Moving Object Detection and Tracking Based on Background Subtraction

LIU Ya    AI Haizhou    XU Guangyou
Department of Computer Science and Technology, Tsinghua University
State Key Laboratory of Intelligent Technology and Systems

ABSTRACT
An approach to detect and track moving objects with a stationary camera is presented in this paper. The mixture Gaussian model is used as an adaptive background updating method. Based on subtraction foreground is separated from background, and then foreground objects are segmented with a modified binary connected component analysis. Kalman filtering is used in object tracking. To deal with problems caused by occlusions between objects in tracking, six representative categories are introduced and analyzed. Experiments on several outdoors video streams resulted with convicitive object detection and tracking performance demonstrate its strong adaptability to lighting changes, shadows and occlusions.

Keywords:  background subtraction, Kalman filtering, moving object detection and tracking

1. INTRODUCTION
Moving object detection is to extract objects from real-time video streams, and tracking is to establish correspondences between these objects to estimate its state. In most automated visual surveillance systems they are usually the first two processing steps, and they are the start point of all following advanced steps, such as object classification, human motion analysis, and event recognition, etc.

A lot of research work has been done on moving object detection and tracking in several visual surveillance systems recently. Pfinder [1] is a real-time system to track a person in a large room, which uses color and shape to segment a person from the background, then detects and tracks the person’s head. W 4 [3] is another real-time system to detect and track people in an outdoor environment, which operates on grayscale or infrared cameras. It employs a combination of shape analysis and tracking to locate multiple people and their body parts in the scene. CMU’s VSAM system [4] uses a distributed network of active video sensors to monitor activities over a large area. It detects and tracks multiple people and vehicles, classifies them into several categories, and monitors their activities over long periods of time. MIT’s system [5] also uses a distributed set of sensors, in which moving objects are detected by background subtraction and then classified into several classes. Common patterns of activity for different object classes are also learned so that unusual activities can be detected.

In this paper, foreground is separated from background using a mixture Gaussian background model described in [5]. The background model is adapted differently on moving foreground pixels, static foreground pixels and background pixels, so that it is able to catch up with sudden changes in the background while hard to be damaged by moving objects. Foreground objects are then segmented from the foreground based on a modified binary connected component analysis. During moving object tracking, Kalman filters are used to predict each object’s position in the subsequent frames, and a correspondence graph is established between all detected foreground objects and moving objects being tracked. The correspondence graph is then divided into six representative categories and different moving object updating methods are developed for each of them. Occlusions between objects are detected during tracking, and artificial objects are generated if several moving objects are predicted in a large foreground object because of occlusion.

2. BACKGROUND SUBTRACTION
Generally speaking, three approaches are mostly used to detect foreground that represents the region of all moving pixels: temporal differencing, optical flow analysis, and background subtraction. Temporal differencing computes the difference between two or three consecutive frames, and detect foreground by thresholding the differential image. It is adaptive to dynamic environments, but the result is not precise enough because images used in differencing are neither pure background images. Optical flow provides complete motion information, and can be used to distinguish foreground from background, different moving objects, and even different parts of a same object. However, most
optical flow computation methods are too complex to use in real-time applications without special hardware. Background subtraction can extract the most precise foreground region by modeling the background and subtracting frame images from it, but it is sensitive to scene changes due to lighting, weather and sudden events.

In our approach, foreground is detected by background subtraction. The approach presented here is similar to that in [5], and some special processing is performed to make it more responsive to sudden changes in the environment and more robust to moving objects. The key idea of background subtraction is to maintain an evolving statistical model of the background, and to provide a mechanism to adapt to changes in the scene. There are two types of background model, unimodal background and multimodal background.

In a unimodal background model, each pixel in video stream is modeled with a single statistical probability distribution, usually a Gaussian distribution \( \eta(x, \mu_t, \Sigma_t) \), where \( \mu_t \) and \( \Sigma_t \) are the mean value and covariance matrix of the distribution at frame \( t \) respectively. Pixels where image colors observed are close enough to the background distributions (called matched) are classified as background points, while those too far away as foreground points. The background model is typically adapted by IIR filtering:

\[
\mu_{t+1} = (1 - \alpha) \cdot \mu_t + \alpha \cdot d_t \tag{1}
\]

\[
\Sigma_{t+1} = (1 - \alpha) \cdot \Sigma_t + \alpha \cdot d_t d_t^T \tag{2}
\]

where \( \alpha \) is a constant called learning rate, specifying how fast (responsive) the background model is adapted to changes.

For a multimodal background model, a mixture of multiple independent distributions is needed to model each pixel. Each distribution is assigned a weight representing its priority, and only the first several distributions with large weights represent background. A pixel is classified as a background point only if the color observed there matches one of the background distributions. A new distribution of the observation should be imported into the background model if none of the distributions matches it; otherwise, weights of all the distributions are to be updated in such a way:

\[
w_{i+1,j} = \begin{cases} 
(1 - \beta) \cdot w_{i,j} + \beta \cdot w_{i,j} & \text{if } i = m \\
(1 - \beta) \cdot w_{i,j} & \text{otherwise} 
\end{cases} \tag{3}
\]

where \( m \) is the index of the matching distribution, and \( \beta \) is called weight learning rate.

The key process in background model adaptation is how to select an appropriate learning rate for each pixel. In our consideration, the following two rules should be complied during background model adaptation: background models should be (i) reaction to changes in the scene as fast as possible and (ii) robust not to be damaged by moving objects. If the background model is updated too slowly to catch up the changes in the environment or too fast on moving objects, errors will occur in the detected foreground. Otherwise problems shown in Fig.1 will occur, where some background is wrongly classified as foreground and foreground as background respectively.

![Fig. 1: Problems due to background model: (a) A region of ground is classified as foreground when a parked car starts moving. (b) A region on a moving car is classified as background behind a pedestrian.](image)
car in Fig 1 (a) will be difficult to disappear due to tiny learning rates on it. With multimodal background models, the “ghost” will disappear eventually, but in a rather long time after all. To solve both the problems illustrated in Fig.1, foreground points are further divided into moving or static according to whether they are on moving or static objects. That is to say, background models should be updated slightly on moving objects in order to reserve the original background model, while greatly on stationary objects and background to make any changes in scenes quickly incorporated into the background model. In this way, the background model will be adapted quickly to the changes in the environment while keeping it from damaged by moving objects at the same time.

3. FOREGROUND OBJECT SEGMENTATION

Individual objects that are called foreground objects are usually segmented from foreground by connectivity analysis or color consistency analysis. Because color consistency is not a reliable hypothesis for moving objects in many cases, most visual surveillance systems separate foreground objects by connectivity component analysis[3,4].

Thresholding in background subtraction always results in a significant level of noise in detected foreground image, which will cause trouble in connectivity analysis, and should be eliminated in a preprocess step. Morphological processing, such as erosion or dilation, can remove noising foreground points or fill noising holes. However, the boundaries of objects are also damaged due to high frequency information there, therefore they should be restored in a postprocess step by projecting the original foreground region onto each segmented region of connectivity. With these additional processes, the influence of noises occurred in background subtraction can be reduced and the boundaries of objects are preserved. The whole algorithm to segment foreground objects is listed as follows:

1. Let \( F \) be the foreground region from which foreground objects are to be extracted.
2. Expand (i.e., dilation) \( F \) to an expanded set \( E \), and contract (i.e., erosion) \( F \) to a contracted set \( C \).
3. Segment foreground objects in \( E \) by connectivity analysis with seeding points from \( C \).
4. Project these object regions onto \( F \), and the intersection regions are taken as detected objects.

Fig. 2: An example for foreground object segmentation. (a) Original foreground image, (b) expanded image of (a) by dilation, (c) contracted image of (a) by erosion, and (d) segmentation result.

An example of this algorithm is shown in Fig. 2. There are some scattered noising foreground points in the original foreground image in (a), and the head of the person is separated from the torso due to a “hole” in the neck. The hole is filled in (b) and the noising foreground points are erased in (c), resulting in an entire neat person in (d).

4. MOVING OBJECT TRACKING

Moving object tracking is the procedure to find the trajectory of each moving object in the video stream, that is to establish the correspondence between different foreground objects of successive image and already known moving objects. In practice, the establishment of this correspondence may be very difficult due to complicated motions, cluttered scenes, multiple objects, occlusions, etc. The tracking algorithm generally consists of three steps: 1) Prediction of moving objects known, 2) Establishing the correspondence graph of moving objects and foreground objects, 3) Updating moving objects.

In this approach, Kalman filtering is used to represent the dynamic features of an object, including position, velocity and acceleration. Here the median point of an object is selected as the reference point of its position, which is believed to result in a more robust position estimate than its centroid [3]. A predicted object is created for each moving object based on Kalman filtering, and then matched with all foreground objects one by one to establish a correspondence graph between them. Features used in object matching are those most relevant to physical motion, such as position, size, and bounding box, etc. Image patterns of objects are also used to make the matching more robust, and to locate occluded objects accurately.
The correspondence graph of all moving objects and foreground objects is then divided into six representative categories, and different moving object updating methods are developed for each of them, which is similar to that in [4]. Each moving object is assigned with a belief value to measure its reliability of tracking. The six categories are as follows:

**Appearing** (0 to 1). If a foreground object that does not match with any moving object known, then a new moving object is thought to be appearing and is assigned a low belief.

**Disappearing** (1 to 0). If a moving object does not match with any foreground object, then it is thought to be disappearing and its belief is decreased.

**Ideal tracking** (1 to 1). One moving object matches with exactly one foreground object, which is the ideal case in tracking. The belief of the moving object is increased, and its dynamics features are updated. Whether the object is moving or static is also recorded for uses in background subtraction of next frame.

**Merging** (n to 1). If multiple moving objects match with a single foreground object, then it is thought that they are to be merged or occlusions occur between them. These moving objects are merged into one if this lasts for several frames and velocities of the moving objects are close enough to each other.

**Splitting** (1 to m). If a single moving object matches with multiple foreground objects, then it is thought to split or some occluded objects are uncovered. If this lasts for several frames, the moving object is split into several ones.

**Complex tracking** (m to n). Multiple moving objects matches with multiple foreground objects. In this paper, the moving objects are simply removed, and some new ones are created from the foreground objects.

The beliefs of all moving objects are checked during tracking, and they are classified into three classes: visible, active, and inactive objects. Typically, a moving object is created as an active but not a visible one in appearing, and its belief is increased gradually in ideal tracking until it become visible, and decreased in disappearing until it become inactive, that is to say, it really disappears.

### 5. EXPERIMENTAL RESULTS

The system is implemented in C++ on a 600 MHz Pentium-III PC. Currently, for color video streams in 320 × 240 resolution, the system runs at 15 fps.

Fig.3 illustrates the result of background subtraction using different adaptation methods. A “ghost” is left when the bicycle moves in the left column. The middle column shows what happens when a pedestrian and a car meet, and the right shows at what time when the “ghost” is about to disappear. It is clear that the adaptation method in our approach can make background models more adaptive to sudden changes in the scene, while keep themselves robust not damaged by moving objects.

![Fig. 3: Comparison of background model adaptation methods used in [5] and this paper. (a) Frame 8 and 338 of the video; (b)(c) Detected foreground region of unimodal background model in [5] and in this paper; (d)(e) Detected foreground region of multimodal background model in [5] and in this paper.](image-url)
An example of tracking when occlusion happens is illustrated in Fig. 4. The upper row shows some frames of the video, and the lower shows the moving objects being tracked in different colors. When the occlusion happens, an artificial pedestrian and an artificial car are created and placed in proper locations according to prediction and image matching, and the image patterns used for them are taken from the history image lists of the moving objects. It is not analyzed which one occludes the other, in this case the car looks as if it is closer to the camera, while the pedestrian is closer than the car actually.

![Fig. 4: Tracking when occlusion happens. (Artificial objects are created during occlusion for seamless tracking.)](image)

6. CONCLUSION

An algorithm to detect and track moving objects based on background subtraction with a stationary camera is described in this paper. The method consists of three parts: 1) foreground region detection by background subtraction, where the adaptation method is improved to make the algorithm quickly react to scene changes; 2) foreground object segmentation, which separates the foreground points into connected areas as possible objects; 3) moving object tracking, which establishes correspondence between foreground objects at different times, and estimates the states of the moving objects. The effectiveness of the method and its robustness to noise, lighting changes, shadows, and occlusions are demonstrated through several outdoor video streams.

REFERENCES