Moving Object Tracking Within Surveillance Video Sequences Based On EDContours

Gokhan Ciplak[1], Sedat Telceken[1]

[1] Anadolu University Computer Engineering, Eskisehir, Turkey
gciplak@anadolu.edu.tr , stelceken@anadolu.edu.tr

Abstract

We present a technique to track moving object in a video sequences created by still camera. Contours of each frame is detected using EDContours. Background model obtained with three point estimation technique is created from contour maps of video sequence. Differences between contour map of current frame and background model is our target frame. Post-processing operations are applied to the target frame to reduce noises and moving objects are revealed. Last, objects are inclosed with bounding box. Performance of our work is determined using with bounding boxes, considered as trackers.

1. Introduction

Object tracking is fundamental and challenging area in computer vision. Surveillance systems are largely about tracking and detection of objects. Security requirements in traffic, border land, building guarding and many situation, visual tracking and detection are the fundamental bases of these systems. Noise in frames, complex object motions, scene illumination changes, and real-time system requirements make it challenging in object detection and tracking [1]. Numerous approaches have been developed in tracking area. Object tracking methods range due to the object presentation. Broida and Chellappa [2] have employed Kalman filter to track points. Kernel-based object tracking in [3] presents the tracker, which uses mean shift method, dealing with object histogram. Contour evolution method is introduced in [4] handling with edges within the image and Chen *et al.* [5] have proposed real-time contour tracking. Spatiatemporal moving object detection is also a technique to track moving object as seen in work of Chakraborty *et al.* [6].

In this paper, we have described a technique to track moving object with bounding boxes in video sequences. Video sequences are obtained from still cameras, so frames don't have any camera motions. We have applied our tests on *Walk3*, *Browse1*, and *WalkByShop1front* from CAVIAR dataset [7]. Moving object contours can be described as boundary of objects. We have proposed our overall technique in section 2. Contours of frames are one of the fundamental step and EDContours [8] is explained in section 2.1. Three point estimation technique [9] is applied to create background model in section 2.2. Section 2.3 target frame including moving objects are yielded from differences between model and contour map of current frame. Some post processing techniques are applied to the target frame and bounding box trackers are constructed in section 2.4. Evaluation results are presented in section 3.

2. Proposed method

Several approaches have been developed in object tracking and detection area. Such as Mixture of Gaussians, Eigenbackground, Mean-shift etc. for detections and Kalman filter, State space models, Hough transform etc. for tracking are categorized at survey of Yilmaz *et al.* [1]. Proposed method is under contour evolution category of silhouette tracking.

2.1. Contour map: EDContours

EDContours [8] is fast, parameter-free contour detector. The contour detector works on Edge Drawing Parameter Free (EDPF) [10] at different scale-space representation of an image. EDContours combines results of EDPF with different sigma values. Performance of this detector is very satisfactory since F-measure score of this detector is 0.62.

Detecting contours is the first step in our technique. Contours of both frames used in background model and current frame are yielded from EDContours. These contour maps are gray-scale images.

Figure 1 shows contour map of frame 280 from *Walk3 sequence* from CAVIAR [7]. Boundaries in the frame with walking man in scene are clearly detected this detector.

2.2. Background model

Three point approximation [9] is used to estimate parameters for next situation. Using PERT (Program Evaluation and Review Technique) formula and three point approximation, mean and variance calculations are present at [9]. Chakraborty *et al.* [6] have applied this technique to construct background model for temporal segmentation. We have also applied this method for the same purpose. To construct background model, optimistic, most likely, and pessimistic values for each pixel should be calculated with following equations respectively using motionless N(chosen as 10) frames (f_1, f_1, ..., f_N). Also, we have applied threshold to all contour maps for eliminating low intensity values. The threshold value have been chosen as 30.

$$a = max(f_1, f_2, ..., f_N) \quad (1)$$
$$b = median(f_1, f_2, ..., f_N) \quad (2)$$
$$c = min(f_1, f_2, ..., f_N) \quad (3)$$

According to a, b, and c background model and standard deviation are calculated using the formula below.

$$\hat{f}_b = \frac{a + 4b + c}{6} \quad (4)$$
$$\hat{\sigma}_b = \frac{a - c}{6} \quad (5)$$

(a)

(b)

Figure 1. (a) original frame no. 280 from Walk3 and (b) EDContours of **Fig.1(a)**

Figure 2. Background model from first 10 Walk3 sequence frames after Gaussian blur

Figure 3. Contour map differences between background model (Fig. 2) and frame no. 180 from Walk3 sequence

In Eq. (4), $\hat{f}_b$ denotes our background model. When background model is constructed from N contour map, it is possible that fragmentation may occur between contours. To prevent this situation, we applied Gaussian blur to background model. Figure 2 is a background model constructed with first 10 frame from *Walk3 sequence*.

2.3. Target frame using frame differences

Absolute differences between current frame and model is the basic method to detect foreground object. This gives us changed pixel between two frames. Thus, pixels are classified as foreground or background with respect to difference. We have employed this method to obtain changing pixels between current contour map and background model.

$$|f - \hat{f}_b| > E\hat{\sigma}_b \qquad (6)$$

In Eq. (6), if absolute differences of the pixel between current contour map and background model is greater than E (chosen as 19) times standard deviation of the pixel calculated in Eq. (5), this pixel is assigned as target pixel namely foreground pixel. Otherwise it is background pixel. Result of frame differences for frame no.180 from Walk3 is seen in Figure 3.

2.4. Tracking

As seen in Figure 3, the image has undesired white pixels. These noises can be eliminated with various methods. We have employed median filter and morphological opening methods respectively. Applying median filter with size of 3x3 to the image, small white noises are eliminated but not at all. After median filter, we have also applied morphological opening (Eq. (7)) which is combination of an erosion followed by dilation. The opening method removes noisy pixels from binary images. Figure 4 displays target frame after post processing techniques.

$$B \circ S = (B \ominus S) \oplus S \qquad (7)$$

So far, we have summarized steps for moving object detection. We have also need to determine their bounding boxes for tracking. OpenCV library has been employed to construct bounding boxes. Some frames may also contain noises (Fig. 5) and they need not be tracked. We have extracted sets of points of detected objects within target frames using OpenCV's *findContours* method. We have reimplemented line fit parameters in [11] for set of points using width W and height H as $n \geqslant \frac{-4log(\sqrt{WxH})}{log(p)}$, where *p=0.125*. The set of points have
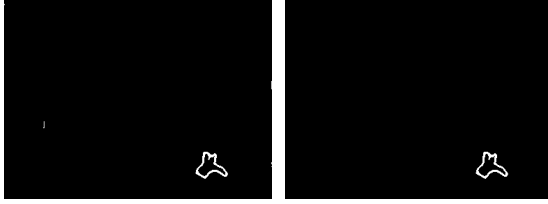
(a) Applying median filter to Fig. 3

(b) Morphological opening to Fig. 4(a)

Figure 4. Post processing operations

been discarded if set size is less than *n*. Applying this formula, small size of detected object wouldn't be tracked. Then, bounding box of each set have determined with OpenCV's *boundingRect* method. Intersected and nearest boxes are merged to cover entire detected object as possible.
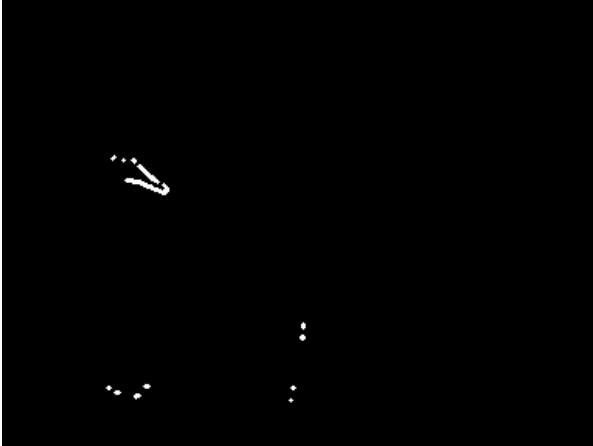

Figure 5. Noises in frame no.1216 from Walk3

3. Experimental Results

In this section, we have evaluated tracker performance. We have run our technique on still cameras, so frames don't have any camera motions. We have applied our tests on *Walk3*, *Browse1*, and *WalkByShop1front* from CAVIAR dataset [7]. Sample datasets used in experiment contain 2086 frames and 1734 boxes, which each one belongs to only one frame according to [12]. We have tracked walking motion within the frames. We have calculated Tracker Detection (TRDR) Rate and False Alarm Rate (FAR) just as in Eq. 8 and Eq. 9 respectively. If the bounding boxes of the targets A_{obs} and A_{truth} overlap at least $T = 50\%$, this is denoted as correct match as in Eq. 10.

$$TRDR = \frac{TP}{TotalGroundtruth} \qquad (8)$$

$$FAR = \frac{FP}{TP + FP} \qquad (9)$$

$$\frac{A_{obs} \bigcap A_{truth}}{A_{obs} \bigcup A_{truth}} > T \qquad (10)$$

Results in table 1 present TRDR and FAR values of each dataset and overall dataset with 50% threshold value. TP and FP are counted as 1048 and 681 respectively and total ground truth

Table 1. TRDR and FAR results of datasets *Walk3*, *Browse1*, and *WalkByShop1front* from CAVIAR with T=50%

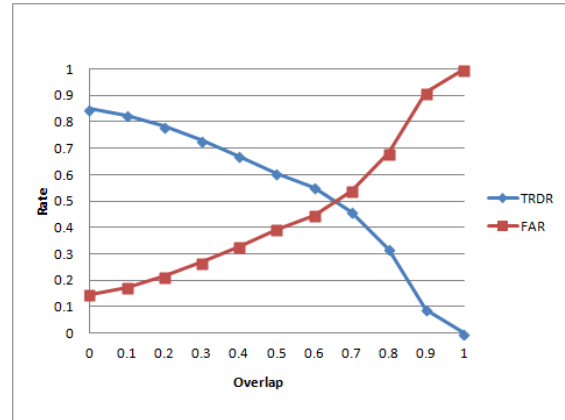|	TP	FP	TRDR	FAR
Browse1	223	115	0.61	0.34
WalkByShop1front	232	136	0.63	0.37
Walk3	593	430	0.60	0.42
Overall Dataset	**1048**	**681**	**0.60**	**0.39**

Figure 6. TRDR and FAR respect to overlap

of these three dataset is 1734. Due to the these values, TRDR and FAR are calculated as 0.60 and 0.39 with 50% overlap. Also, Figure 6 displays TRDR and FAR respectively with different threshold values. FAR begins with 0.15 value and TRDR begins with 0.85 value when overlap threshold equals to 0. The intersection of TRDR and FAR has the overlap threshold value between 0.6 and 0.7. After 0.7 overlap value, increase of slopes of both TRDR and FAR are seen clearly in the Fig. 6. Tracker results on several frames are displayed in Fig. 7.

4. Conclusion and Feature Works

In this paper, we have employed EDContours with three point approximation to track moving object using the dataset which is yielded from still camera. We have tracked single walking motions from the dataset. We have used background subtraction for moving object detection. Morphological opening and median filtering are utilized to reduce image noise. All experiments are carried out with OpenCV. We have evaluated our method respect to TRDR and FAR. Our method has been run with *Walk3*, *Browse1*, and *WalkByShop1front* from CAVIAR dataset. We have solely applied EDContours for tracking.

Remember that EDContours is robust contour detector, it can be used with different segmentation algorithms to generate robust tracker. Trackers with contours can handle various object shapes. Using with texture, it can be employed for foreground subtraction. Since contours have ability for handling various shapes, it can also performed supervised learning for object recognition.

| (a) | (b) | (c) |

Figure 7. a frame no.800, **b** frame no.1615, **c** frame no.280 Browse1, WalkByShop1front, Walk3 CAVIAR dataset respectively. Green box is ground truth and red one is detected.

5. References

[1] Yilmaz, Alper, Omar Javed, and Mubarak Shah. "Object tracking: A survey." Acm computing surveys (CSUR) 38.4 (2006): 13.

[2] Broida, Ted J., and Rama Chellappa. "Estimation of object motion parameters from noisy images." Pattern Analysis and Machine Intelligence, IEEE Transactions on 1 (1986): 90-99.

[3] Comaniciu, Dorin, Visvanathan Ramesh, and Peter Meer. "Kernel-based object tracking." Pattern Analysis and Machine Intelligence, IEEE Transactions on 25.5 (2003): 564-577.

[4] Isard, Michael, and Andrew Blake. "Condensation-conditional density propagation for visual tracking." International journal of computer vision 29.1 (1998): 5-28.

[5] Chen, Yunqiang, Yong Rui, and Thomas S. Huang. "JPDAF based HMM for real-time contour tracking." Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on. Vol. 1. IEEE, 2001.

[6] Chakraborty, Debarati, B. Uma Shankar, and Sankar K. Pal. "Granulation, rough entropy and spatiotemporal moving object detection." Applied Soft Computing 13.9 (2013): 4001-4009.

[7] Caviar dataset. http://homepages.inf.ed.ac.uk/rbf/CAVIAR/

[8] Akinlar, Cuneyt, and Cihan Topal. "EDContours: High-speed parameter-free contour detector using EDPF." Multimedia (ISM), 2012 IEEE International Symposium on. IEEE, 2012.

[9] Keefer, Donald L., and Samuel E. Bodily. "Three-point approximations for continuous random variables." Management Science 29.5 (1983): 595-609.

[10] Akinlar, Cuneyt, and Cihan Topal. "EDPF: a real-time parameter-free edge segment detector with a false detection control." International Journal of Pattern Recognition and Artificial Intelligence 26.01 (2012): 1255002.

[11] Akinlar, Cuneyt, and Cihan Topal. "EDLines: A real-time line segment detector with a false detection control." Pattern Recognition Letters 32.13 (2011): 1633-1642.

[12] Hall, Daniela, et al. "Comparison of target detection algorithms using adaptive background models." Visual Surveillance and Performance Evaluation of Tracking and Surveillance, 2005. 2nd Joint IEEE International Workshop on. IEEE, 2005.