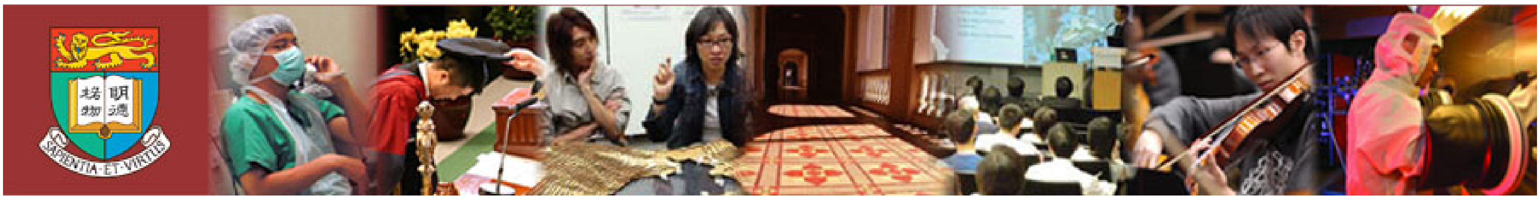

A Novel Object Segmentaion Method for Silhouette Tracker in Video Surveillance Application

Tao LUO, Ronald H. Y. Chung, K. P. Chow
Department of Computer Science
The University of Hong Kong

***Abstract*— In recent years, surveillance cameras are deployed almost everywhere. More and more video analytics features have been developed and incorporated with video surveillance system for conducting intelligence tasks, such as motion detection, human identification, etc. One typical requirement is to track suspicious humans or vehicles in the cameras' live or recorded footages, and over the years researchers have proposed different tracking methods, such as point tracking, kernel tracking and silhouette tracking to support this requirement. In particular, silhouette tracker has received considerable attention because it works well for objects with a large variety of shape, provided that reasonably good object masks or contours are initialized properly for the silhouette tracker. A properly initialized object mask and contour, however, cannot be obtained easily. On one hand, a simple bounding box contains too much irrelevant background objects, while a manually specified mask could provide accurate silhouette but this also requires lots of interactive which greatly limits its practicality. In this paper, we present a novel block based object mask segmentation method for silhouette tracker initialization. Essentially, the proposed method re-uses the motion information extracted during the video encoding phase, which provides approximated object masks for silhouette tracker. Experimental results confirm that such a block-based object masks is sufficient for a robust silhouette tracker to reliably track moving objects.**

Keywords—object silhuette segmentation; motion vectors consistency; graph cuts optimization;

I. Introduction

Nowadays, the prevalence of IP video surveillance system is expanding rapidly. When compared with last two generations surveillance systems, namely analog video tape recorder, and digital video recorder, IP system provides more flexibility and extendibility. Meanwhile, video surveillance system users are more inclined towards systems with analytics functions, since it could help them to reduce manpower on security operations. In other words, some systems prefer to use cameras armed with video analytics module for unmanned monitoring of the scene. A typical video analytics application consists of early vision processing and semantic interpretation, as illustrated in Fig 1.

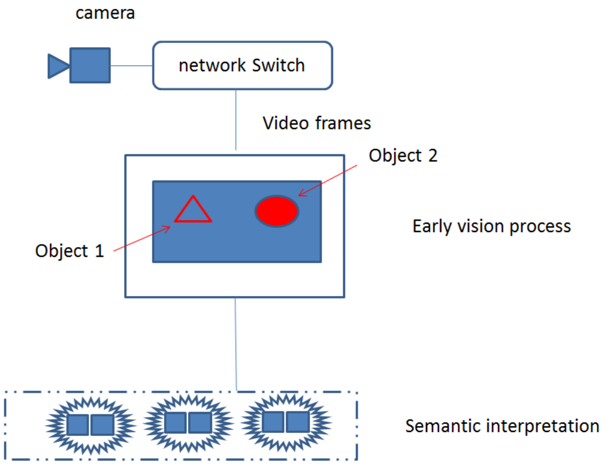

Fig. 1. System architecture for video analytics application

Firstly the video acquired from edge device like IP camera/video encoder, from which video frames are de-noised, de-interlaced, and de-blurred whenever necessary. Thereafter, the frames are decomposed into independent regions, which are tracked as static background objects or moving objects. Lastly, based on these information, semantic interpretation tasks such scene analysis, human behavior identification, etc. could be performed.

The process to decompose scene into independent tracked moving objects or regions is called early vision process, which plays an important foundation for subsequent semantic interpretation process. [1] classifies tracking methods into three categories: point tracking, kernel tracking and silhouette tracking. Since silhouette tracking has the flexibility to handle a large variety of object shapes, it works well in video analytics application, provided that a reasonably good object mask is there for the silhouette tracker to initialize the tracking process for the target object. In other words, the mask plays an important role and neither a simple rectangular mask is sufficient, nor a manually outlined contour is appropriate from practical standpoint. It is because rectangular mask encloses too much irrelevant details, while accurate shape contour outlining will need a lot of manual efforts.

978-1-4799-3010-4/14 $31.00 © 2014 IEEE
DOI 10.1109/CSCI.2014.24

To segment the object mask for silhouette tracker, existing methods can be broadly classified into two approaches: 1) Pixel-wise segmentation; and 2) Block-wise segmentation.

For pixel-wise segmentation it could be further divided into automatic way or interactive way. In automatic way, a background model is automatically learned [2] for the current scene. When an object moves into the view, a pixel-wise object mask could be automatically segmented. As for the interactive way, some pixels are first specified manually as foreground or background pixels, and then the whole image is further segmented as foreground regions and background regions [3]. However these methods are applicable in limited number of scenarios. For the background modeling method, it can be only applied in scenes with static background, which means that the associated cameras must be fixed cameras but not pan-tilt-zoom cameras. In typical video surveillance systems, however, pan-tilt-zoom (PTZ) cameras are unavoidable. Besides, the high computation cost makes it difficult to be applied in real-time application, because if most of resources have been used to for obtaining object masks, not much headroom will be available for subsequent tracking and analysis jobs.

As for block-wise segmentation, it is a natural evolution of block-based video encoding because most, if not all, IP cameras compress video with block-based compression framework such as mpeg2, mpeg4 or h264. Under this framework, motion information – motion vectors, have already been extracted during the video encoding phase and they can be recovered from compressed format at the decoder end. Considerable computation power will be saved if block-wise segmentation of objects can be done based on these readily available motion vectors. [4] reported a seeded region growing based motion vector grouping method for segmentation. However, it could only be applied in the video with static background.

In this paper, we adopted block-wise segmentation approach because it is more favorable from computation efficiency standpoint, while the approximated block-based object boundary is far better than a simple rectangular box. Essentially, the graph cuts based segmentation technique is used, which is a technique commonly used in pixel-wise segmentation methods. Unlike learning-based method, the graph cuts segmentation technique does not assume static background, and so the proposed method here works well with PTZ cameras also.

Graph cuts is a global energy optimization procedure, which has been successfully applied to image restoration, stereo and multi-view reconstruction [5]. In particular, in [6] a matching quality cost function model is built upon two target images and then graph cuts procedure is used to find an optimal seam to stitch the images from different views. A video scene could be regarded as a composite of several objects, moving foreground objects and static background objects. In essence, the object segmentation problem could be regarded as finding the optimal seam among these different objects.

In this paper, we introduce a new matching quality cost function derived from a score which indicates the motion vector consistency from the same object. The score is to evaluate the consistency of moving vectors between different objects, where a higher score indicates the motion vectors are coming from different objects. As a result, the motion vectors from the same object will be grouped as a cut of the graph. To evaluate the method performance, the object segmentation result is feed into a silhouette tracker (PB tracker [7]). PB tracker is a silhouette tracker, which needs a mandatory object mask to initialize interested object and an optional object mask to improve tracking result on each of the subsequent video frame.

This paper is organized as follows: Section II introduces motion vector consistency model and our proposed cost function, Section III describes the proposed object segmentation method, followed by Section IV which presents the results on a lab dataset (podium) and a public dataset, BMSD (Berkeley Motion Segmentation Dataset) [8]. Finally, section V presents conclusions and discusses future research topics.

II. Motion Vector Consistency Model and Proposed Energy Fuction

A. *Motion Vector Consistency Model*

Let $B(m,n)$ denotes the block with block size $W \times W$ in the m-th column and n-th row of the current frame; and let $MV(m,n) = \left[MV_x(m,n), MV_y(m,n)\right]^t$ denotes the motion vector of $B(m,n)$. For any two blocks $B(m,n)$ and $B(m',n')$ in the current frame as depicted in Fig 2, it is shown in [4] that the following condition holds when both blocks $B(m,n)$ and $B(m',n')$ lie on the same object:

$$[m'-m, n'-n][MV(m',n') - MV(m,n)] = 0 \qquad (1)$$

Motion consistency index between two blocks is define as

$$MCI = [m'-m, n'-n][MV(m',n') - MV(m,n)] \qquad (2)$$

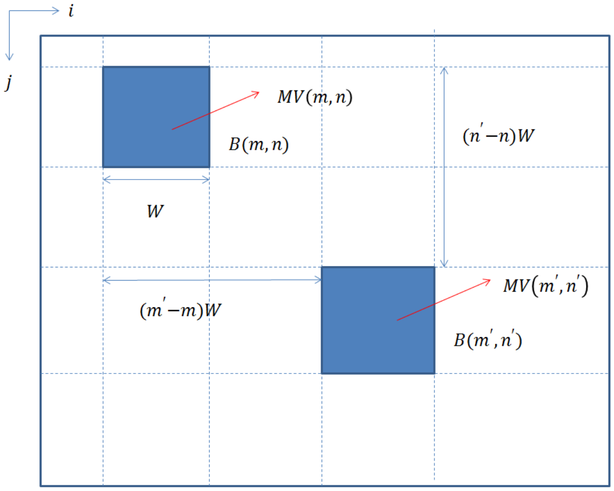

Fig. 2. Two blocks $B(m,n)$ and $B(m',n')$ with motion vectors $MV(m,n)$ and $MV(m',n')$ in the current frame.

B. Proposed Cost Function

The graph cut algorithm works by formulating the image as a connected graph with edge weights based on the difference between neighboring blocks. This graph is then treated as a max-flow/min-cut problem where the sources are any blocks to be taken only from one object and the sinks are any blocks to be taken only from the other object. As illustrated in Fig 3, motion segmentation is a task to find the graph cut between different objects.

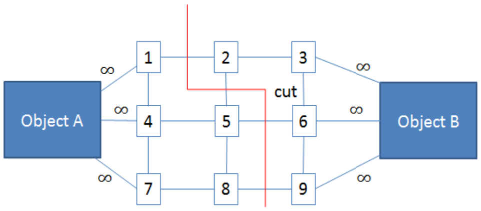

Fig. 3. Connected graph with edge weights based the difference between neighboring blocks. Object A is one of objects, Object B is another object.

For our motion field segmentation, the matching quality cost function is defined as the following:

$$M(s,t,A,B) =$$

$$\|MCI(A,s) - MCI(B,s)\| + \|MCI(A,t) - MCI(B,t)\| \quad (3)$$

s : current block

t : neighboring block

A: candidate object A

B: candidate object B

$MCI(A,x)$: MCI between object A and block x

$MCI(B,x)$: MCI between object B and block x

III. Proposed Method

A. Motion Vector Extraction

Firstly the motion vectors are extracted from compressed video, these motion vectors are estimated at video encoding phase, as illustrated in Fig. 4.

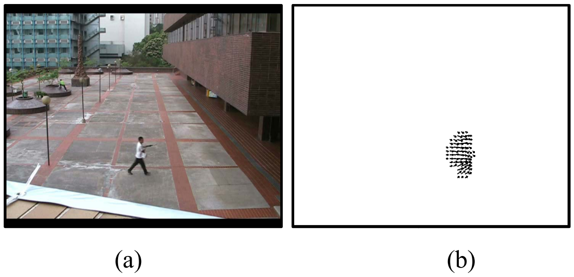

Fig. 4. Motion vectors extracted from compressed video. (a) the 88-th frame on the "podium" data set, (b) the corresponding motion vectors of the whoe image are estimated from video encoding phase.

B. Object Initialization

Followed the motion vectors extraction process, the interested object needs initialized as the source node of our graph cut based method. The object could be selected manually by the user with a simple contour (Fig 5(a)) or automatically from the block with local minimum/maximum motion vectors (Fig 5 (b)).

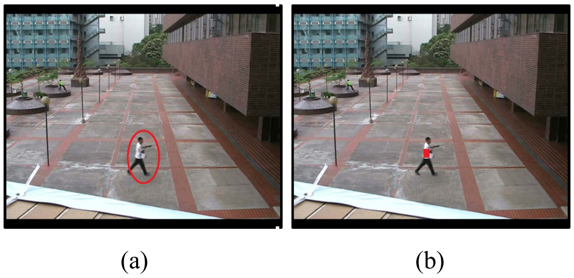

Fig. 5. Object initialization. (a) the object enclosed with a contour manually selected by user (b) auto selected blocks with local minimum/maximum motion vectors.

And then the boundary blocks of the scene will be regarded as the background object, which is the sink node of our graph cut based method. Applying graph cuts optimization procedure with our proposed matching quality cost function (Eq. (3)), the object boundary could be segmented.

IV. Experiment and Discussion

To test our proposed method, we implemented the method in C language and test it on a general PC with Intel i7 2670QM (2.2GHz) CPU, 4G RAM. The testing image sequences are a lab dataset "podium" with resolution 720x576 and people1 from BMSD, in which the video resolution is 640x480.

Firstly the image sequence is encoded into H.264 (MPEG4/Part 10) format video, the motion vector is estimated in block size 8×8 with method suggested in [9]. Although graph cuts optimization process is computationally intensity operation, it is conducted on the block-wise instead of pixel-wise manner and so the data size is much smaller. There are altogether 80x60 blocks in each video frame, and according to our test the motion segmentation execution time ranges from 20 milliseconds to 25 milliseconds, depending on the objects' size. The processing speed is listed in table I.

TABLE I. Processing Speed of Datasets in Terms of Frames per Second (FPS)

Dataset (Sequence Name, Number of objects, Video Resolution)	*FPS*
podium, 1 object, 720x576	50
people 1, 1 object, 640x480	40

A. podium dataset

The dataset has 160 video frames. The scene consists of a single moving people with static background. Fig. 6 shows video images on frame 1, 40, 80, 160

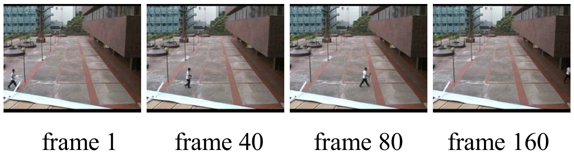

Fig. 6. podium dataset (160 video frames), source images on frame 1, 40, 80 and 160. There is a single man walking through a podium with a static background.

Fig 7 shows the PB tracker results initialized with a simple bounding box on the first frame and no object mask provided in the following frames. While Fig 8 shows the results using our proposed method.

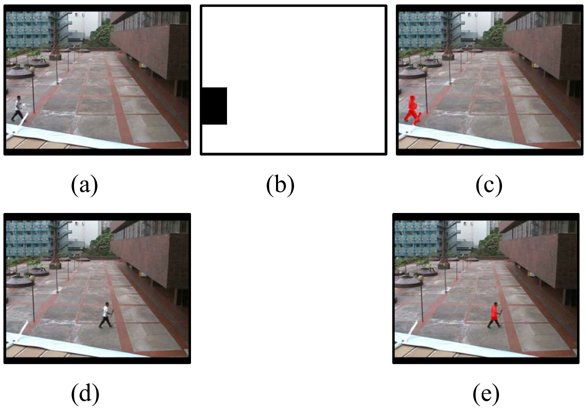

Fig. 7. (a,b,c) a simple bounding box mask is used to initialize the object and PB tracker works on the dataset without per frame object mask. (a) source image of frame 1, (b) the bounding box object to initialize object. (c) the tracker result on frame after initialization. Since the mask contains irrelevant white background object, which is falsely used to initialize insterested object. (d, e) tracking result on frame 80 withut any object mask. (d) source image of frame 80, (e) tracking result of frame 80. Without a provided object mask, only the human body could be tracked.

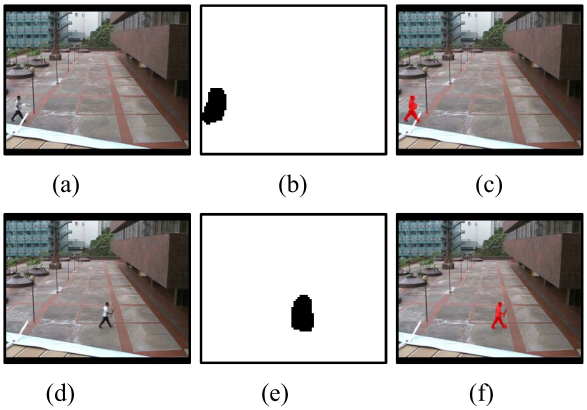

Fig. 8. (a,b,c) object mask to initialize the man on frame 1. (a) source image of frame 1, (b) the obtained object mask from frame 1, (c) the tracker result on frame 1 after intialization (d, e, f) object mask to refine tracking result on frame 80 (d) source image of frame 80, (e) the obtained object mask, (f) tracker result with provided object mask

Compared results of Fig 8 with Fig 9. Without provided object mask, the object is initialized with irrelevant background objects (white lanes), in addition, only the human body could be well tracked in the following video frames. If a proper object mask is provided, the object's head and legs are also well tracked.

B. people 1 dataset

The dataset has 40 video frames. The scene consists of a single moving people with dynamic background Fig. 9 shows video images on frame 1, 10, 20, 40

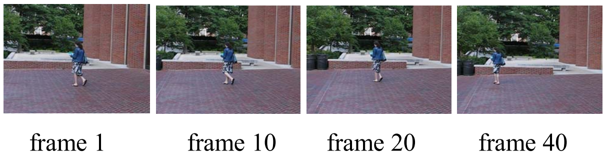

Fig. 9. People 1 dataset (40 video frames), source images on frame 1, 10, 20 and 40. There is a single lady walking through a podium with moving background.

Fig. 10 shows PB tracker results initialized with a simple bounding box on the first frame and no object mask provided in the following frames.

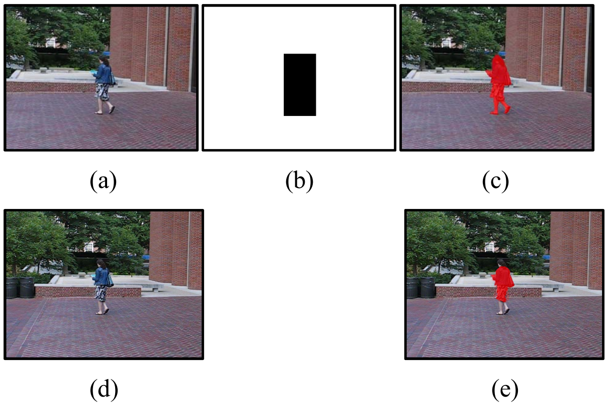

Fig. 10. (a,b,c) a simple bounding box mask is used to initialize the object and PB tracker works on the dataset without per frame object mask. (a) source image of frame 1, (b) the bounding box object to initialize object. (c) the tracker result on frame after initialization. Since the mask contains irrelevant green leaves background object, which is falsely used to initialze interested object. (d, e) tracking result on frame 20 withut any object mask. (d) source image of frame 20, (e) tracking result of frame 20. Without a provided object mask, only the human body could be tracked.

Fig 11 shows using our proposed method, the object mask , Fig 11(a), on frame 1 is used to initialize the object, while Fig 11(e) is the mask on frame 20 to refine the. Without provided object mask, only the human body could be well tracked, the human head and legs lost tracked. Refined with object mask, the head and legs are also could be tracked.

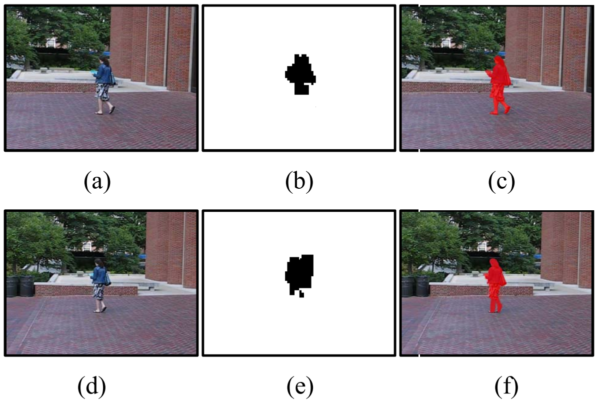

Fig. 11. (a,b,c) object mask to initialize the lady on frame 1. (a) source image of frame 1, (b) the obtained object mask from frame 1, (c) the tracker result on frame 1 after intialization (d, e, f) object mask to refine tracking result on frame 20 (d) source image of frame 20, (e) the obtained object mask. (f) tracker result with provided object mask.

Compared results of Fig 10 with Fig 11. Without provided object mask, the object is initialized with irrelevant background objects (green leaves), in addition, only the human body could be well tracked in the following video frames. If a proper object mask is provided, the object's head and legs are also well tracked

V. Conclusions

In this paper, a new matching quality cost function is proposed to do object segmentation with graph cuts optimization process. According to our experimental results, notable improvements on tracker performance can be observed. Future directions will be focused on working out a real time implementation of our proposed method on GPU architecture.

References

[1] Alper Yilmaz, Omar Javed and Mubarak Shah, “Object tracking: A survey”. *ACM Computing Surveys*, Vol. 38, No. 4, Article 13, 2006.

[2] Stauffer, C, and Grimson, W., “Adaptive background mixture models for real time tracking. *In Computer Vision and Pattern Recognition*, 1999.

[3] Boykov Y. and Jolly, M.P. “Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images.” *In Proceedings of International Conference on Computer Vision*, Vol. 1, pp. 105-112, July 2001

[4] Ronald H. Y Chung, Francis Y. L. Chin, Kwan-Yee K. Wong, K.P. Chow, T. LUO and Henry S.K. Fung., “Efficient Block-based Motion Segmentation Method using Motion Vector Consistency” *IAPR Conference on Machine Vision Application*, May 16-18, 2005 Tsukuba Science City, Japan.

[5] Boykov Y. Veksler O. and Zabih, R. “Markov random fields with efficient approximation” *In CVPR’ 98.*

[6] Vivek Kwatra, Amo Schodl, Irfan Essa, Greg Turk and Aaron Bobick. “Graphcut Texture: Image and Video Synthesis Using Graph Cuts” *ACM Transactions on Graphics, SIGGRAPH 2003*

[7] Papadakis, N. and Bugeau, A. “Tracking with occlustions via graph cuts”. *IEEE Transactions on Pattern Analysis and Machine Intelligence (Volumn: 33, Issue: 1)* Jan, 2011

[8] Berkeley Motion Segmentation Dataset. (*http://lmb.informatik.uni-freiburg.de/resources/datasets/*)

[9] Chung H.Y, N.H.C. Yung and P.Y.S. Cheung, “Fast motion estimation with search center prediction,” *Optical Engineering – The Journal of SPIE*, Vol. 40, No. 6, June 2001, pp.952-963