Abstract

This paper is an attempt to develop a system that will automatically track intruding object in predefined area under surveillance. This system can also work in adverse environmental conditions where it is hard for a Human Soldier to fight. Entry-Restricted areas such as Line of Control need to be safe from intrusions. This system will be installed at some suitable place from which complete view of area under surveillance can be captured with camera. A battery powered computer installed on the system will analyze images captured from camera. It will track intrusion and recognize the intruding object by comparing its features with features of the objects stored in database. If feature match is found, the intruding object will get tracked to find its velocity and get bombarded with bullets and bombs until object gets destroyed completely. Thus very tight security and safety can be assured without endangering precious life of Human Soldier.
1. Introduction

The project is proposed for the purpose of intrusion track and destroys the intruding object. The system will be mounted at some suitable place from which complete and clear view of the area under surveillance can be captured with camera. Thus the image will be captured; processed and desired action will be performed on it. Object segmentation separates regions of interest in image data that identify real world objects. Segmenting and tracking regions of arbitrary size within a scene allow the application to focus on more complex tasks like object recognition within a smaller spatial domain of the entire spatial scene which reduces the processing time required to identify the object of interest. Reducing the spatial domain of the image decreases the computational resources necessary for the detailed analyses required for object recognition. The system is provided with a high resolution camera, image processing hardware, microcontroller, two servo motors and other supplementary hardware and mechanisms. Image Processing Hardware will acquire images captured by camera after some predefined interval of time. Then it will process every captured image for detecting intrusion. If intrusion is detected Image Processing Hardware will extract the features of that intruding object and compare them with features of objects stored in database. We have collected database for the objects those are to be destroyed. If match between intruding object and one of the objects from database is found object is said to be recognized. System will track that object to calculate its velocity of motion. This velocity information is needed to decide the angle and time instant at which projectile is to be launched at intruding object to destroy it. Position of the intruding object in the form of x-y co-ordinate is found and sent to microcontroller. Microcontroller will control the angle of rotation of two Servo Motors to position the cannon aiming at the intruding object. At last cannon will get fired.

Object tracking is an important task within the field of computer vision. The proliferation of high-powered computers, the availability of high quality and inexpensive video cameras, and the increasing need for automated video analysis has generated a great deal of interest in object tracking algorithms. In its simplest form, tracking can be defined as the problem of estimating the trajectory of an object in the image plane as it moves around a scene. In other words, a tracker assigns consistent labels to the tracked objects in different frames of a video. Additionally, depending on the tracking domain, a tracker can also provide object centric information, such as orientation, area, or shape of an object. For Object Tracking and Classification basic thing is to obtaining an initial mask for a moving object, and to pre-process the mask. Normally the mask is affected by “salt-and-pepper” noises. We apply morphological filters based on combinations of dilation and erosion to reduce the influence of noise, followed by a connected component analysis for labelling each moving object region.

Very small regions are discarded. At this stage we calculate the following features for each moving object region: bounding rectangle: the smallest isothetic rectangle that contains the object region. We keep record of the coordinate of the upper left position and the lower right position, what also provides size information (width and height of each rectangle).

**Color:** these mean R G B values of the moving object.

**Center:** we use the center of the bounding box as a simple approximation of the centroid of a moving object region.

**Velocity:** It is movement of number of pixels/second in both horizontal and vertical direction. In order to track moving objects accurately, especially when objects are partially occluded, and the
position of the camera is not restricted to any predefined viewing angle, these features are actually insufficient. We have to add further features that are robust and which can also be extracted even if partial occlusion occurs.

One can simplify tracking by imposing constraints on the motion and/or appearance of objects. For example, almost all tracking algorithms assume that the object motion is smooth with no abrupt changes. One can further constrain the object motion to be of constant velocity or constant acceleration based on a priori information. Prior knowledge about the number and the size of objects, or the object appearance and shape, can also be used to simplify the problem.

2. System Overview

This system is proposed for detecting intrusions, tracking intruding object and destroying it. The system will be fixed at some suitable place, from which complete and clear view of the area under surveillance can be captured with camera. The system is provided with a high resolution camera, image processing hardware, microcontroller, two servo motors and other supplementary hardware and mechanisms.

![Block Diagram for Complete System](image)

Figure 2.1: Block Diagram for Complete System

Image Processing Algorithm will acquire images captured by camera after some predefined interval of time. Then it will process every captured image for detecting intrusion. If intrusion is detected Image Processing Hardware will extract the features of that intruding object and compare them with features of objects stored in database. We have collected database for the objects those are to be destroyed. If match between intruding object and one of the objects from database is found object is said to be recognized. System will track that object to calculate its velocity of motion. This velocity information is needed to decide the angle and time instant at which projectile is to be launched at intruding object to destroy it. Position of the intruding object in the form of x-y coordinate is found and sent to microcontroller. Microcontroller will control the angle of rotation of two Servo Motors to position the cannon aiming at the intruding object. At last cannon will get fired.

3. Object Tracking

The goal of the tracking system is to control the camera pan and tilt such that a detected object remains projected at the center of the image. The camera tracking system hardware will include a microcontroller based servo controller that interfaces to the software running on the computer.

The servo controller will adjust the viewing field of the camera by applying the adjustment output as a pulsed width modulated signal to the servo motors. Adjustment output will be calculated by the software in the acquisition feedback loop to center the object found with the specified user
constraints. A camera mount will be fabricated for the camera as well as housing for the servos this allows a range of motion for tracking moving objects.

**Algorithm for Object Tracking**

1. Start
2. Initialize camera and hardware
3. Send adjustment to servo motor
4. Track the moving object
5. Repeat step no 3 to 4

**Flowchart for Object Tracking**

![Flowchart for Object Tracking](image)

**4. Results**

After running the recognition code it will ask for BACKGROUND image. After few seconds it will again ask for TEST image again with the same delay it will ask for the OBJECT image. After capturing these images MATLAB will calculate the results as below.

- **CASE A** when COLOR and SHAPE both not MATCHED
- **CASE B** when COLOR and SHAPE both MATCHED
- **CASE C** when COLOR is MATCHED but SHAPE isn’t MATCHED
- **CASE D** when COLOR isn’t MATCHED but SHAPE is MATCHED

Simulation Results when COLOR and SHAPE both not MATCHED

**INPUT:**

---

**ISSN (Online): 2347-1697**  
*International Journal of Informative & Futuristic Research (IJIFR)*  
*Volume - 2, Issue - 7, March 2015*  
*19th Edition, Page No: 2386-2393*
Simulation Results when both COLOR and SHAPE MATCHED

INPUT:

Simulation Results when COLOR is MATCHED but SHAPE isn’t MATCHED

INPUT:
Simulation Results when COLOR isn’t MATCHED but SHAPE is MATCHED

INPUT:

![Images showing background image, test image, and object image.]

Advantages

1. Fully automated system.
3. Extended features unlike other security system like image extraction, recognition.
5. Low power consumption as per the design of circuit.

Applications

- Military purposes.
- RADAR system.
- Robotic Vision.
- Security Cameras.
- Video Editing.

4. Future Scope

We can create database for the objects those are to be destroyed. If match between intruding object and one of the objects from database is found, system keep tracking that object to calculate its velocity of motion. This velocity information is needed to decide the angle and time instant at which projectile is to be launched at intruding object to destroy it. After these calculations, position of that intruding object in the form of x-y co-ordinate is found and accordingly cannon will be aimed at that object with help of two servo motors, driven by microcontroller and finally the destruction of the intruder can be done.

5. Conclusion

In this article, an automated surveillance system is described, which includes the following four main building blocks: moving object detection, object tracking and event recognition. In this thesis we presented a set of methods and tools for a “smart” visual surveillance system.
The proposed whole-body object tracking algorithm successfully tracks objects in consecutive frames. Our tests in sample applications show that using nearest neighbour matching scheme gives promising results and no complicated methods are necessary for whole-body tracking of objects. However, due to the nature of the heuristic we use, our occlusion handling algorithm would fail in distinguishing occluding objects if they are of the same size and color. Also, in crowded scenes handling occlusions becomes infeasible with such an approach, thus a pixel-based method, like optical flow is required to identify object segments accurately.

References