Research Journal of Applied Sciences, Engineering and Technology 4(24): 5497-5501, 2012
ISSN: 2040-7467

Submitted: March 23, 2012 Accepted: April 20, 2012 Published: December 15, 2012

Object Tracking Using Frame Differencing and Template Matching

[1]N. Prabhakar, [2]V. Vaithiyanathan, [3]Akshaya Prakash Sharma, [4]Anurag Singh and [5]Pulkit Singhal
[1]Scientist, DRDL, Hydrabad, India
[2]Associate Dean(Research), School of Computingm, SASTRA Univesity, Tamilnadu, India
[3]Student, B. Tech-Information and Communication Technology,
SASTRA University, Tamilnadu, India
[4]Student, B. Tech-Information and Communication Technology,
SASTRA University, Tamilnadu, India
[5]Student, B. Tech-Computer Science and Engineering, SASTRA University, Tamilnadu, India

Abstract: The aim of this study is to develop a Robust and reliable object tracking is an efficient way to improve security services. The proposed object tracking system uses concepts of frame differencing and template matching. The frame differencing is used frame-by-frame to detect a moving object in an efficient manner. The moving object detected by frame differencing is tracked by employing an efficient and robust Template Matching algorithm. The templates used for matching purpose are generated dynamically which ensures that the change in orientation and position of object does not hinder the tracking system. This system is highly cost effective and can be used as a surveillance tool in various applications.

Key words: Frame differencing, morphological operation, motion detection, surveillance, template, thresholding

INTRODUCTION

The Object tracking in real-time environment is a difficult task in different computer vision applications (Kettnaker and Zabih, 1999; Collins *et al*., 2001; Greiffenhagen *et al*., 2001). Object tracking consists of estimation of trajectory of moving objects in the sequence of frames generated from a video. Automation of the computer object tracking is a difficult task. The dynamics of multiple parameters changes representing features and motion of the objects and temporary partial or full occlusion of the tracked objects have to be considered (Hanna, 2011). This monograph presents the development of object tracking algorithms, methods and systems and also the state of the art of object tracking methods and also the new trends in this proposed system.

Tracking objects can be complex (Yilmaz *et al*., 2006) due to

- Loss of information caused by projection of the 3D world on a 2D image
- Noise in images
- Complex object motion
- Non rigid or articulated nature of objects (Richard *et al*., 2004)
- Partial and full object occlusions, complex object shapes
- Scene illumination changes
- Real-time processing requirements
- Change in illumination
- Shadows of moving object

This problem of object tracking can be made simple by imposing constraints on the motion and appearance of the object to be tracked. Unlike (Richard *et al*., 2004; Fieguth and Tterzopoulos, 1997), the current system does not require any color information to track an object. Neither there is a constraint for the appearance of the object, which means if the object is little brighter than the background will not affect the output.

The advancement in the computer technologies and the availability of high-quality and inexpensive video cameras leads to great interest in the automated object tracking algorithms. The implementation of object tracking system is based on following key concepts:

- Detection of interested moving objects in a frame
- Such objects are tracked from frame to frame
- Analysis of object tracks to recognize their behavior and trajectory of object can be estimated

Frame differencing: Frame Differencing is a technique where the computer checks the difference between two video frames. If the pixels have changed there apparently was something changing in the image (consider frame). The Frame Differencing Algorithm (Jain and Nagel, 1979;

Corresponding Author: N. Prabhakar, [1]Scientist, DRDL, Hydrabad, India

Haritaoglu *et al*., 2000) is used for this purpose, which gives the position of object as output. This extracted position is then used to extract a rectangular image template (size is dynamic depending upon the dimension of object) from that region of the image (frame). The sequence of templates is generated as object changes its position.

Template matching: The generated templates from each frame are passed on to the tracking module, which starts tracking the object with an input reference template. The module uses template-matching (Richard *et al*., 2004; Comaniciu *et al*., 2003) to search for the input template in the scene grabbed by the camera. A new template is generated if the object is lost while tracking due to change in its appearance and used further. Generations of such templates are dynamic which helps to track the object in a robust manner.

The main objective of this study is to provide a better and enhanced method to find the moving objects in the continuous video frame as well as to track them dynamically using template matching of the desired object. The proposed method is effective in reducing the number of false alarms that may be triggered by a number of reasons such as bad weather or other natural calamity.

Algorithm: The following section contains the analysis and explanation of algorithm used for detection and tracking an interested moving object. The following subsections in this category are:

Object detection using frame differencing: The task to identify moving objects in a video sequence is critical and fundamental for a general object tracking system. For this approach Frame Differencing technique (Jain and Nagel, 1979) is applied to the consecutive frames, which identifies all the moving objects in consecutive frames.

This basic technique employs the image subtraction operator (Rafael and Richard, 2002), which takes two images (or frames) as input and produces the output. This output is simply a third image produced after subtracting the second image pixel values from the first image pixel values. This subtraction is in a single pass. The general operation performed for this purpose is given by:

$$DIFF[i,j] = I_1[i,j] - I_3[i,j]$$

$DIFF[i,j]$ represents the difference image of two frames.

It seems to be a simple process but the challenge occurs is to develop a good frame differencing algorithm for object detection. These challenges can be of any type like

- Due to change in illumination the algorithm should be robust.
- The detection of non-stationary object (like wind, snow etc.) is to be removed.

To overcome such challenges we need to pre-process the $DIFF[i, j]$ image. Pre-processing includes some mathematical morphological operations which results in an efficient difference image. $DIFF[i, j]$ image is first converted into a binary image by using binary threshold and the resultant binary image is processed by morphological operations.

This proposed algorithm for object detection is to achieve these challenges and provide a highly efficient algorithm to maintain such task of object tracking. This algorithm provides the position of the moving object.

Algorithm for object detection:

Assumption: All previous frames are stored in a memory buffer and the current frame in video is F_i

- Take i^{th} frame (F_i) as input.
- Take $(i\text{-}3)^{th}$ frame (F_{i-3}) from the image buffer.

This image buffer is generally a temporary buffer used to store some of previous frames for future use.

- Now, perform Frame Differencing Operation on the i^{th} and $(i\text{-}3)^{th}$ frame. The resultant image generated is represented as:

$$DIFF_i = F_{i-3} - F_i$$

In earlier methods, Frame Differencing is to be performed on i^{th} and $(i\text{-}1)^{th}$ frame, which has limitation to detect slow moving objects. But this proposed method, the Frame Differencing between i^{th} and $(i\text{-}3)^{th}$ frame. This method removes the limitation to detect slow moving object, which makes it independent of speed of moving object and more reliable.

- After the Frame Differencing Operation the Binary Threshold Operation is performed to convert difference image into a binary image with some threshold value and thus the moving object is identified with some irrelevant non-moving pixels due to flickering of camera. And some moving pixels are also there in binary image which corresponds to wind, dust, illusion etc., All these extra pixels should be removed in steps of preprocessing. The binary image (F_{bin}), in which the pixel corresponding to moving object is set to 1 and rest is treated as background which sets to 0.

This threshold technique work as, a brightness Threshold (T) is chosen with the $DIFF[i, j]$ to which threshold is to be applied:

if $DIFF[i, j] >= T$ then
$F_{bin[i, j]} = 1$ //for object
else
$F_{bin[i, j]} = 0$ *//for background*

This assumes that the interested parts are only light objects with a dark background. But for dark object having light background we use:

if $DIFF[i, j] <= T$ then
$F_{bin} = 1$ //for object
else
$F_{bin} = 0$ //for background

The threshold taken here is not fixed it can vary according to our perception. The use of threshold T is just to separate the objects' pixels from the background. Now, perform the Iterative Mathematical Morphological Operation on this binary image F_{bin} .This is to remove all the small particles present in it and ensures us that the all insignificant moving objects are removed. These small particles may come into account because of trees, winds, camera swing etc., which are irrelevant to our aim and should not be treated as moving objects. The resultant image of morphological operation is represented as F_{mor}.

- The next operation is to calculate the Center of Gravity (COG) of the binary objects in image F_{mor}. According to the centroid position a fixed sized rectangular box or a bounding box or perimeter is made for all the binary objects in i^{th} frame F_i.
- All of the centroid information is stored in a global array. By using threads and producer-consumer concept the centroid information of some objects is transferred to the object tracking module.
- Now, follow same procedure for other frame (starting from step 1).

Tracking of object using template matching: The limitation with this tracking module is that all the centroid information received by motion detection module for tracking of objects should always be in camera view. The next operation is to track the only interesting moving object irrespective of other moving objects. The object tracker module is used for this purpose, which keep track of interesting objects over time by locating the position of moving object in every frame of the video. The proposed algorithm has flexibility to perform both task, object detection and to track object instances across frames simultaneously.

First of all tracking module will generate a template for all received centroid information and this template is used for the matching in next upcoming frame. By using mathematical correlation we match the template in next upcoming frame and the centroid of matched area is appended to make an estimation of trajectory and meanwhile the template is updated with new matched region. The coupling of the template matching with the frame differencing gives the good result for Object Tracking System. For this purpose multi-threading and producer consumer concepts are used.

Fig. 1: I^{th} (current) frame F_i, of the continuous video stream

Fig. 2: $(i-3)^{th}$ (previous) frame F_{i-3} of the continuous video stream

Algorithm: Take the centroid coordinates as input from motion detection module.

- For each pair of centroid, generates a template of size 200X300 and each template is assigned with its own stored memory which will have its coordinate in all upcoming frames. Thus giving an estimated trajectory.
- The template matching is as:

If template is matched in next upcoming frame using correlation, then
If match = found, then
Co-ordinate positions are saved and the template has been updated with the match
Else
Template is discarded considering that object has been gone out of camera view.

- Trajectory is plotted whenever it is demanded.

EXPERIMENT AND IMPLEMENTATION

The sequence of following images shows the working and result of object tracking module as described above. Figure 1 and 2, are two frames at i^{th} and $(i-3)^{th}$ sec,

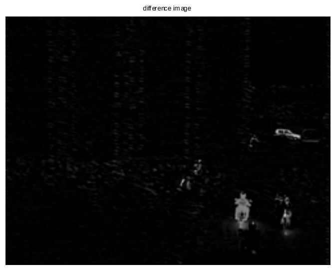

Fig. 3: Difference image DIFF [i, j] obtained by difference of the current and previous frame from continuous video stream

Fig. 4: Image obtained by applying binary thresholding on image obtained in Fig. 3 to obtain binary image

Fig. 5: Image obtained by applying morphological operation (discarding objects with pixels<10) to the binary image

respectively. This sequence of images, taken from a video shows a number of objects that changes its position with respect to time. The motion detection module is being applied to this two frames to obtain the various moving objects.

Figure 3 is obtained by differencing the two previous images viz. Figure 1 and 2. This image contains a lot of disturbances and noise as being observed. This has to be removed before applying further processing. This difference image is converted to binary image using threshold value, such that unwanted pixels are removed. The threshold application on the difference image results to Fig. 4. A series of morphological operations are applied to this binary image to enhance the desired information content of the image.

Figure 5 contains the result obtained after applying morphological operation for discarding the objects with number of pixel less than 10. This removes the noisy content that may have been resulted from slight camera flickering, atmospheric disturbances.

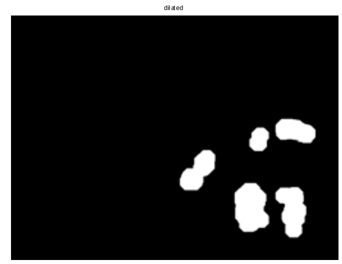

Fig. 6: Image obtained by applying morphological operation (dilation) to the previous image

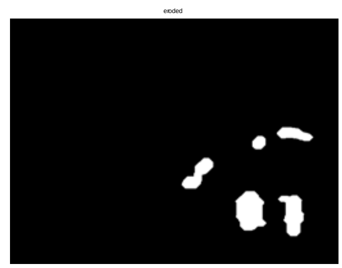

Fig. 7: Image obtained by applying morphological operation (erosion) to the previous image

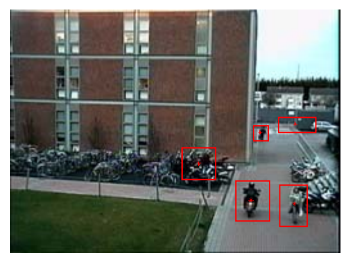

Fig. 8: Moving objects detected with bounding box

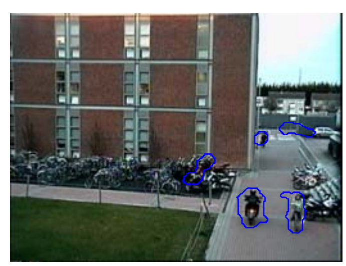

Fig. 9: Moving objects detected using approximate boundary

In order to increase the connectivity dilation is performed. This results in modified movement detection such that all the pixels related to a single object are connected together. The result of application of dilation on the Fig. 6.

To provide detected objects as near to original dimension, erosion is performed just after dilation. The result of erosion is shown in Fig. 7. Then centroids are calculated and stored in memory. For each centroid, a bounding box is found in the original current frame. And

Fig. 10: A sample template of object to be tracked

Fig. 11: Result of object tracking module

hence the moving objects are detected as shown in Fig. 8. Instead of a bounding box, an estimated perimeter can also be found to denote moving objects as shown in Fig. 9.

Figure 10 represents a sample template generated by bounding box from the output of motion detection module. This template will be used to track the object in upcoming video frames.

The template has been matched, in Fig. 11 and the coordinates of centroid of object of interest has been returned and a bounding box has been shown to highlight and track the object.

RESULTS

This study, hence presents an effective way to detect the moving body of interest, discarding the smaller unwanted objects causing false alarms during implementation. Further, this study also presents a way to track an interested object, from a group of moving and detected object, in the subsequent frames of the continuous video stream. This method is simple in implementation and needs very less computational requirements. However, this method is suitably and efficiently applicable to darker objects over light background. For the reverse condition, i.e., for lighter object over dark background, a slight shift in the position of the detected object is being observed.

CONCLUSION

Thus motion detection using frame differencing method provides a better result for the object tracking and which can be easily applied to a number of fields. Further, the study also presents a method to keep the track of movement of object which can be accessed later to analyze the motion.

RECOMMENDATIONS

The algorithm used here is to track the moving object in a video. The proposed system will track the object as long as it can be captured by the stationary camera. But for future study we can use stepper motor to track objects even if they are going out of view by adjusting camera itself. The stepper motor is used to rotate the camera in the direction of the moving object. This can be useful in various surveillance applications where the suspected object is to be tracked as far as possible. And also a separate sub-module for the objects classified on the basis of their contrast (lighter or darker) with respect to their background can be developed in future to improve it's effectiveness a lot.

REFERENCES

Collins, R., A. Lipton, H. Fujiyoshi and T. Kanade, 2001. Algorithms for cooperative multisensor surveillance. Proc. IEEE, 89(10): 1456-1477.

Comaniciu, D., V. Ramesh and P. Meer, 2003. Kernel-based object tracking. IEEE T. Pattern Anal., 25: 564-575.

Fieguth, P. and D. Tterzopoulos, 1997. Color based tracking of heads and other mobile objects at video frame rates. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 17-19 Jun, Waterloo Univ., Ont, pp: 21-27.

Greiffenhagen, D., H. Comaniciu, Niemann and V. Ramesh, 2001. Design, analysis and engineering of video monitoring systems: An approach and a case study. Proc. IEEE, 89(10): 1498-1517.

Hanna, G., 2011. Object Tracking. Hard Cover, Publisher: In Tech, Subject: Artificial Intelligence, pp: 284, ISBN: 978-953-307-360-6.

Haritaoglu, I., D. Harwood and I. Davis, 2000. W4: Realtime surveillance of people and their activities. IEEE T. Pattern Anal., 22(8): 809-830.

Jain, R. and H. Nagel, 1979. On the analysis of accumulative difference pictures from image sequences of real world scenes. IEEE T. Pattern Anal., 1(2): 206-214.

Kettnaker, K. and R. Zabih, 1999. Bayesian multi-camera surveillance. Proceeding of IEEE Conference on Computer Vision and Pattern Recognition, Fort Collins, CO, pp: 253-259.

Rafael, C.G. and E.W. Richard, 2002. Digital Image Processing. 2nd Edn., Prentice Hall International, UK.

Richard, Y., D. Xu, G.A. John and S.J. Jesse, 2004. Robust real-time tracking of non-rigid objects. Proceedings of the Pan-Sydney area workshop on Visual information processing, pp: 95-98.

Yilmaz, A., O. Javed and M. Shah, 2006. Object tracking: A survey. ACM Comput. Surv., 38: 4.