Asian Journal of Convergence in Technology *Volume III, Issue I*
Issn No.:2350-1146, I.F-2.71

Video surveillance using raspberry Pi with particle filtering

Bhor Dipali Ambadas
E&TC dept. Sharadchandra Pawar collage of engineering
Otur (Pune), India.
Email: dipalibhor_28@yahoo.com

G.U.Kharat
E&TC. Sharadchandra pawar collage of engineering
Otur (Pune), India.
Email:gukharat@gmail.com

Abstract—One of the fundamental problem in vision is that of tracking objects through sequences of images. Within this report the design of Particle filter algorithm to track the target and show the resulting improvement in tracking. More specifically, this project described the technique of how to track Moving Object. The proposed architecture is suitable for indoors and outdoors scenes with static background and overcomes the problem of stationary targets fading into the background. Optimal estimation problems for non-linear non-Gaussian state-space models do not typically admit analytic solutions. Since their introduction in 1993, particle filtering methods have become a very popular class of algorithms to solve these estimation problems numerically in an online manner, i.e. recursively as observations become available, and are now routinely used in fields as diverse as computer. The real time video surveillance using a system equipped for sensor support data is being developed to provide situational in the target of interest & also provide a safe means of video surveillance in the target place using raspberry pi.

Keywords—***video surveillance***, **opencv**, ***raspberry pi, camera module, particle filter, object detection***

I. INTRODUCTION

For this process of video Surveillance OPENCV, an open source computer vision library is used. For using opencv program, which can be integrated only with the micro controller kit known as raspberry pi which uses ARM 11 board the core frequency is set to 700MHz. It has the ability to boot from the SD card. It uses GPIO where its header consists of 17 pins. These pins could be connected to simple LED drivers, relay modules through driver transistors, or even stepper motor or servo motor. The use of microcontroller board, raspberry pi is only possible only when the provided operating system is Linux either Ubuntu or some Linux based operating system where ubuntu13.3 raring ringtail is considerably suitable Opencv is a set of libraries targeted for real time computer vision applications such as human computer interaction, object identification, segmentation and recognition, face recognition, gesture recognition, Motion for this application. **CMake**, the cross-platform, open-source build system. CMake is a family of tools designed to build, test and package software. CMake is used to control the software compilation process using simple platform and compiler independent configuration files. It also serves as a graphical user interface for compilation of opencv programs.

II. EASE OF USE

Like Kalman filters, There is a great way to track the state of a dynamic system for which you have a Bayesian model. That means if a model that how the system changes in time, possibly in response to inputs, and a model of what observation should see in particular states, by using particle filters to track required belief state. So why should you use particle filters instead of Kalman filters? Well, the main reason is that for a lot of large or high-dimensional problems, particle filters are tractable whereas Kalman filters are not. The key idea is that a lot of methods, like Kalman filters, try to make problems more tractable by using a simplified version of full & complex model. Then the model can find an exact solution using that simplified model. But sometimes that exact solution is still computationally expensive to calculate, and sometimes a simplified model just isn't good enough. So then the model need something like particle filters, which let by using the full, complex model, but just find an approximate solution instead.

OPENCV supports several languages including c, visual c ++, python etc. in our paper we use python, is an agile programming language. Python is an open source object-oriented programming language that offers two to ten folds programmer productivity increases over languages like C, C++, Java, C#, Visual Basic (VB), and Perl. Opencv, which is user friendly and most of the programs are available in internet as an open source.

III. IMPLENENTATION METHOD.

This project is implementing a system with particle filtering method with the major component like Raspberry Pi, Object detection & Tracking. The real time video and all other sensor data's are displayed in the webpage which can be viewed from anywhere in the world by setting a specific internet address.

A. Architecture

Fig: 1 Architecture model

The system consists of voltage sensor, camera, PC, Raspberry Pi etc. All the sensor data's are controlled using raspberry pi. Those data's are sent to the webpage using Wi-Fi interface and the real time video is also sent along with sensor related data. The video is taken only when the particular template matching is verified. For template matching and real time video we use OPENCV. All Process is work out by sequential manner which show in Fig: 1 Architecture model.

B.Raspberry Pi

The Raspberry Pi is low cost ARM based palm-size computer. The Raspberry Pi has microprocessor ARM1176JZF-S which is a member of ARM11 family and has ARMv6 architecture. It is built around a BCM2835 broadcom processor. ARM processor operates at 700 MHz & it has 512 MB RAM. It consumes 5V electricity at 1A current due to which power consumption of raspberry pi is less. It has many peripherals such as USB port, 10/100 Ethernet, GPIO, HDMI & composite video outputs and SD card slot.SD card slot is used to connect the SD card which consists of raspberry Linux operating system. The out looking of raspberry pi as shown in Fig: 2 Raspberry Pi module.

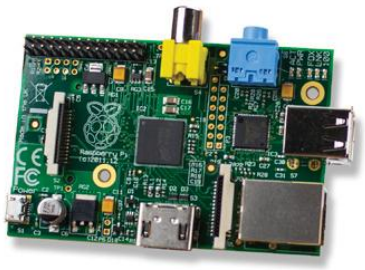

Fig: 2 Raspberry pi (Model B - **756-8308)**

C. Camera module of Raspberry Pi

High definition camera module compatible with the Raspberry Pi model A and model B. Provides high sensitivity, low crosstalk and low noise image capture in an ultra-small and lightweight design. The camera module connects to the Raspberry Pi board via the CSI connector designed specifically for interfacing to cameras. The CSI bus is capable of extremely high data rates, and it exclusively carries pixel data to the BCM2835 processor.

Specifications

Image Sensor: Omnivision 5647 CMOS image sensor in a fixed-focus module with integral IR filter
Resolution: 5-megapixel
Still picture resolution: 2592 x 1944
Max image transfer rate: 1080p: 30fps (encode and decode)
720p: 60fps
Connection to Raspberry Pi: 15 Pin ribbon cable, to the dedicated 15- pin MIPI Camera Serial Interface (CSI-2)
Image control function:s Automatic exposure control
Automatic white balance
Automatic band filter
Automatic 50/60 Hz luminance detection
Automatic black level calibration
Temp range: Operating: -30° to 70°
Stable image: 0° to 50°
Lens size: 1/4"
Dimensions: 20 x 25 x 10mm
Weight: 3g

Fig: 3 Camera module

This module is including all specification regarding Raspberry Pi module as shown in Fig: 3 camera module.

Updating the firmware on the Raspberry Pi

Updating the firmware on your Raspberry Pi computer will install all of the drivers necessary to run the camera module.

1. Installing the git-core package pi@raspberrypi ~ $ sudo apt-get install git-core –y
2. Download the 'rpi-update' script pi@raspberrypi ~ $ sudo wget http://goo.gl/1BOfJ -O /usr/bin/rpi-update
3. Change the permissions in the 'rpi-update' script so we can run it pi@raspberrypi ~ $ sudo chmod +x /usr/bin/rpi-update

or

4. pi@raspberrypi ~ $ sudo nano /boot.config.txt Execute the 'rpi-update' script to update the firmware on the Raspberry Pi pi@raspberrypi ~ $ sudo rpi-update (This will take several minutes to complete)

Update the boot configuration file

Edit the /boot/config.txt boot file to add in the following two lines at the bottom of the file:

start_file=start_x.elf

fixup_file=fixup_x.dat
Also make sure that the following line is present within the file (This line was added when the GPU memory allocation was set to 128 meg). If it is not then add the following line at the bottom of the file:
gpu_mem=128
You can edit this file on the Raspberry Pi using one of the following commands:
pi@raspberrypi ~ $ sudo vi /boot.config.txt

D. Ethernet

Ethernet is the computer networking technology which allows computers to communicate and share resources over the internet. Ethernet was standardized as IEEE 802.3. it is one of the most widely implemented LAN standard originally developed by Xerox. Different Ethernet networks also connect to a router that provides access to the internet.

E. Particle filter

Particle filter is example of bayes filter. It is also known as sequential monto carlo method. It is use for representing pdf as set of particles. Each particle contains one set of values for the state variables. Find an approximate solution using a complex model (arbitrary pdf) rather than an exact solution using a simplified model (Gaussians). The general state space hidden Markov models, which are summarized in section 2.1, provide an extremely flexible framework for modeling time series. The great descriptive power of these models comes at the expense of intractability: it is impossible to obtain analytic solutions to the inference problems of interest with the exception of a small number of particularly simple cases. The \particle" methods described by this tutorial are a broad and popular class of Monte Carlo algorithms which have been developed over the past fifteen years to provide approximate solutions to these intractable inference problems. Particle filter algorithm is executed into three steps like

i. Resampling
ii. Update
iii. Predict

IV. Object detection

Most of people detection and tracking systems have an initial phase where a background model is created. It is employed to easily detect the moving objects in the environment (foreground). Techniques usually employed for that purpose consist in creating a background image (pixel by pixel) using several images of the scene, if possible, without motion elements in them [16], [19], [20], [21]. Real-time object tracking is the critical task in many computer vision applications such as driver assistance , surveillance , perceptual user interfaces augmented reality , smart rooms , object-based video compression .

Because target can be considered movable elements in the environment, it is very helpful to separate the points that belong to the environment (back-ground) from those that do not (foreground). The background information is important for at least two reasons.

- First, if some of the target features are also present in the background; their relevance for the localization of the target is diminished.
- Second, in many applications it is difficult to exactly delineate the target, and its model might contain background features as well.

A. *Algorithms for Background Modeling*

Background maintenance by median filtering

1. Initialize: Acquire K frames. At each pixel, determine the median intensity of these K. If the image sequence is in colour, this is done for each of the R, G, B streams. The median represents the current background value.

2. Acquire frame K+1. Compute the difference between this frame and the current background at each pixel(a scalar for gray image; a vector for RGB)

3. Threshold this difference to remove/reduce noise. Simple thresholds may be too crude at this point and there is scope for using, e.g., hysteresis

4. Use some combination of blurring and morphological operations to remove very small regions in the difference image, and to fill in ‘holes’ etc. in larger ones. Surviving regions represent the moving objects in the scene.

5. Update the median measurement, incorporating frame K+1 and discarding the current earliest measurement.

6. Return to (2) for the next frame.

Step 5 here is potentially very expensive, demanding the storage of the last K images and a per-pixel sort at each frame. An efficient short cut is available only the current background is stored, and intensities are incremented by one if the current frame is brighter at the pixel, and decremented if it is darker. This trick converges the background onto an intensity for which half the updates are brighter and half darker- that is, the median[23].

V. Overview

The real time video taken by the voltage sensor intimation is viewed from the webpage with the given Ip address. Overview of the project as shown in fig: 5. The figure explains the wired set up for the real time video surveillance. The video captured

by the webcam is viewed in the webpage in the Ip address with port 8081 configuration.

Fig: 4 wired setup of project

VI. Result

The output window shows video of camera obtained with raspberry pi with particle filtering in fig. Particle filtering obtained final filtering output including different stages like create particle, decision, update, resample with the help of Matlab coding. Particle filter method is used obtaining noise free output with video taken at different detected position.

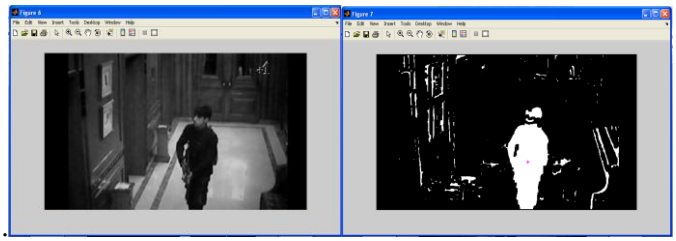

Fig: 5 Raspberry video & filter output.

VII. Conclusion

This method is tested on the images from the CCTV footage and shows good results. The program that is written in this paper intends to provide a basic understanding of the Particle filter. This program is too slow for real time computations, because the handling with symbolic expressions in MATLAB needs a lot of calculation time. But in general the recursive nature of the Particle filter allows for efficient real-time processing.

VIII. Acknowlwdgement

This paper has been conducted under the guidance of our tutor Prof. Dr. G. U. Kharat and Prof. Bodake S.H Also authors would like to thank the department of E&TC, Sharadchandra Pawar College of Engineering, Otur, Pune .

References

[1] Anderson, B. D. O. and Moore, J. B.(1979) *Optimal Filtering*, Prentice Hall, Englewood Clis, New Jersey.
[2] Andrieu, C. and Doucet, A. (2002) Particle filtering for partially observed Gaussian state space models. *Journal of the Royal Statistical Society* B, 64(4), 827-836.
[3] Arulampalam, S., Maskell, S., Gordon, N. and Clapp, T. (2002) A tutorial on particle filters for on-line nonlinear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2):174{188.
[4] Bresler, Y. (1986) Two-filter formula for discrete-time non-linear Bayesian smoothing. *International Journal of Control*,43(2), 629{641.
[5] Briers, M., Doucet, A. and Maskell, S. (2008) Smoothing algorithms for state-space models, Annals of the *Institute of Statistical Mathematics*,to appear.
[6] Briers, M., Doucet, A. and Singh, S.S. (2005) Sequential auxiliary particle belief propagation, *Proceedings Conference Fusion.*
[7] Carpenter, J., Cli_ord, P. and Fearnhead, P. (1999) An improved particle filter for non-linear problems.*IEE proceedings | Radar, Sonar and Navigation*, 146, 2{7.
[8] Capp_e, O. , Godsill, S. J. and Moulines, E. (2007) An overview of existing methods and recent advances in sequential Monte Carlo. *IEEE Proceedings*, 95(5):899{924.
[9] Chopin, N. (2004) Central limit theorem for sequential Monte Carlo and its application to Bayesian inference. Ann. Statist., 32, 2385-2411.
[10] Crisan, D. and Doucet, A. (2002) A survey of convergence results on particle filtering for practitioners. *IEEE Transactions on Signal Processing*,50(3), 736{746.
[11] Del Moral, P. (2004) *Feynman-Kac Formulae: Genealogical and Interacting Particle Systems with Applications. Series: Probability and Applications*, Springer-Verlag, New York.
[12] Del Moral, P., Doucet, A. and Jasra, A. (2008) On adaptive resampling procedures for sequential Monte Carlo methods. Technical report INRIA.
[13] Douc, R., Capp_e, O. and Moulines, E. (2005) Comparison of resampling schemes for particle filtering. In 4th International Symposium on Image and Signal Processing and Analysis (ISPA).
[14] Doucet, A., Godsill, S. J. and Andrieu, C. (2000) On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*,10, 197{208.
[15] Doucet, A., Briers, M. and Senecal, S. (2006) E_cient block sampling strategies for sequential Monte Carlo. *Journal of Computational and Graphical Statistics*,15(3), 693{711.
[16] Doucet, A., de Freitas, N. and Gordon, N.J. (eds.) (2001) *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, New York.
[17] Doucet, A., de Freitas, N. and Gordon, N.J. (2001) An introduction to sequential Monte Carlo methods. In [16], 3{13.
[18] Fearnhead, P. (2008) Computational methods for complex stochastic systems: A review of some alternatives to MCMC. *Statistics and Computing*,18, 151{171.

[19] Fearnhead, P., Wyncoll, D. and Tawn, J. (2008) A sequential smoothing algorithm with linear computational cost. Technical report,Department of Mathematics and Statistics, Lancaster University.
[20] Gilks, W.R. and Berzuini, C. (2001). Following a moving target -Monte Carlo inference for dynamic Bayesian models. *Journal of the Royal Statistical Society* B,63, 127{146.
[21] Godsill, S.J. and Clapp, T. (2001) Improvement strategies for Monte Carlo particle filters, in [16], 139-158.
[22] Gordon N.J., Salmond D.J. and Smith A.F.M. (1993) Novel approach to nonlinear/non-Gaussian Bayesian state estimation.*IEE-Proceedings-F*, 140, 107{113.
[23] Johansen, A.M. and Doucet, A. (2008) A note on auxiliary particle filters. *Statistics and Probability Letters*, 78(12),1498{1504.