Implementation of Real Time Object Detection &Tracking System with Rotating Camera: A Survey

Ravi D. Simaria[1] Prof. D.S.Pipalia[2]
[1]Student [2]Assistant Professor
[1,2]Department of Embedded System Design Engineering
[1,2]School of Engineering, R.K. University, Rajkot, India

Abstract— The task of real time detection and tracking of a moving object in a video stream is quite challenging if camera itself is moving. This paper presents a survey of real time detection and tracking of a known/unknown object in video stream with 360° (azimuth) rotating camera. It also presents a comparative study of various image tracking algorithms. The system described in this paper contains a camera that is connected to an embedded system which is capable of processing an input video stream data. The Embedded system is also having an image processing algorithm which detects an object first and then tracks it as long as it is in the line of sight of the camera. As the object moves, the embedded system rotates the camera which is mounted on a stepper motor according to the path of the moving object. The embedded system also sends data to an end user via a Wi-Fi server so that the end user can monitor the activity of the object of interest. The object can be known (already saved in system memory) or can be unknown (can be defined directly by the end user by selecting a portion of the frame in video stream). The embedded system also saves the video stream in a storage device for playback purpose.

Key words: Embedded system, embedded LinuxOS, object detection, object tracking, rotating camera, wireless communication

I. INTRODUCTION

Moving object detection is a necessity for any surveillance system. A static camera can detect and track an object as long as the object is inside the frame of the camera. But as the object [3] goes beyond the boundary of camera frame, the camera stops tracking it, which is a major limiting factor for use of a static camera. This limitation can be overcome by using a rotating camera. This paper describes a system in which a camera rotates according to the path of a moving object and follows it. Along with tracking of an object, the recording of a video stream is also necessary for playback purposes. The system will store all the data into a storage device. The storage device can be a hard disk or a flash drive. It is the responsibility of the system to delete previously stored data and start storing of fresh data from the current frame in case storage device becomes full. The system also sends data to a remote computer over a Wi-Fi network so that an end user can monitor the activity of the object. The end user can also choose the object of interest by selecting a portion of the frame on a monitor screen. So it can be a very intelligent video surveillance system using an embedded system. An embedded system with an embedded OS which provides a platform that supports various languages can make implementation of image processing algorithm little bit easy. One example of such language is the Python which is supported by embedded Linux and python is both, simple and powerful which makes algorithm implementation easier.

The aim [1] of an object detection system is to estimate an object location in the image sequence when initial position of the object is given in the first frame. Additionally based on the application it can also calculate object based information such as orientation of the object, area of the object and/or shape of the object. Alike, the aim of an object tracker is to follow that detected object in every video frame. This work is quite challenging. If the object goes beyond video frame then camera should move in that direction and it should follow the object until it gets disappeared from the camera's sight. But the interesting thing is that, if the object comes again in the sight of the camera then it should be detected and tracked so object occlusion should be handled properly. Because of advancement in technology and availability of low cost cameras in market, realization of this kind of system at low budget is possible.

A. Embedded System:

An embedded system [6] is a combination of hardware and software. It is extensively used in many devices in today's scenario. It is a computer system with a dedicated function within a larger mechanical and electrical system, often with real time computing constraints. The fact that it is application specific makes it different from general purpose system.

For my application, I need an embedded system which is capable to process real time data e.g., video and audio. Many processors that are available in market support this requirement. Various DSPs and ARM processor are such processors. The fact that ARM processor supports various embedded OS which in turn support different languages like C, C++, Python etc., makes it is easy to develop an algorithm with it. Some single board computers [10] with ARM processor and USB ports which are suitable to our needs are available. The camera and the storage device can directly be connected to USB port which removes any requirement of interfacing work.

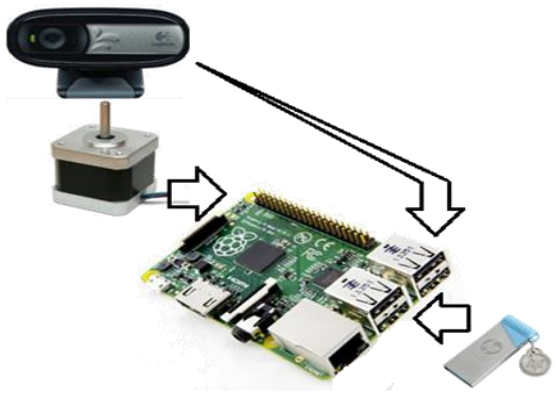

Fig. 1: Interface of camera on a stepper motor and storage device with embedded system.

B. Embedded Linux OS:

Operating systems [7] based on Linux kernels are currently used in embedded system devices such as consumer electronics, networking equipment, machine control etc.

There are many versions of embedded Linux available in the market. Some of them are:

- Android
- Maemo
- Debian
- Raspbian
- Mobilinux

C. Object Detection:

Objection detection [4] is a method to identify an object in a video or a picture frame. The object can be anything that is of interest for further analysis. Numbers of methods have been developed to detect a moving object using an active camera. Usually a moving object detected by static camera using background subtraction.

An object [1] can be represented by its shape, position and appearances. Some of the commonly used object representations are:

- Points
- Articulated shape models
- Skeletal models
- Probability densities
- Templates
- Active appearance models
- Multi view models

The choice of unique feature is very crucial in object detection. Most necessary property of feature is in its uniqueness so that the object can be easily distinguished from its feature space. Different kinds of features that are used to represent an object are:

- Gradient features
- Color features
- Texture features
- Spatio-temporal features
- Multiple feature fusion

The relationship between the object representation and tracking algorithm is very strong. So the object representation should be chosen according to application.

D. Object Tracking:

The basic aim of object tracking is to estimate object location in image sequences. To track an object, many kind of calculations are necessary e.g. object orientation, object shape, direction of object, speed of object, area of object and etc. Major challenges [1] involved in object tracking are:

- Illumination change
- Pose variation
- Object deformation
- Scale variation
- Motion blur
- Noise in image/video sequence
- Partial/Full occlusion of object

Object tracking with moving a camera becomes challenging because many difficulties arise due to these challenges.

E. Rotating Camera:

In my application, I want to track the object as long as it is in the line of sight of the camera. A simple way to do this is to use a 360° rotating camera. The camera is mounted on a stepper motor so that it can rotate by 360° (clockwise and counter-clockwise in azimuth).The stepper motor is basically used to get an accurate position. Whenever the object goes out of boundary of the camera, the motor will rotate to keep the object in the frame. But selecting a suitable stepper motor [11] also very important as many factors such as revolution per minute, step size, voltage specification etc. are involved. For example, for my application, step size of the motor should be small so that the camera can cover larger area. The camera [9] is connected to the system through a USB port and it doesn't need external power supply but the motor is attached with the output lines of the system and it will need an external power supply.

F. Wireless Communication:

The main task of the embedded system is to track the moving object, but parallel to that it should also send tracking data to admin or an end user for monitoring. This can be done by simple wireless communication. To do this, the embedded system should support wireless communication protocol. The receiver can be a laptop or a mobile phone with Wi-Fi facility. The end user has another important role to play i.e. selection of a dynamic object. He can define an object of interest just by selecting a portion of current frame on the monitor screen. After that he can monitor the tracking of that object on his device.

II. LITERATURE SURVEY

As per my survey there are many methods existing for object detection and tracking. First I started with simple object detection methods. Some of the algorithms [12] developed of object detection are:

- Camshift
- Background subtractor
- Meanshift
- Kalman filter
- Update motion history
- calopticalflowSF
- update motion history

Then I realized that lots of methods are also developed for image detection with tracking. The methods [1] for object tracking can be classified as:

A. Generative Tracking Methods:

As per these methods, learn object model to represent the appearance of object. Simple example of this type of method is Eigen tracking.

B. Discriminative Tracking Methods:

These methods lead to find decision boundary that can best separate the object from background and adaptive ensemble of classifier for visual tracking.

C. Hybrid Tracking Methods:

These methods combine both generative and discriminative methods.

Now as per my requirements of application, I need to use hybrid method because of many challenges present in the algorithm implementation. The main challenges involved are:

- To detect the moving object in real time with moving camera, so I have to deal with both, object motion and camera movement in real time.
- How to increase processing frames per second in video stream that is taken from camera.
- How to make a low cost and highly efficient system.
- Which method should be used to achieve the goals mentioned above?

There are some facts about different algorithms as per application requirements:

1) Median Flow Tracker:

Works well until object disappears from screen or lost. No redetection and frame rate is also high.

2) Mean Shift:

Mean Shift works well, but fails for some sequences.The frame rate is also higher.

3) Salient Motion:

In this method object detection is done using temporal differencing and Optical Flow. It does not provide compensation for a moving camera.

4) Open TLD Aka Predator Algorithm [7]:

Open TLD means Tracking Learning and Detection. Previous methods use tracking by detection. Predicts objects location by detector and adapts object parameter at same time. So fails in case of occlusions.TLD has different detector and tracker. Detector is trained with example found on the trajectory a tracker that itself does not depend on the object detector. It is very efficient and an award winning method.

5) CMT [8]:

CMT stands for Consensus-based Matching and Tracking of Keypoints for Object Tracking. It is very accurate algorithm. It is very faster and also an award winning algorithm.

So both, openTLD and CMT are that algorithms which perfectly fit for the application. Out of these two, I have chosen CMT, mainly because of: first, it is latest algorithm and secondly I may add some features of my own to it.

III. METHODOLOGY

A. Embedded System:

I have chosen an embedded system with an ARM processor because the ARM processor is capable to run an OS on it and is also capable to handle and process incoming video stream. The camera can be connected to the system through USB port of the ARM processor based board. Selection of camera involves many parameters such as resolution, voltage-current specification etc. The camera gives continuous video stream to ARM processor which is loaded with an embedded Linux OS. The video stream is processed using CMT object tracking algorithm. This algorithm detects the object into the video stream and follows it, when object goes out of the camera frame, processor gives command to stepper motor to move camera in the direction of the object. Processor would have to consider speed of object as well. Processed data will be stored to storage device which is connected to USB port of the board. In Fig. 2 a flash drive is attached with the board but storage device can be a hard disk as well. When this storage device becomes full, processor will delete all pre-recorded data and start saving fresh incoming processed data from the beginning. The object can be known from the beginning by loading it from memory and algorithm will have static input that should be detected in whole time or in another approach, input can be dynamically given by the end user who is watching processed data from monitor just by selecting a portion of the frame. The final video stream can be viewed directly on a monitor (attached with board via HDMI cable or indirectly via a Wi-Fi server.).

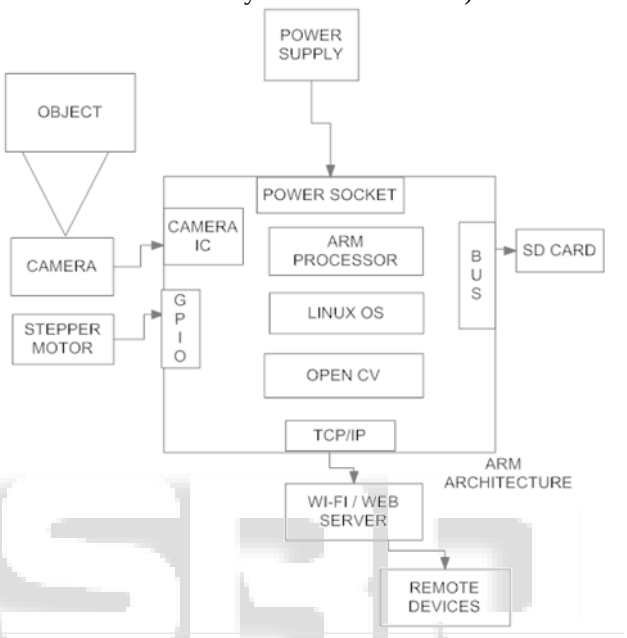

Fig. 2: basic block diagram for image object tracking system

B. CMT Algorithm:

Consensus-based [2] Matching and Tracking of Key points (CMT) is an award-winning object tracking algorithm, initially published at the Winter Conference on Applications of Computer Vision 2014, where it received the Best Paper Award. CMT is able to track a wide variety of object classes in a multitude of scenes without the need of adapting the algorithm to the concrete scenario in any way. CMT is able to achieve excellent results on a dataset that is as large as 60 sequences. It is open source algorithm under the BSD license.

The main idea behind CMT is to break down the object of interest into tiny parts, known as key points. In each frame, it tries to again find the key points that were already there in the initial selection of the object of interest. This is done by employing two different kinds of methods. First, it *tracks* keypoints from the previous frame to the current frame by estimating what is known as its *optic flow*. Second, it *matches* keypoints globally by comparing their *descriptors*. As both of these methods are error-prone, it employs a novel way of looking for consensus within the found keypoints by letting each keypoints vote for the object center. Based on the remaining keypoints, the new bounding box is computed and the process continues.

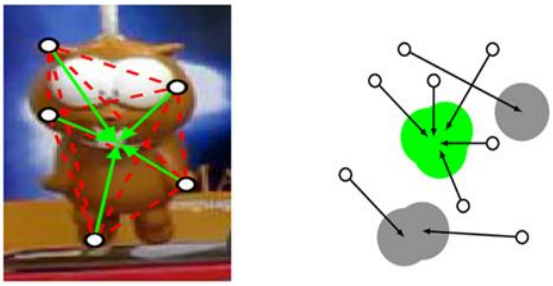

Fig. 3: CMT keypoints detection and clustering

CMT algorithm is capable of taking dynamic input by screen. This is very good advantage of this algorithm to use in my application.

C. Camera Movement:

Moving the camera in the direction of moving object is very crucial part of the application because camera movement should match the path and velocity of the moving object. For camera movement an algorithm for rotation has to be developed. The system will take tracking data from CMT algorithm and based on object and camera position, system will move camera. There are many different methods available for the camera movement. But two different general methods came into my mind: first when the object is going to out of the boundary of the video frame, the camera will be rotated by giving a pulse to the motor according to the object direction and velocity. So when every time the object gets near to the frame boundary, camera will rotate accordingly. In the second method, the object will always be in the center of the frame. Whenever the object moves, camera will also move. The system will always try to keepthe object position at the center of the frame. If the object tries to move from center of the frame, the system will give pulse to the motor and it will move camera. I will evaluate both methods and use the method which will give best performance for my application.

D. Applications:

There are many applications in which this system is useful. Some of them are:

- Motion-based recognition, recognition of an object and/or it motion, based on motion in a series of images e.g. human identification based on the gait.
- Automated surveillance, scene monitoring to detect suspicious activities or unlikely event.
- Human-computer interaction, recognition of gesture, tracking the eye gaze for data input to computers, etc.
- Road traffic monitoring, real time gathering of traffic statistics to direct traffic.
- Follow the object into sports, research studies etc.
- Military applications.
- Industrial application like object counting, Textile web inspection
- Medical applications.

IV. CONCLUSION AND FUTURE WORK

In this paper I described my application and a survey to choose a proper method for detecting and tracking a known/unknown moving object in an incoming video stream. After studying so many methods I finally found that CMT is fit for my application. I also proposed two methods for rotation of the camera according to data given by object detection algorithm. I will use CMT algorithm and one of the two methods for rotation of the camera in my application.

This implementation can be expanded for multiple object tracking. With static camera multiple objects can be detected and tracked as long as they are in the line of sight of the camera. But with a moving camera multiple objects can be detected and any one object can be tracked. Multiple cameras can be used to detect and track multiple objects. The cameras can be synchronized to detect and track different objects.

REFERENCES

[1] Balachander chokkalingam "Evaluation of TLD (aka Predator) algorithm used for real-time tracking of unknown objects in a video stream from eye tracking perspective" master's thesis in Systems, Control and Robotics at Kungliga Tekniska Högskolan (KTH), Sweden, Dec 2012.

[2] Georg Nebehay and Roman Pflugfelder "Consensus-based Matching and Tracking of Key points for Object Tracking" Applications of Computer Vision (WACV), 2014 IEEE Winter Conference, pp. 862-869, March 2014.

[3] P. J. Burt, J. R. Bergen. R. Hingorani, R. Kolczynski, Y.A. Lee, A. Leung, J. Liibin, H. Slivaytser "OBJECT TRACKING WITH A MOVING CAMERA: An Application of Dynaiiiic Motion Analysis" visual motion workshop, pp 2-12, March 1989.

[4] Kye Kyung Kim, Soo Hyun Cho, Hae Jin Kim and Jae Yeon Lee "Detecting and Tracking Moving Object Using an Active Camera" The 7th international conference on advance communication technology ICACT, pp. 817-820, 2005.

[5] "Embedded system basic discussion" internet: http://www.engineersgarage.com/articles/embedded-systems [12 Dec. 2014].

[6] "Embedded linux OS" internet: http://en.wikipedia.org/wiki/Linux_on_embedded_systems [12 Dec 2014].

[7] "OpenTLD algorithm for image tracking" internet: http://www.gnebehay.com/tld/ [15 Dec. 2014].

[8] "CMT algorithm for image tracking" internet: http://www.gnebehay.com/cmt/ [15 Dec. 2014].

[9] "Embedded system camera support" internet: http://elinux.org/RPi_USB_Webcams [20 Dec. 2014].

[10] "Comparison of embedded system boards" internet: https://learn.adafruit.com/embedded-linux-board-comparison/overview [20 Dec. 2014].

[11] "Stepper motor general information" internet: http://www.engineersgarage.com/articles/stepper-motors [20 Dec. 2014].

[12] "Various object tracking algorithms" Internet: http://docs.opencv.org/modules/video/doc/motion_analysis_and_object_tracking.html [15 Dec. 2014]

