Visual Summarization of foreground object motion using boundary initialization of object tracking

Chaitanya Ahuja, Pratik Somani

Advised by:
Prof. Vinay P. Namboodiri
Prof. K.S. Venkatesh

Abstract

Video Surveillance of hours of long footage is an impractical task. To reduce the time and energy required for analyzing surveillance videos, a synopsis can be synthesised. The synopsis must maintain the causality of the video to prevent giving false information to the viewers. Cases of occlusion cannot be handled well by standard blob tracking and clustering techniques. Keeping this in mind, a new system for synopsis has been proposed which uses a state of the art tracker to avoid the issue of occlusion. Finally some examples are illustrated to demonstrate the effectiveness of the proposed method.

1 Introduction

An important issue pertaining to analysis of surveillance videos is the sheer length of the video. It is practically impossible for one to watch an entire surveillance video to obtain any sort of useful information for a particular object from videos which are, say, of an entire day in length. We wish to have some sort of summary or synopsis of long videos for quick and easy analysis of data for any object in the video.

A lot of work has been put into video synopsis and summarization. [1] and [2] use various methods for superimposing and combining different images to form a single image summary. [3] and [4] use a combination of image fusion with enhancement of the images for several objects which may or may not interact. [5] tries to improve video synopsis with prior clustering of object activities, and displays all those activities which are similar in appearance or motion.

In this project we use automated detection and tracking of objects moving into the surveillance video frame to locate the approximate trajectories of each of the foreground objects. We propose a novel cost-efficient method for detection of bounding boxes of the objects moving in from one of the boundaries. Having found the approximate paths for the bounding boxes and the centroids, we can analyze individually each object for the durations it was in the frame, and come up with a suitable single-image synopsis of the route the object followed.

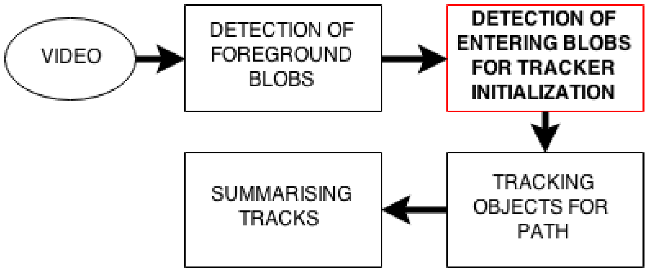
VIDEO
DETECTION OF FOREGROUND BLOBS
DETECTION OF ENTERING BLOBS FOR TRACKER INITIALIZATION
TRACKING OBJECTS FOR PATH
SUMMARISING TRACKS

Figure 1: Flow-Chart illustrating the system proposed for Video-Synopsis

2 Theoretical Basis of Video Synopsis

A thorough understanding of the theory behind each subsystem is essential to the understanding of the complete system, which is explained in the following sections. Section 2.1 gives a description on initial foreground detection. Initialization of Objects is proposed in Section 2.2. Finally Tracking and Synopsis Synthesis are discussed in Sections 2.3 and 2.4 respectively. All these subsystems, if combined as shown in Fig. 1, can be a useful tool for video synopsis.

2.1 Blob Detection

Blob Detection for foreground-background classification has been discussed extensively in [6]. This method follows a local approach for detection of a background. At any given time instance t_i for a patch of the image, point-wise Euclidean distance is calculated with respect to the background model. With the threshold set at R, number of points (= *count*) which have a distance lesser than R is calculated. If *count* is greater than a pre-decided value, the current patch is classified as the background.

Although this method is generalised for a changing background, it still takes a finite number of frames to stabilise. Our methodology would face hiccups in such a scenario and hence we restrict our discussion strictly to non-moving cameras.

2.2 Object Initialization

We present a cost-effective novel technique for detecting objects moving into the frame of the video at any point of time. As described in Section 2.1, we process the entire video to identify the foreground objects and remove the background pixels. All the pixels in the background get marked as *black* whereas the foreground pixels are *white*.

Now, we use a method which is linear in the number of pixels on the boundary of the frame to detect objects coming in. We traverse each pixel on the boundary and determine its color. If the color of the pixels in a neighbourhood on the boundary was *white* in earlier frames, but has now turned *black*, the region corresponds to an object either coming into the frame or moving out of the frame. We check in a neighbourhood perpendicular to a formerly *white* pixel for a region which is currently *white*. If we manage to find such a region, then an object has moved into the frame and now its bounding box can be determined.

The proposed method has been compiled to give a pseudo algorithm, which has been discussed in Section 2.2.1

2.2.1 Edge Detection of Objects- Pseudocode

```
Take pixels on the edges as a linear array $EdgePixels$;
$NumberOfPixels = \text{length}(EdgePixels)$;
$StartPixel = 1$;
Assign all pixels $group$ the value 0;
# $A$ pixel group is the group of white pixels the pixel belongs to
while $StartPixel$ is $white$:
  $StartPixel = (StarPixel - 1)$ % $NumberOfPixels$;
$count$=$StartPixel$;
while ($i \leq (StarPixel - 1)$ % $NumberOfPixels$):
  if $EdgePixels[i]$ is $white$:
    if $EdgePixels[i]$ was $white$ in the previous frame:
      $EdgePixels[i].currgroup = EdgePixels[i].prevgroup$;
    else:
      Search for group in neighbour in a radius;
      if a group is found:
        $EdgePixels[i].currgroup = Neighbour.currgroup$;
      else:
        Assign $EdgePixels[i]$ a new group;
  $count++$;
for all $OldGroups$ which are not in the list of $NewGroups$:
  Check if the group corresponds to an object coming in;
  if yes:
    Initialize tracker using Bounding Box Detection;
return;
```

2.3 Object Tracking

Object Tracking has been used to find out the tracks of different objects for the purpose of synopsis. Tracking based on object model, rather than foreground blob-tracking, is useful in cases of occlusion. Surveillance videos are full of such scenarios and hence tackling this issue becomes imperative. As discussed in [7], Kernel based tracking is a very efficient approach, which has been used as a crucial step for tracking path of objects in our system.

Tracking is performed by predicting a transformation function $f : \mathcal{X} \rightarrow \mathcal{Y}$ to estimate transformation between frames. The optimized transformation is obtained by maximizing the discriminant function $F : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbf{R}$.

$$y = f(x) = \arg\ \max_{y \in \mathcal{Y}} F(x, y) \tag{1}$$

where (x,y) is a labelled example pair. Solving the aforementioned optimization problem along with budget constraints, tracking with occluding objects becomes possible. This leads us to our final objective of summarizing the obtained individual object-tracks.

2.4 Summarizing Object Tracks

The object tracking described in Section 2.3 allows us to obtain the individual trajectories of each of the foreground objects. We now process this information to obtain relevant video synopsis in a single image for each of the individual objects. By superimposing the motion of the foreground object across the width of frame for the entire duration the object was in the frame, we create a snapshot of the entire path traversed by the particular object.

One of the features of this summarization is that the particular object's path can be observed clearly and without any occlusion with other foreground objects. This is obtained by finding the possible regions and time of occlusion, using which we can determine which parts of the object's motion to be displayed in the synopsis to obtain a clean and unobstructed summary of the object's trajectory.

Hence, using this process, we can obtain a single-image summary for the motions each of the objects entering the surveillance video, which provides a useful and elegant tool for analyzing individual objects in long surveillance videos in single images.

3 Results

Videos were recorded and tested upon by the proposed system. The first video involves just the motion of a single person to demonstrate the working nature of the algorithm. It is illustrated in Fig. 2. The second video is more complicated as it consists of 2 people walking in the frame at almost the same time and there are moments of occlusion. Our algorithm succeeds in detecting the occlusion and creates the summary of each object separately. Fig. 3 illustrates a frame from the original video and Fig. 4 & 5 are the working summaries of the two separated objects.

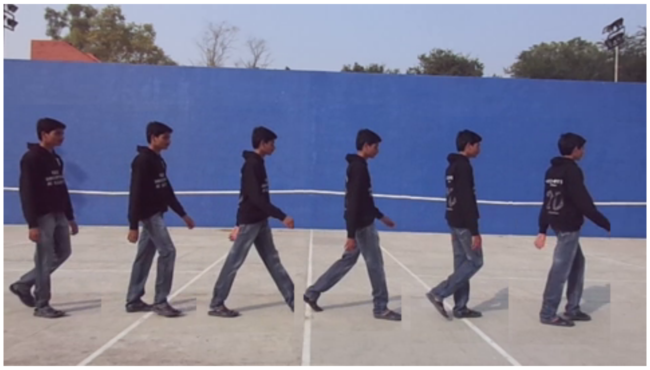

Figure 2: Video 1:Summary of one person walking across the frame

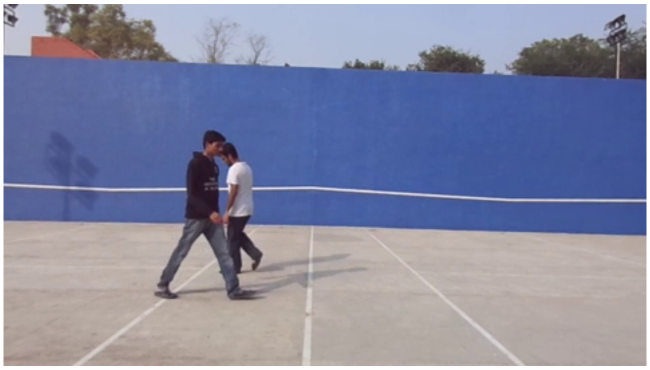

Figure 3: Video 2: A single frame from the original video containing both persons **A** and **B**

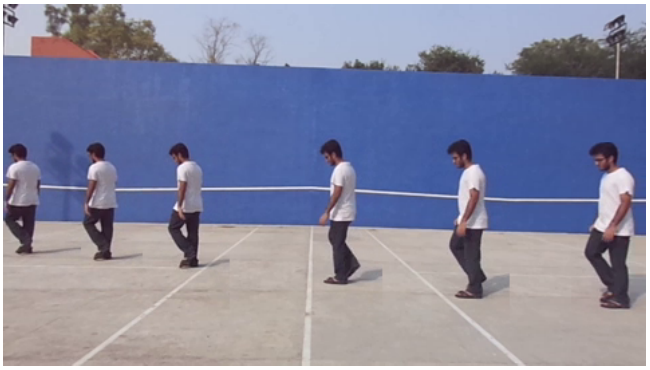

Figure 4: Video 2: Summary of person **A** walking across the frame

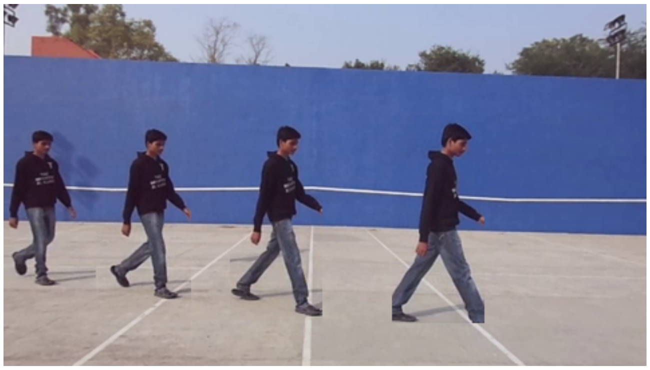

Figure 5: Video 2: Summary of person **B** walking across the frame

4 Conclusions

This project provides a fresh approach at video summarization and extraction of data from long surveillance videos. The fact that the entire synopsis process is automated makes it a viable tool for analyzing video data of any durations.

The techniques used in tracking ensure that all objects are tracked efficiently in spite of occlusion of foreground entities. Hence, our system works in a robust manner even in cases where the objects overlap each other in the frame.

Another feature is the cost-efficient method proposed to detect objects entering the frame using only the data at the boundaries of the frame. This attribute reduces a lot of the computation involved in detecting objects and provides us with reasonable estimations for the time and location of entry of foreground entities.

As an added advantage, this whole process can be done on-line as it requires information of just the previous and the background frames. The background frame is updated after analysing every frame, hence this method is a viable option for on the fly operation.

There are also opportunities for further improvements to the model. The tracker can be modified so as to be more adaptive with respect to the size of the foreground object, which would significantly improve the synopsis of the object motion. Also, using better noise reduction techniques would increase the accuracy of detection of objects entering the frame.

References

[1] Yael Pritch, Alex Rav-Acha, and Shmuel Peleg, "Nonchronological video synopsis and indexing," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 30, no. 11, pp. 1971–1984, 2008.

[2] Alex Rav-Acha, Yael Pritch, and Shmuel Peleg, "Making a long video short: Dynamic video synopsis," in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*. IEEE, 2006, vol. 1, pp. 435–441.

[3] S.S. Thomas, S. Gupta, and K.S. Venkatesh, "Autoensum: Automated enhanced summary for multiple interacting objects," 2014, cited By (since 1996)0.

[4] Kalyan Sunkavalli, Neel Joshi, Sing Bing Kang, Michael F Cohen, and Hanspeter Pfister, "Video snapshots: Creating high-quality images from video clips," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 18, no. 11, pp. 1868–1879, 2012.

[5] Yael Pritch, Sarit Ratovitch, Avishai Hendel, and Shmuel Peleg, "Clustered synopsis of surveillance video," in *Advanced Video and Signal Based Surveillance, 2009. AVSS'09. Sixth IEEE International Conference on*. IEEE, 2009, pp. 195–200.

[6] Olivier Barnich and Marc Van Droogenbroeck, "Vibe: A universal background subtraction algorithm for video sequences," *Image Processing, IEEE Transactions on*, vol. 20, no. 6, pp. 1709–1724, 2011.

[7] Sam Hare, Amir Saffari, and Philip HS Torr, "Struck: Structured output tracking with kernels," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 263–270.