RAMKRISHNA MAHATO GOVT. ENGINEERING COLLEGE, PURULIA

A

Project Report

On

Automatic Target Tracking Machine

SUBMITTED BY:

Name: Suvendu Sekhar Roy

Roll No: 35000117025

Dept: CSE (3rd Year)

Index

Si No.	Topics	Page No.
1	Abstract	3
2	Introduction	3
3	What is Automatic Target Tracking	3
4	Algorithm	4
5	Procedure	4
6	Basic Requirements	4
7	Hardware and Software Components	5
7.1	Hardware components	5
7.2	Software components	5
8	Flow Chart	5
9	Target Estimation Generation	6
9.1	Camera Motion Compensation	6
9.2	Potential Target Detection	7
9.3	Tracking and Target Validation	8
10	Conclusion	8
11	References	8

Abstract:

In this study, a real-time fully automatic target detection and tracking method is introduced which is capable of handling variable number of targets. The procedure starts with multiple scale target hypothesis generation in which the distinctive targets are revealed. To measure distinctiveness; first, the interested blobs are detected based on Canny edge detection. Then, the irrelevant blobs are eliminated by two metrics, namely effective saliency and compactness. To handle the missing and noisy observations, the temporary consistency of each target estimate is assessed and external observations are removed. To merge data from multiple scales, a target likelihood map is generated by using kernel density estimation in which weights of the observations are determined by temporal consistency and scale factor. Finally, significant targets are selected by an adaptive thresholding scheme; then the tracking is achieved by minimizing spatial distance between the selected targets in consecutive frames.

Introduction:

Tracking moving targets in complex scenes using an active video camera is a challenging task. Tracking accuracy and efficiency are two keys, yet generally incompatible aspects of a Target Tracking System (TTS). At first, the video image is acquired in real time by camera on the screen. When the target information is known, tracking algorithm is adopted to track the target. Then, the sensor is used to rotate with the target, when the target is in the center of the image, the laser ranging module is opened to obtain the distance between the machine and the target. Finally, through the target location algorithm we calculate the coordinates of the target. The results show that the system is stable for real-time tracking of targets and positioning.
Multiple target detection and tracking has significant importance for many applications, including reconnaissance and surveillance in which the major goal is to reveal trajectories of the targets throughout the scenario. Considering the recent developments, many electro-optical systems are in need of full automation for achieving this task. Therefore, many multitracking algorithms include two fundamental stages as the automatic, time independent detection of targets; and association of the detections in the temporal space.

What is Automatic Target Tracking?

Automatic target tracking is based on radar, photo-optical, infrared, laser, acoustic, magnetic and other principles for the sending and receiving of signals which are processed by computers according to a specified program. Automatic target tracking is preceded by detection of and locking on to a target. During automatic target tracking the present coordinates and lead angles are computed. The ultimate task of automatic target tracking is reliability in damaging the target by increasing the probability of a hit (using percussion projectiles) or the probability of catching the target in a vulnerable area (using proximity projectiles).

Fig: Sample output of automatic target tracking

Algorithm:

The definition sounds straight forward but in computer vision and machine learning, tracking is a very broad term that encompasses conceptually similar but technically different ideas. Some different but related ideas are generally studied under **object tracking.**

1. **Dense optical flow**: These algorithms help estimate the motion vector of every pixel in a video frame.
2. **Sparse optical flow**: These algorithms, like the Kanade-Lucas-Tomashi (KLT) feature tracker, track the location of a few feature points in an image.
3. **Kalman filtering**: A very popular signal processing algorithm used to predict the location of a moving object based on prior motion information. One of the early applications of this algorithm was missile guidance.
4. **Meanshift and Camshift**: These are algorithms for locating the maxima of a density function. They are also used for tracking.
5. **Single object trackers**: in this class of trackers, the first frame is marked using a rectangle to indicate the location of the object we want to track. The object is then tracked in subsequent frames using the tracking algorithm. In most real-life applications, these trackers are used in conjunction with an object detector.
6. **Multiple object track finding algorithms**: in cases when we have a fast object detector, it makes sense to detect multiple objects in each frame and then run a track finding algorithm that identifies which rectangle in one frame corresponds to a rectangle in the next frame.

Procedure:

1. At first, the video image is captured in real time by camera on the screen.
2. When the target information is known, tracking algorithm is adopted to track the target.
3. Then the sensor is used to rotate with the target, when the target is in the center of the image, the laser ranging module is opened to obtain the distance between the machine and the target.
4. Finally, through the target location algorithm we calculate the coordinates of the target.
5. The ultimate task of automatic target tracking is reliability in damaging the target by increasing the probability of a hit.
6. The results show that the system is stable for real-time tracking of targets and positioning.

Basic Requirements:

The desired Quality Requirements for ATTM is as under:

1. Provision for image stitching and image enhancement.
2. Video of the targets to be acquired to ensure 360^o field of view.
3. Detect and track multiple targets.
4. Provide user interface to prioritise the targets.
5. Provision for aligning the selected target to Gunner's Main Sight (GMS).
6. Gunner to have an automatic target tracker (ATT) to track the handed over target or an independent target he has acquired.
7. Target tracking system should able to characterize targets with respect to sensor resolution and distances.

Hardware and Software Components:

i. **Hardware components:**
   1. Raspberry pi
   2. Webcam
   3. Laser
   4. Stepper motor
   5. Jumper wires
   6. Acrylic sheet
   7. Battery

ii. **Software Components:**
   1. Python 3.7
   2. OpenCV python library
   3. Tensor Flow library
   4. Raspberry pi 4

Flow Chart:

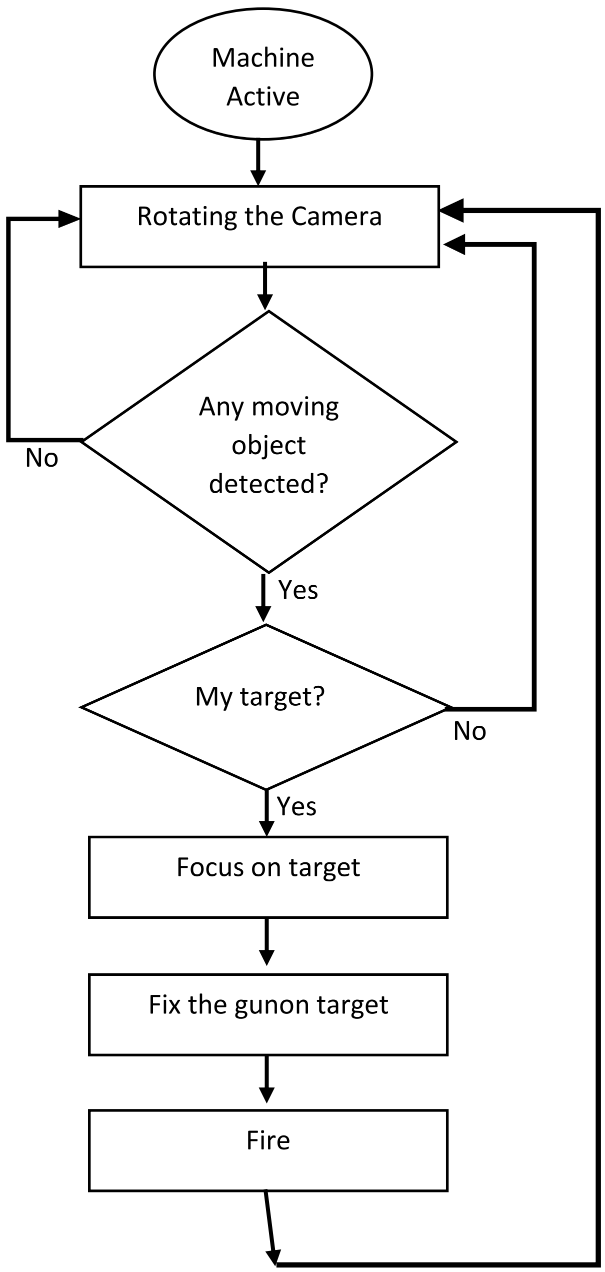

Fig: Flow Chart of ATT system

Target Estimation Generation:

To achieve automatic target detection, each target candidate fulfilling some preliminary requirements, which should be further analysed to decide whether it is a relevant target or not. The target candidates are referred as target hypotheses and generated at each scale of the observation pyramid, obtained by down-sampling the original frame, separately. Therefore, for both hypotheses generation and selection, some assumptions are made to describe the target model.

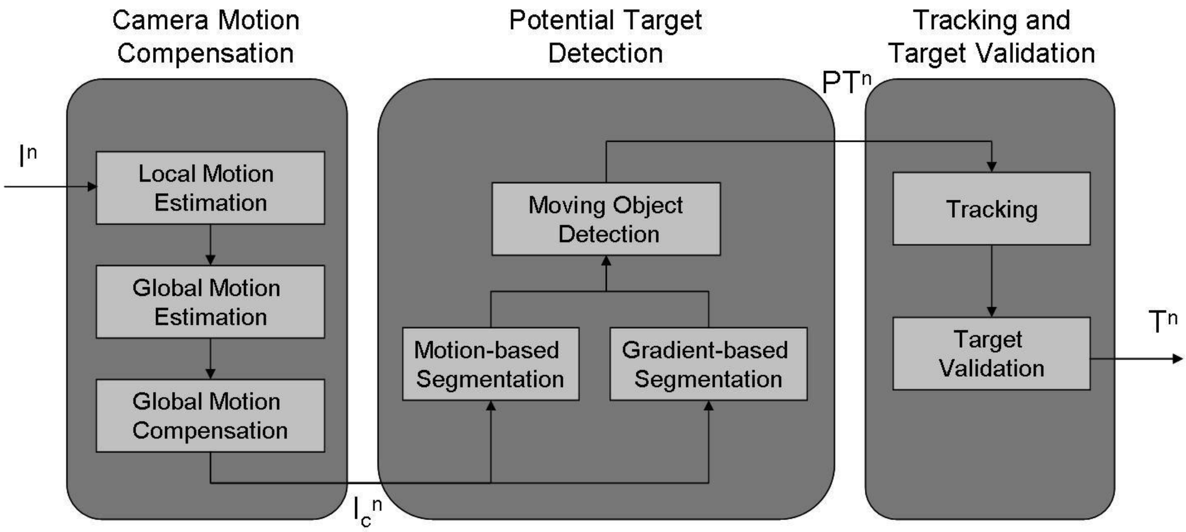

Fig: System Block Diagram

A. Camera Motion Compensation:

Fig: Original image captured by camera

The first assumption is the distinctiveness assumption stating that target candidates should be distinctive from their surroundings. Actually, this assumption is made based on human visual attentional system in which robust saliency detection mechanisms provide focus of attention to the salient regions pre-attentively for further processing. Again similar to human visual system, the distinctiveness is measured by the intensity difference. Most of the saliency detection methods are founded on the same principle; however saliency detection in global scale (by considering the whole scene) would generally require high processing time which may not be suitable for real-time applications. Since the computational complexity is one of the key issues, target hypothesis generation procedure starts with edge detection which is a simple way of detecting contrast between neighbouring pixels. For edge detection, Canny edge detector is preferred for both its low computational complexity and capability of generating closed contours by merging weak edges with the strong edges around their vicinity.

B. Potential Target Detection:

After employing Canny edge detection, morphological closure (to increase probability of generation of closed contours) and filling operations are performed on edge map to obtain the possible target blobs. The importance of filling operation becomes more prominent when a possible target has a layered structure, having nested closed contours inside the target as in Figure, in which an inner loop is detected due to the reflection of the daylight. In such a scenario, detection of the complete vehicle is more preferable than detection of the spot as a separate target; and filling the closed contours inherently yields the selection of the outer most closed contour since both the inside of the spot and the vehicle are filled. After morphological operations, centroids of the resulting filled blobs are obtained by using connected component analysis. Usage of static thresholding in Canny edge detection can be problematic since different scenes may have different contrast spans. Therefore, while a static threshold can satisfactorily detect targets in scenes having high contrast, it may fail to disclose any edges in scenes having low contrast in which targets are still visible to the human visual system. Since the aim is to detect relatively high intensity differences, a dynamic thresholding scheme is applied in which Canny thresholds are adjusted dynamically with a feedback loop, Fig. 4, whose input is the target hypothesis count from the previous frame.

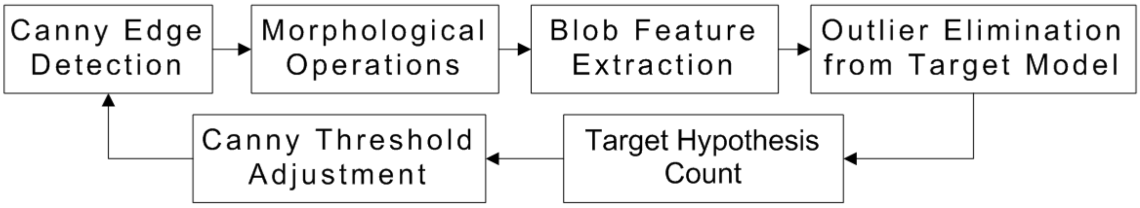

Fig: Flowchart for target hypotheses generation

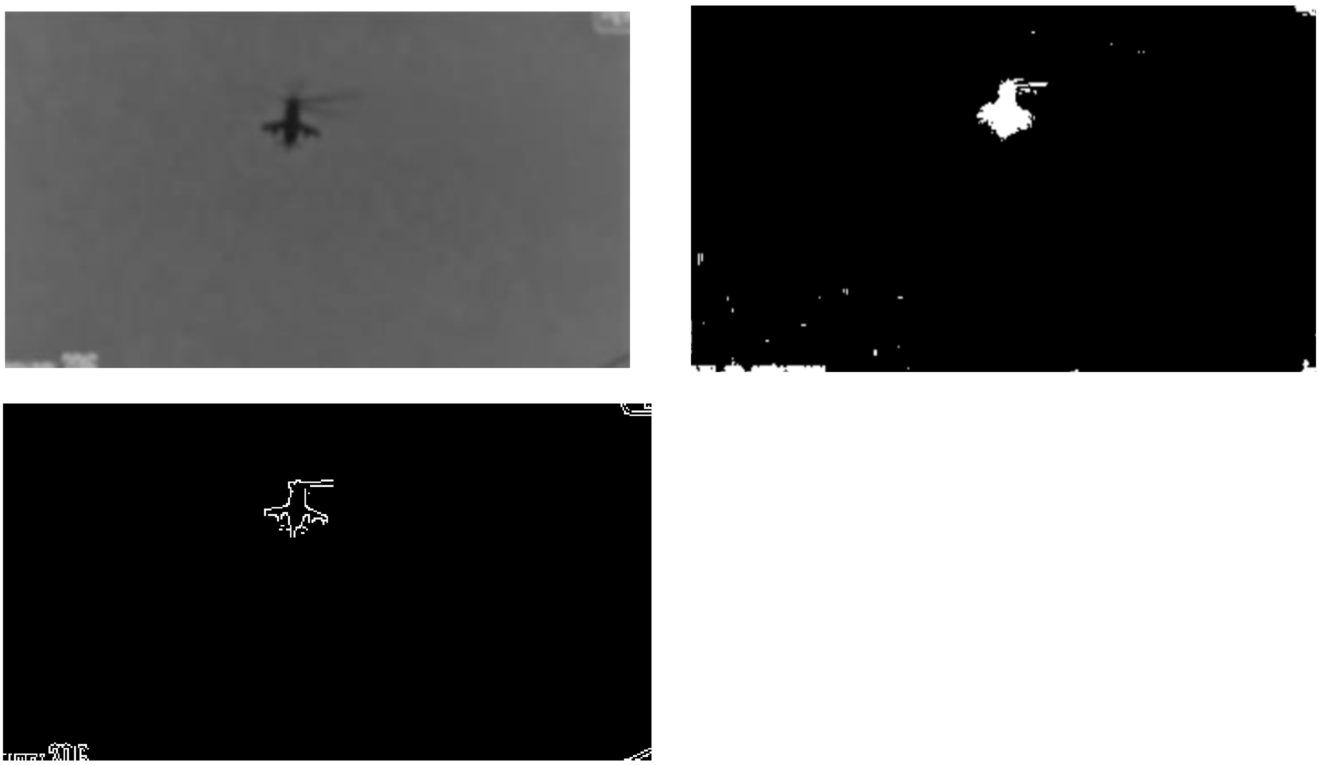

Fig: Effect of filling on a target with layered contours

C. Tracking and Target Validation:

Potential targets are tracked throughout the sequence using a template correlation technique and a Kalman filter to obtain their trajectories, which are analysed by means of curve fitting technique to determine whether they are congruent with an aerial target.

Target Tracking:

Potential moving targets are used to carry out the tracking between consecutive images through a template correlation technique. This technique creates an intensity model for each potential target in PTn−1 to search its best correspondence within In c. This correspondence represents the temporal evolution of that potential target between consecutive frames.

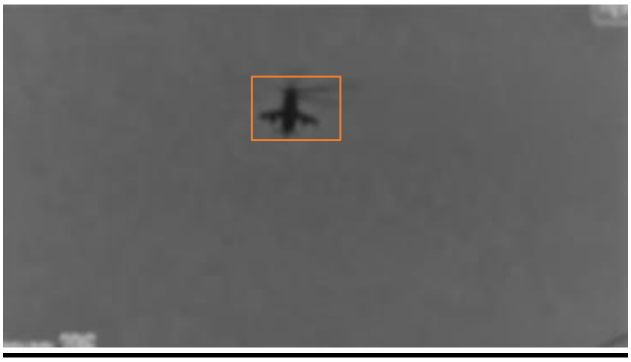

Fig: Target tracking highlighted with bounding box

Target Validation:

Each potential target trajectory is analysed to test if it is consistent with a real target trajectory, in order to validate it, discarding those potential targets associated to erratic or non linear-wise trajectories produced by inherent drawbacks of target images and any other object (i.e. birds).

Conclusion:

In this study, automatic target detection and tracking method designed for real-time systems is introduced. The experiments showed that the proposed algorithm achieves a sufficient true positive rate with a relatively low false discovery rate on the utilized test sets.

Moreover, it is also seen that, usage of a successful detection scheme reduces the complexity of tracker; and even with the simplest association scheme, a sufficient tracking performance can be obtained.

The system estimates local motion through a very fast selective-search block-matching algorithm. Global motion is robustly inferred from local motion using a restricted-affine motion model, and a combination of gradient and motion segmentation techniques are used to detect potential aerial targets from each compensated image. Potential aerial targets are tracked by means of a target template-based correlation technique.

References:

1. Programming Computer Vision with Python by Jan Erik Solem
2. target tracking system: Topics by Science.gov
3. Object Tracking using OpenCV (Python) by Boris Babenko https://www.learnopencv.com/object-tracking-using-opencv-cpp-python/
4. www.pyimagesearch.com
5. Multi Target Tracking System (MTTS) by Tunc¸ Alkanat, Emre Tunali and Sinan O¨ z
6. Automatic aerial target detection and tracking system in airborne FLIR images
7. based on efficient target trajectory filtering in ResearchGate