. Each frame is divided into patches and further, to extract texture features, Gradient Orientation Co-occurrence Matrix (GOCM) was proposed. The GOCM features were used for visual vocabulary construction. However, all these methods require training to estimate density.

On the other hand, model-based approaches estimate parameters to estimate density. Mao *et al.* [11] estimate the density using eight low-level features and regression analysis. A low-complex background model was used for background subtraction. Eight features (blob area, Harris corner, KLT feature points, contour number, contour perimeter, ratio of contour perimeter to area, Canny edge and fractal dimension) were extracted. Furthermore, perspective and occlusion corrections were made. To estimate the crowd density, multivariable linear regression model was used on feature inputs. Hou and Pang [12] used foreground pixels and neural network for density estimation. Based on three different on the morphological operations on extracted foreground pixels, the crowd density was estimated with the use of neural network. Guo *et al.* [13] used optical flow and Markov process for crowd density estimation. Optical flow was used for motion information and later the noise was removed. The position of the objects in the crowd are modeled as Markov Random Field (MRF). The density of the crowd is estimated using neighbors by applying least-squares method.

Hsu *et al.* [14] used motion frequency to estimate the density. Using cell approach, Discrete Cosine Transform (DCT) was used to analyze the frequency of moving objects. Six feature vectors are extracted from DCT coefficients. Further, SVM was used for training and classifying the estimated density into five categories: very low density, low density, moderate density, high density and very high density. Srivastava *et al.* [15] estimated the crowd flow density by accumulating foreground pixels over time. Using a scaling factor based on number of pixels required to represent a person in the given strip (area), the foreground pixels over time reveal the number of people traversed across the region. Four Gray Level Co-occurrence matrix (GLCM) were created based four orientations : $0°$, $45°$, $90°$ and $135°$. In addition, four statistical features (energy, entropy, homogeneity, and contrast) for each of the GLCM matrices were extracted. Again, both of these methods require training for density estimation.

Rodriguez *et al.* [16] consider the high density scenes as global approach considering the scene geometry instead of localizing the individual person and tracking. This is formulated as energy minimization problem by jointly optimizing the individuals' detection and estimate of the crowd density. These parameters are learned from training images. Xiong *et al.* [17] approached the problem of crowd density based on image potential energy. The system uses adaptive Gaussian Mixture Model (GMM) for background modeling and consequently extract foreground pixels. The image potential energy is calculated based on object distance from the camera and occlusion factor. It is clear from the above review that most of the methods use training and texture features estimate density. Chen et al. [18] use quantized optical flow directions and Euclidean distance among feature points to form clusters. In contrast, the proposed approach uses optical flow magnitude to detect the presence of objects and hierarchical clustering to estimate density.

III. Methodology

This section details the proposed approach to crowd density estimation. The width of frames in videos are of the size of 640×480. To start with, let $I(x, y, t)$ represent the video frame with x, y corresponding to the coordinates of the pixels and t representing the time. The flowchart of the proposed approach is shown in figure Fig. 1.

A. Preprocessing

Video data often contains high-frequency noise information. The presence of high-frequency noises cause the low-frequency motion information to be undetected. At first, the video sequence is fed to a subroutine that creates a structure for video data. Each of the video frames are converted from RGB to grayscale for processing. In order to handle noise information, a 2D Gaussian filter was designed over the entire image with a standard deviation of $\sigma = 0.5$ and a block size of 5×5. These parameters were chosen such that we do not loose complete edge information and at the same time keeping low-frequency information.

B. Motion Estimation

After obtaining the grayscale images, dense optical flow between two frames are computed using Horn-Shunck [4]. In this method, irradiance of the scene is assumed to be constant while determining the optical flow. The optical flow between two frames is given by:

$$O := \{x + iy : x, y \in \mathbb{R}\} \in \mathbb{C}^{m \times n} \tag{1}$$

where $m, n \in \mathbb{R}$ and $i = \sqrt{-1}$. Motion was estimated by considering every 5th frame.

C. Filtering

The optical flow obtained corresponds to horizontal and vertical velocities of objects in the scene. The resultant magnitude of the velocities is computed by:

$$\text{mag} := \{(x^2 + y^2)^{\frac{1}{2}}\} \in \mathbb{R}^{m \times n} \tag{2}$$

1) Spatial Filtering: A median filter of 5×5 is used to filter out the noise. The resultant optical flow is divided into blocks of size $b \times b$. For each block, a single optical flow magnitude is assigned by computing the median of all the flow values in that block. This is continued for all the blocks of the frame. Similarly, the optical flow for up to t seconds is computed and all of them are processed and stored as aforementioned.

2) Temporal Filtering: For each block, the maximum of all the values along time axis is computed from the stored information. This results in a map of spatio-temporal activity. This step is important because, often, if only instantaneous optical flow information of two frames is considered, the time motion information will be discontinuous due to object movements and occlusion. This acts like a sliding window along the time axis.

D. Hierarchical Clustering and Density Estimation

Hierarchical clustering is an unsupervised approach as compared to other clustering algorithms such as k-means. Hence, the aim of Hierarchical clustering is find out the number of clusters based on feature input matrix. In this case, the feature input matrix is the spatio-temporal activity. Now that a spatio-temporal information about the movement is available, only the region of interest in the scene is considered and rest of the regions are masked. After masking, the region of interest contains the motion activities pertaining to objects with some blocks delineating high activities and others low. The observation is that the center of objects posses peak values and decreasing as we move away from the center. Similarly, if there are several objects in the region of interest, several peaks separated by low values are available. Since each peak approximately corresponds to individual objects, a hierarchical clustering algorithm is designed to cluster the masked values. There are three main steps in clustering: (a) find the distance among the masked values, and (b) group the objects (two nodes) with similar distances, and (c) continue till all the nodes have been grouped. We used Euclidean distance among masked values for calculating distances. Density (number of people) in a given area is mapped to number of distinct clusters that can obtained from the clustering algorithm. From our observation, the distinct peaks correspond to individual objects. Thus, density using both spatial and temporal information is determined.

IV. Results and Discussion

The proposed method was tested using 2 separate videos (varying object sizes) collected at the Melbourne Cricket Ground (MCG). The videos included crowded scenarios. The implementations were performed in MATLAB 8.0 using Computer Vision System Toolbox on Windows XP (SP2 Professional, 32-bit system) equipped with an Intel® i7 – 2600 CPU running at 3.4 GHz. The system also included 512 MB ATI Radeon™ HD 5450 Graphics card. In this work, we have compared our results with ground truth. In the proposed method, optical flow for 0.2s is calculated, which is equivalent to skipping 5 frames and find maximum of optical flow temporally for up to 5 frames ($t = 1$s). The frame number, cophenetic correlation coefficient of the clusters and accuracy in estimating the density are provided in Table I and Table II respectively for Video 1 and Video 2. Cophenetic correlation coefficient (CCC) is a measure indicating as to how well the cluster is formed using the distance metric. A CCC of closer to 1 provides better clustering. The result for Video 1 are shown in Figs. 2, 3, 4 and 5 for no-object, single-object, two-object and four-object scenarios respectively. Similarly, Fig. 6 provides the result for three-object scenario. Table II has another result for Video 2. The output is similar to Fig. 6 result, hence, the only the result is provided in Table reftab:table2. In Video 1, the block size was set to 32×32 and in case of Video 2, it was set to 16×16 in order to cater to varying object sizes. In Fig. 2, Fig. 2-(a) indicates the region of interest (ROI) where we would like to estimate the crowd density; Fig. 2-(b) is the RGB frame; Fig. 2-(c) is the masked output after calculating optical flow and performing spatial and temporal filtering; Fig. 2-(d) is the motion (velocity) map obtained corresponding to optical flow and filtering; Fig. 2-(e) is the figure indicating the cluster distances after hierarchical clustering—only distinct distances (clusters) are used to map to number of people as an estimate of density; and Fig. 2-(f) is corresponding dendrogram output for clusters (x-axis indicates the object indices and y-axis the distance from that object and each object is made of two nodes). Similar description hold for Figs. 3, 4, 5 and 6. From the results it is evident that the block-based optical flow analysis coupled with spatio-temporal filtering provides us motion information of crowded scenarios. The accuracy in the first video is better than the second video.

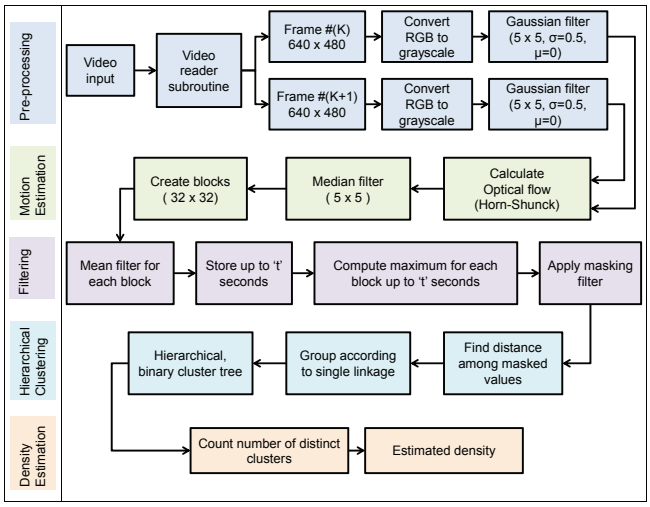

Fig. 1: Flowchart of our approach to density estimation

TABLE I: Analysis of Video 1

Frame Number	CCC	Ground Truth	Estimated	Accuracy
150	0.998	0	0	100%
165	0.9962	1	1	100%
245	0.9929	2	2	100%
305	0.9862	4	3	75%

• CCC - cophenetic correlation coefficient

TABLE II: Analysis of Video 2

Frame Number	CCC	Ground Truth	Estimated	Accuracy
2140	0.9993	3	2	66.67%
2185	0.9982	3	2	66.67%

• CCC - cophenetic correlation coefficient

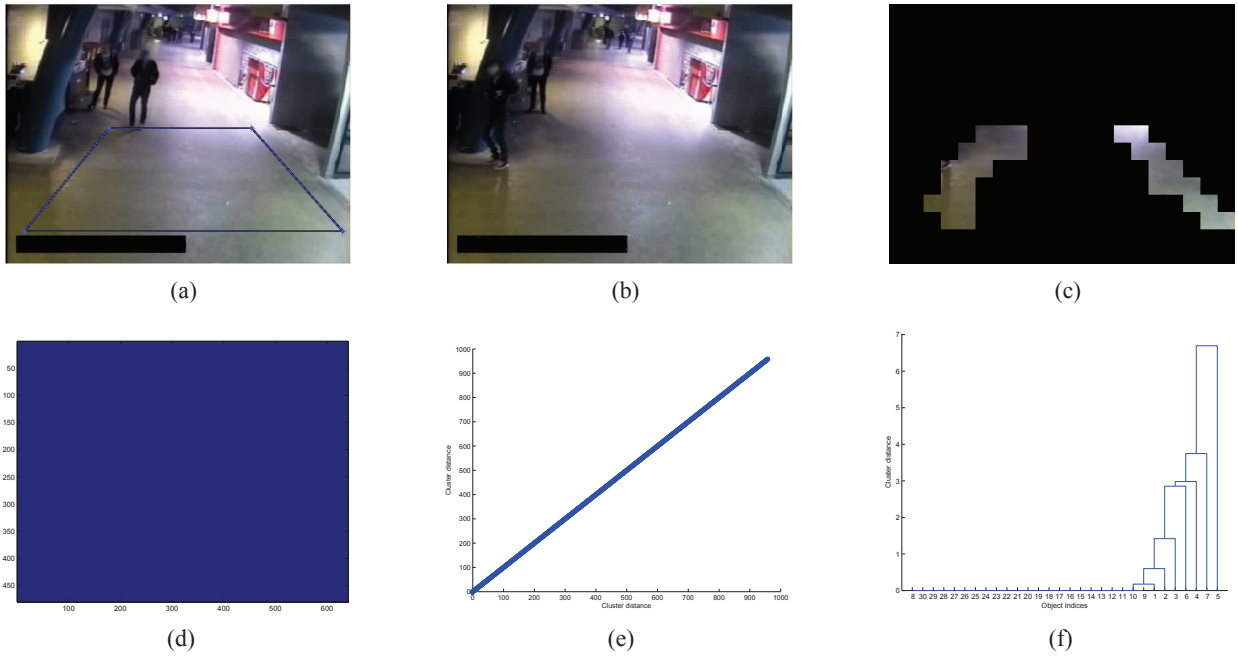

Fig. 2: (a) region of interest (ROI), (b) RGB frame, (c) optical flow output after spatial and temporal filtering, (d) corresponding motion (velocity) map, (e) distinct clusters = estimated number of people = 0, (f) dendrogram output

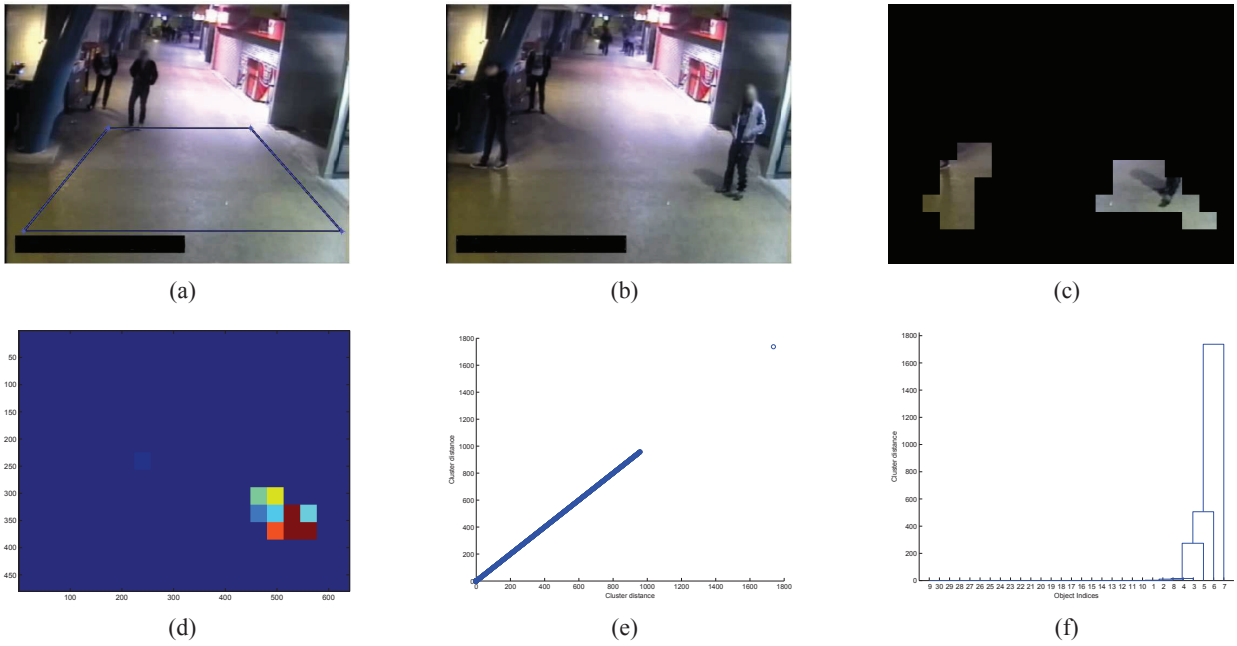

Fig. 3: (a) region of interest (ROI), (b) RGB frame, (c) optical flow output after spatial and temporal filtering, (d) corresponding motion (velocity) map, (e) distinct clusters = estimated number of people = 1, (f) dendrogram output

One of the reasons for this is that in Video 1, because of perspective angle, the objects provide distinguishable optical flow values, whereas in Video 2, the flow velocities would be almost same everywhere. The second reason is that because of camera angle, objects in Video 1 split as they approach camera and hence, they can be separated resulting better accuracy. However, in Video 2, objects move together and provide little information about separation. Further work is required to handle inter-object occlusions and self-occlusions among objects, particularly when the object size is very small.

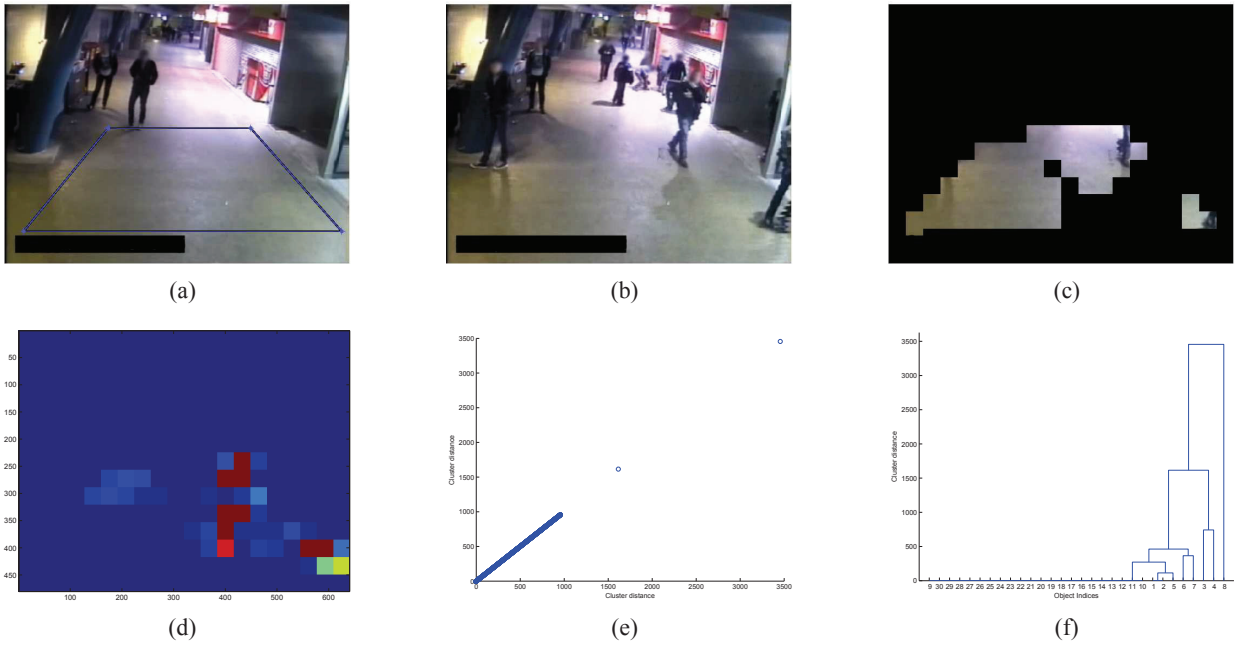

Fig. 4: (a) region of interest (ROI), (b) RGB frame, (c) optical flow output after spatial and temporal filtering, (d) corresponding motion (velocity) map, (e) distinct clusters = estimated number of people = 2, (f) dendrogram output

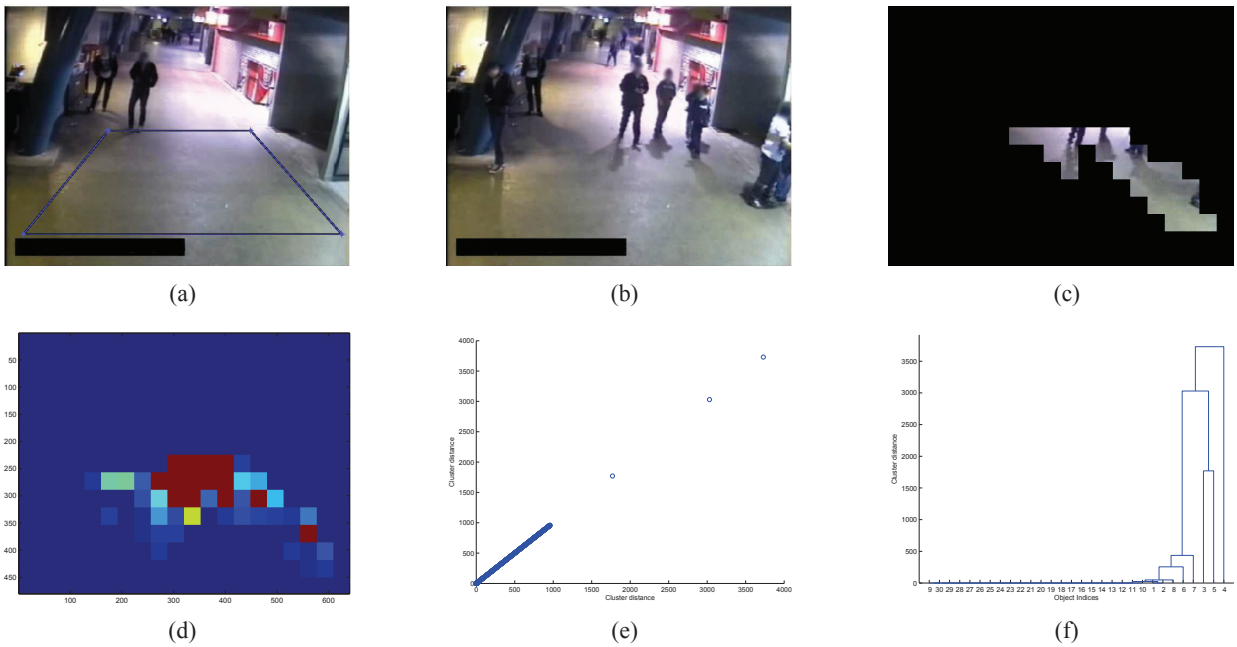

Fig. 5: (a) region of interest (ROI), (b) RGB frame, (c) optical flow output after spatial and temporal filtering, (d) corresponding motion (velocity) map, (e) distinct clusters = estimated number of people = 3, (f) dendrogram output

V. Conclusion

In this work, a new indirect method to estimate crowd density is proposed. A block-based dense optical flow with spatial and temporal filtering is used to obtain velocities that can be used to infer the location of objects among crowded scenarios. Furthermore, a hierarchical clustering to cluster the objects based on Euclidean distance metric is employed. The cophenetic correlation coefficient indicates that our preprocessing and localizing of objects form hierarchical clusters that are structured well. The accuracy of the results indicate that the approach is suitable for obtaining approximate density in crowded scenarios.

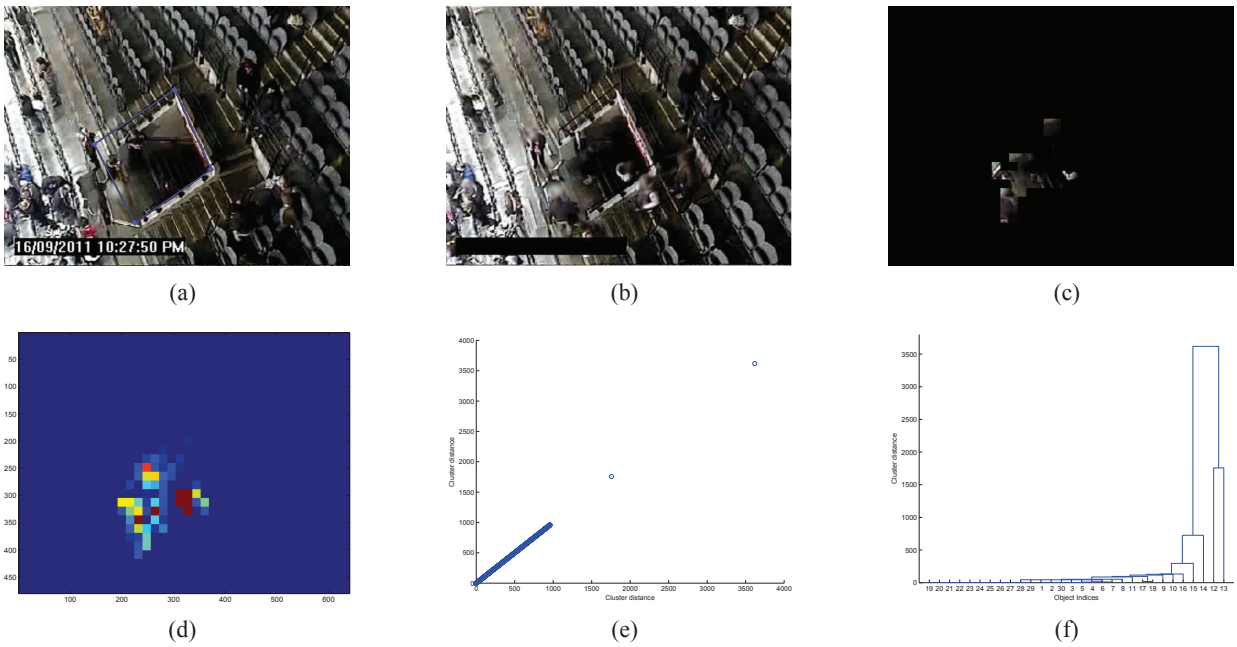

Fig. 6: (a) region of interest (ROI), (b) RGB frame, (c) optical flow output after spatial and temporal filtering, (d) corresponding motion (velocity) map, (e) distinct clusters = estimated number of people = 2, (f) dendrogram output

ACKNOWLEDGMENT

This work is partially supported by the ARC linkage project LP100200430, partnering the University of Melbourne, Melbourne Cricket Club and ARUP. Authors would like to thank representatives and staff of ARUP and MCG.

REFERENCES

[1] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Computing Surveys*, vol. 38, no. 4, pp. 1–45, 2006.

[2] W. E. L. Grimson, C. Stauffer, R. Romano, and L. Lee, "Using adaptive tracking to classify and monitor activities in a site," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 1998, pp. 22–29.

[3] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2. IEEE, 1999, pp. 246–252.

[4] B. K. Horn and B. G. Schunck, "Determining optical flow," Cambridge, MA, USA, Tech. Rep., 1980.

[5] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proceedings of the 7th international joint conference on Artificial intelligence - Volume 2*, ser. IJCAI'81. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1981, pp. 674–679. [Online]. Available: http://dl.acm.org/citation.cfm?id=1623264.1623280

[6] H. Rahmalan, M. S. Nixon, and J. N. Carter, "On crowd density estimation for surveillance," in *The Institution of Engineering and Technology Conference on Crime and Security*. IEEE, 2006, pp. 540–545.

[7] X. Wu, G. Liang, K. K. Lee, and Y. Xu, "Crowd density estimation using texture analysis and learning," in *IEEE International Conference on Robotics and Biomimetics (ROBIO '06)*. IEEE, 2006, pp. 214–219.

[8] S. Aijun, L. Mao, and L. Jianfeng, "Real-time crowd massing risk supervision system based on massing crowd counting in public venue," in *2009 International Symposium on Computer Network and Multimedia Technology (CNMT 2009)*. IEEE, 2009, pp. 1–7.

[9] Y. Ma and G. Bai, "Short term prediction of crowd density using v-svr," in *2010 IEEE Youth Conference on Information Computing and Telecommunications (YC-ICT)*. IEEE, 2010, pp. 234–237.

[10] W. Ma, L. Huang, and C. Liu, "Crowd density analysis using co-occurrence texture features," in *2010 5th International Conference on Computer Sciences and Convergence Information Technology (ICCIT)*. IEEE, 2010, pp. 170–175.

[11] Y. Mao, J. Tong, and W. Xiang, "Estimation of crowd density using multi-local features and regression," in *2010 8th World Congress on Intelligent Control and Automation (WCICA)*. IEEE, 2010, pp. 6295–6300.

[12] Y. li Hou and G. K. H. Pang, "Automated people counting at a mass site," in *IEEE International Conference on Automation and Logistics (ICAL 2008)*. IEEE, 2008, pp. 464–469.

[13] J. Guo, X. Wu, T. Cao, S. Yu, and Y. Xu, "Crowd density estimation via markov random field (mrf)," in *2010 8th World Congress on Intelligent Control and Automation (WCICA)*. IEEE, 2010, pp. 258–263.

[14] W.-L. Hsu, K.-F. Lin, and C.-L. Tsai, "Crowd density estimation based on frequency analysis," in *2011 Seventh International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP)*. IEEE, 2011, pp. 348–351.

[15] S. Srivastava, K. K. Ng, and E. J. Delp, "Crowd flow estimation using multiple visual features for scenes with changing crowd densities," in *2011 8th IEEE International Conference on Advanced Video and Signal-Based Surveillance (AVSS)*. IEEE, 2011, pp. 60–65.

[16] M. Rodriguez, I. Laptev, J. Sivic, and J. Y. Audibert, "Density-aware person detection and tracking in crowds," in *2011 IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2011, pp. 2423–2430.

[17] G. Xiong, X. Wu, J. Cheng, Y.-L. Chen, Y. Ou, and Y. Liu, "Crowd density estimation based on image potential energy model," in *2011 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. IEEE, 2011, pp. 538–543.

[18] D.-Y. Chen and P.-C. Huang, "Motion-based unusual event detection in human crowds," *Journal of Visual Communication and Image Representation*, vol. 22, no. 2, pp. 178 – 186, 2011.