



Kavitha.A et al, International Journal of Computer Science and Mobile Applications,

Vol.2 Issue. 1, January- 2014, pg. 63-78

ISSN: 2321-8363

OBJECT TRACKING AND RECOGNITION FOR REAL TIME VIDEOS USING NAIVE – BAYES CLASSIFICATION

KAVITHA.A,

M.E- IInd yr,

Department of Computer Science and Engineering,

Government College Of Engineering , Tirunelveli,

Anna University , Chennai,

E-Mail id : akavithacse@gmail.com

K.THULASIMANI,

Assistant professor,

Department of Computer Science and Engineering,

Government College Of Engineering , Tirunelveli,

Anna University , Chennai,

E-Mail id : thulasikannan999@gmail.com

ABSTRACT

Tracking is to match the correspondences between a frame, and objects which are in motion. It is usually performed in the context of higher-level applications that require the location and / or shape of the object in every frame. Typically, assumptions are made to constrain the tracking problem in the context of a particular application. In this i had categorizes the tracking methods on the basis of the object and motion representations. In all the previous works, tracking is only depends on the low level correspondences between frames, but here I had proposed that a tracking is done based on both low and high level correspondences, and the tracked result is feed back into a recognition part to find category of an object it is the final result. Tracking is done based on the detection of the movable objects, it's done by Gaussian Mixture Model. Harris Corner is used for tracking and the classification is based on Naive-Bayes Classifier. This paper is works well in a challenging areas like drastic view change, background clutter, and morphable objects, crowd, complex scenes.

KEYWORDS:- Video Analysis, Object Tracking, Object Recognition



INTRODUCTION

Object tracking is an important task in 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. There are three key steps in video analysis they are Detection of interesting moving objects, Tracking of such objects from frame to frame, and Analysis of object tracks to recognize their behavior. Therefore, the use of object tracking is pertinent in the following tasks like Motion-Based Recognition, Automated Surveillance, Video Indexing , Human Computer Interaction, Traffic Monitoring, Vehicle Navigation. Depending on the tracking domain, a tracker can also provide object-centric information, such as orientation, area, or shape of an object.

One can simplify the 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 a 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. Numerous approaches for object tracking have been proposed. These primarily differ from each other based on the way they approach the following questions:

1. Which object representation is suitable for tracking?
2. Which image features should be used?

The answers to these questions depend on the context / environment in which the tracking is performed and the end use for which the tracking information is being sought. A large number of tracking methods have been proposed which attempt to answer these questions for a variety of scenarios.

This paper suggests the method of using point representation of an object, by means of this we can track that objects motion easily, also a classifier based on Gaussian mixture models can be

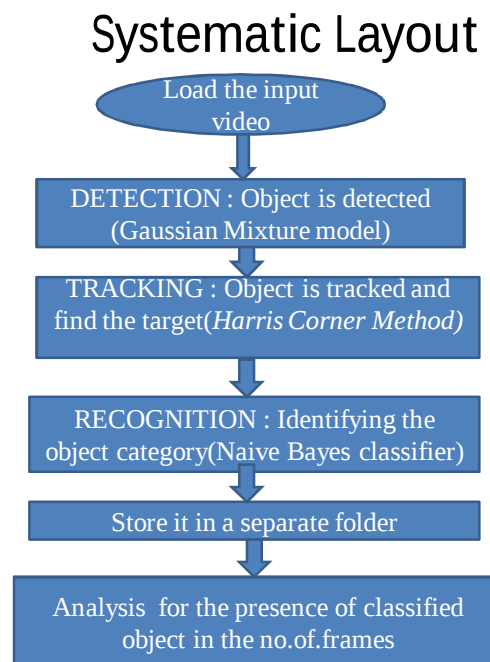
trained discriminatively to improve the accuracy. Naive bayes classifier is fully based on a supervised learning method, posterior probability, maximum likelihood, strong independent assumption between a features. Based on the above features and the class conditional probability the classifier will categorize one object from the others.



Here I had proposed for objects like Human (H), Car (C), Flight (F), Dog (D) and Helicopter (H). if any other objects than the above mentioned are present in a scenario it will categorize as Others (O). Main Contribution of this paper includes it will work well for Multiple Object Tracking (i.e) If two or more above mentioned objects are present in same video it will Classifies well also, Our motivation in studying this problem is to create a visual surveillance system with real-time moving object detection, classification, tracking and activity analysis capabilities.

METHODS

Object of interest is detected by a specified bounding box, but its category is not provided. This target may or may not have a semantic meaning. Therefore, in the first few frames when the tracker does not know the target category, tracking the target only relies on the detection and tracking model. Meanwhile, video-based object recognition is applied on the tracked objects. When the target is recognized properly, the target will be automatically incorporated with their features to provide more information about the target. The framework of this work is :



6

i) DETECTION PROCEDURE

In a tracking scenario, an object can be defined as anything that is region of interest for further analysis. For instance, boats on the sea, fish in an aquarium, vehicles on a road, planes in the air,



people walking on a road, or bubbles in the water are a set of objects that may be important to track in a specific domain. Objects can be represented by their shapes and appearances.

Here I had represent an object by means of point representation. The object is represented by a point that is, the centroid or by a set of points. In general, the point representation is suitable for tracking objects that occupy small regions in an image.

Raw pixel values are widely used in visual object tracking because of their simplicity and efficiency. Raw pixel representation directly utilizes the raw color or intensity values of the image pixels to represent the object regions. After representing an object, its being detected by means of Gaussian mixture model. Some detectors find points of high local symmetry, others find areas of highly varying texture, while others locate corner points. Corner points are interesting as they are formed from two or more edges and edges usually define the boundary between two different objects or parts of the same object.

Distinguishing foreground objects from the stationary background is both a significant and difficult research problem. Almost all of the visual surveillance systems first step is detecting foreground objects. This both creates a focus of attention for higher processing levels such as tracking, classification and behavior understanding and reduces computation time considerably since only pixels belonging to foreground objects need to be dealt with. Short and long term dynamic scene changes such as repetitive motions (e. g. Waiving tree leaves), light reflectance, shadows, camera noise and sudden illumination variations make reliable and fast object detection difficult. Hence, it is important to pay the necessary attention to object detection step to have reliable, robust and fast visual surveillance system.

FOREGROUND DETECTION / BACKGROUND SUBTRACTION

The first step in any moving object tracking system, which distinguishes moving objects from the stationary background, A simple pixel based differencing to detect changes happened in the scene. But, the disadvantage of this approach is that some background objects are often



detected as foreground objects because of the changes in illumination, movement of leaves and presence of walking people. To compensate this problem, a module which dynamically updates the background model to handle the changes in the background scenes. The areas of the image plane where there is a significant difference between the observed and background model images indicate the location of the moving objects or presence of the new object. Usually, a connected component algorithm or blob analysis is applied to obtain connected regions corresponding to the objects. This process is commonly referred to as the background subtraction. It is done by :

vision.ForegroundDetector();

Background subtraction is usually the first step for segmenting out objects of interest in a scene for almost all computer vision applications such as video surveillance systems, traffic monitoring, environment monitoring, obstacle detection, etc. The name “background subtraction” comes from the simple technique of subtracting the observed image from the background image and thresholding the result to find the objects of interest. As a matter of fact, this process is also called “scene change detection” as it detects the changes in the original background scene. We use a combination of a background model and low-level image post-processing methods to create a foreground pixel map and extract object features at every video frame.

GAUSSIAN MIXTURE MODEL

An adaptive online background mixture model that can robustly deal with lighting changes, repetitive motions, clutter, introducing or removing objects from the scene and slowly moving objects. Their motivation was that a unimodal background model could not handle image acquisition noise, light change and multiple surfaces for a particular pixel at the same time. Thus, i used a mixture of Gaussian distributions to represent each pixel in the model. Due to its promising features, we implemented and integrated this model in our visual surveillance system.



In this model, the values of an individual pixel (e. g. Scalars for gray values or vectors for color images) over time is considered as a “pixel process” and the recent history of each pixel, $\{X_1, X_2, \dots, X_t\}$, is modeled by a mixture of K Gaussian distributions. The probability of observing a current pixel value then becomes:

$$P(X_t) = \sum_{i=1}^k w_{i,t} * \phi(X_t, \mu_{i,t}, \Sigma_{i,t})$$

where $w_{i,t}$ is an estimate of the weight of the i th Gaussian ($G_{i,t}$) in the mixture at time t , $\mu_{i,t}$ is the mean value of $G_{i,t}$ and $\Sigma_{i,t}$ is the covariance matrix of $G_{i,t}$ and ϕ is a Gaussian probability density function.

The procedure for detecting foreground pixels is as follows. At the beginning of the system, the K Gaussian distributions for a pixel are initialized with predefined mean, high variance and low prior weight. When a new pixel is observed in the image sequence, to determine its type, its RGB vector is checked against the K Gaussians, until a match is found.

ii) TRACKING PROCEDURE :

After motion detection, surveillance systems generally track moving objects from one frame to another in an image sequence. The tracking algorithms usually have considerable intersected with motion detection during processing. Tracking over time typically involves matching objects in consecutive frames using features such as points, lines or blobs. Useful mathematical tools for tracking include the Kalman filter, the Condensation algorithm, the dynamic Bayesian network, the geodesic method, etc. Tracking methods are divided into four major categories:

- Region-based tracking.
- Active-contour-based tracking.
- Feature based tracking.



- Model-based tracking.
- Point tracking.

In this paper I had implemented Harris Corner Point method for tracking, because some detectors find points of high local symmetry, others find areas of highly varying texture, while others locate corner points. Corner points are interesting as they are formed from two or more edges and edges usually define the boundary between two different objects or parts of the same object.

The corner tracking method is mainly used in the applications like :

- Stereo Matching.
- Image Registration .
- Stitching Of Panoramic Photographs.
- Object Detection / Recognition.
- Motion Tracking.

POINT TRACKING

The detected objects are represented by points, and the tracking of these points is based on the previous object states which can include object positions and motion.

HARRIS CORNER TRACKING :

The Harris corner tracker is based on an underlying assumption that corners are associated with maxima of the local autocorrelation function. It is less sensitive to noise in the image than most other algorithms, because the computations are based entirely on first derivatives. The algorithm has proved popular due to its high reliability in finding junctions and its good temporal stability, making it an attractive corner detector for tracking. It should be noted that because these algorithms rely on spatial derivatives, image smoothing is often required to



improve their performance. While improving the detection reliability, it has been shown that smoothing may result in poor localization accuracy.

TRACKING SINGLE OBJECTS :

The most common approach in this category is template *matching*. Template matching is a brute force method of searching the image, for a region similar to the object template, defined in the previous frame. The position of the template in the current image is computed by a similarity measure. A limitation of template matching is its high computation cost due to the brute force search. To reduce the computational cost, researchers usually limit the object search in the vicinity of its previous position.

Tracking Multiple Objects :

Modeling objects individually does not take into account the interaction between multiple objects and between objects and background during the course of tracking. An example interaction between objects can be one object partially or completely occluding the other. The tracking methods given in the following model the complete image, that is, the background and all moving objects are explicitly tracked, also proposed an object tracking method based on modeling the whole image, as a set of layers. This representation includes a single background layer and one layer for each object. Each layer consists of shape priors (ellipse), θ , motion model (translation and rotation), θ , and layer appearance, A , (intensity modeled using a single Gaussian). Layering is performed by first compensating the background motion modeled by projective motion such that the object's motion can be estimated from the compensated image using 2D parametric motion. Then, each pixel's probability of belonging to a layer (object), is computed based on the object's previous motion and shape characteristics. Any pixel far from a



layer is assigned a uniform background probability. Later, the object's appearance (intensity, color) probability is coupled with previous objects motion to obtain the final layer estimate.

iii) CLASSIFICATION PROCEDURE :

Feature extraction holds an important stepping stone to pattern recognition and machine learning problems. To recognize and classify an object in an image, I must first extract some features out of the image. Feature extraction is the technique to extract various image attributes for identifying or interpreting meaningful physical objects from images. The primary idea is to represent the visual appearance of an object by distinctive key features, or attributes. Once the object is segmented, care- fully chosen features are extracted to perform the desired recognition task using this reduced representation instead of the full size object image. It is often decomposed into feature construction and feature selection. Now under different conditions (e.g. lighting, background, changes in orientation etc.) the feature extraction process will find some of these distinctive keys, The objective is to identify the most discriminative image features with the lowest dimensionality in order to reduce the computational complexity and improve the tracking accuracy.

With the help of object type information, more specific and accurate methods can be developed to recognize higher level actions of video objects. Typical video scenes may contain a variety of objects such as people, vehicles, animals, natural phenomenon (e.g. rain , snow), plants and clutter. However, the main target of interest in surveillance applications is generally humans and vehicles. Also, real time nature and operating environments of visual surveillance applications require a classification scheme which is computationally inexpensive, reasonably effective on small targets and invariant to lighting conditions We have satisfied most of these requirements by implementing a classification scheme which is able to categorize detected video objects into pre-defined groups of human, the human group and vehicle by using image-based object features.



TEMPORAL CONSISTENCY :

The performance of the object classification method is dependent on the quality of the output of the object segmentation step. Due to environmental factors, such as objects being occluded by stationary foreground objects (e.g. A fence or a pole in front of the camera) or due to the fact that only a part of the object enters into the scene, the shape of the detected region does not reflect an object's true silhouette. In such cases, the classification algorithm fails to label the type of the Object correctly. For instance, the part of a vehicle entering into the scene may look like a human, or a partially occluded human may look like a human group. Therefore, we use a multi-hypothesis scheme to increase the accuracy of our classification method.

NAIVE BAYES CLASSIFICATION :

A Bayes classifier is a simple probabilistic classifier based on applying Bayes theorem with strong (naive) independence assumptions. A more descriptive term for the underlying probability model would be "independent feature model". In simple terms, a naive Bayes classifier assumes that the presence (or absence) of a particular feature of a class is unrelated to the presence (or absence) of any other feature. Depending on the precise nature of the probability model, naive Bayes classifiers can be trained very efficiently in a supervised learning setting. Class Probability function is given by :

$$P(h/D) = P(D/h) P(h) / P(D).$$

$P(h)$: Prior probability of hypothesis h .

$P(D)$: Prior probability of training data D .

$P(h/D)$: Probability of h given D .

$P(D/h)$: Probability of D given h .

It is based on the Bayesian theorem It is particularly suited when the dimensionality of the inputs is high. Parameter estimation for naive Bayes models uses the method of maximum likelihood, class conditional probability, posterior probability, strong independent assumption between a features In many practical applications, parameter estimation for naive. In other words, one can work with the naive Bayes model without believing in Bayesian probability or using any Bayesian methods. Along with these features of Naive Bayes classifier, I had also used Mean, Height, Width of the each frame for feature extraction to enhance the performance of the classifier. In spite of their naive design and apparently over-simplified assumptions, naive Bayes classifiers have worked quite well in many complex real-world situations. Training and Testing set of this paper is given by :

```
x = X(training(c,1),:);
```

```
% test set
```

```
u=X(test(c,1),:);
```

Where X is the area / frames which containing the result of tracking module.

RESULTS

The result of this paper includes some sample input videos collected from standard datasets like INRIA,PASCAL VOC 2007.

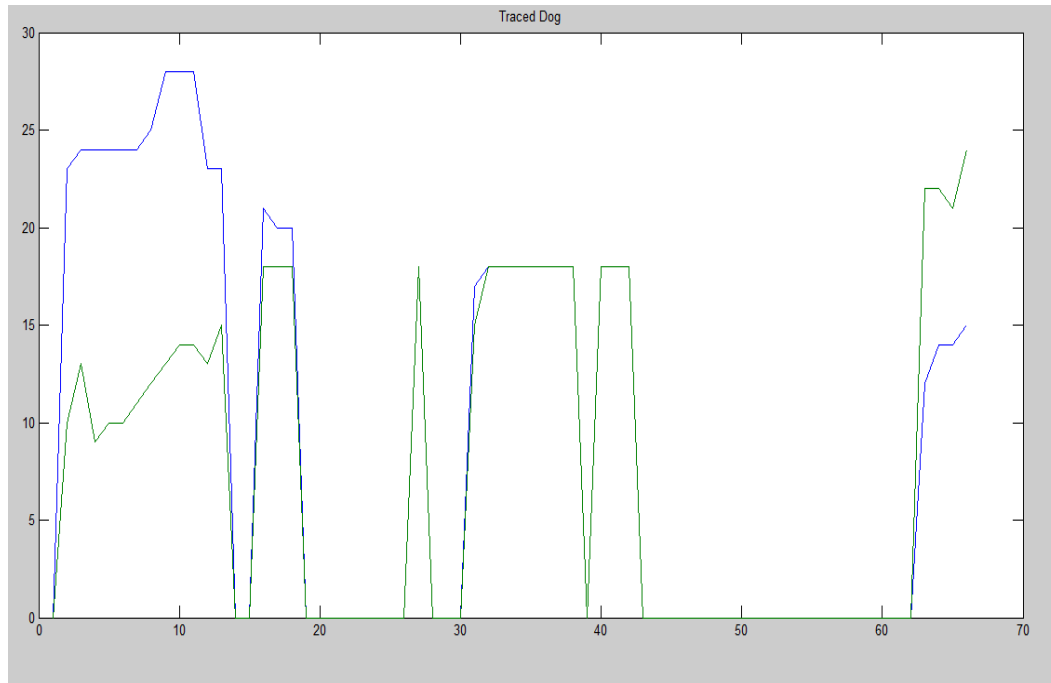


Tracking results, contain the detection of a moving as well how their motions are being tracked and classified (i.e) categorized. This result is how the multiple object are being recognized.



ANALYSIS

It will tell us detailed about how the tracking, classification done in each frame. Here I had taken the height, width of the object in each frame in Y- axis and total number of frames in X – axis.



GREEN -> Width.

BLUE -> Height.

Video Specification

- Frame width : 320, Height : 240
- Frame Rate : 25 frames / sec.

Total.No.of.Frames	No.Of.Frames Containing Objects	Classified Object
226	226	FLIGHT
1399	400	CAR



66	39	MULTIPLE TRACKING-CAR,HUMAN
347	347	HELICOPTER

CONCLUSION

The trackers are divided into many categories based on the tracked target. The classification is only relative because there are still many tracking algorithms which combine different approaches of tracking. Naive Bayes provides the better classification by its class probability feature. We divide the tracking methods into three categories based on the use of object representations, namely, methods establishing point correspondence, methods using primitive geometric models, and methods using contour evolution. Note that all these classes require object detection at some point. For instance, the point trackers require detection in every frame, whereas geometric region or contours-based trackers require detection only when the object first appears in the scene. Moreover, we describe the context of use, degree of applicability, evaluation criteria, and qualitative comparisons. Extensive experiments demonstrate the efficiency of the proposed method. We believe that, this article, an object tracking and recognition with a rich bibliography content, can give valuable insight into this important research topic and encourage new research.

REFERENCES

1. Weiming Hu, Tieniu Tan, Fellow, IEEE, Liang Wang, and Steve Maybank, A Survey on Visual Surveillance of Object Motion and Behaviors in *Ieee Transactions On Systems, Man, And Cybernetics—Part C: Applications And Reviews*, Vol. 34, No. 3, August 2004.



Kavitha.A et al, International Journal of Computer Science and Mobile Applications,

Vol.2 Issue. 1, January- 2014, pg. 63-78

ISSN: 2321-8363

2. SERBY, D., KOLLER-MEIER, S., AND GOOL, L. V. 2004. Probabilistic object tracking using multiple features. In *IEEE International Conference of Pattern Recognition (ICPR)*. 184-187.
3. COMANICIU, D., RAMESH, V., AND MEER, P. 2003. Kernel-based object tracking. *IEEE Trans. Patt. Anal. Mach. Intell.* 25, 564–575.
4. YILMAZ, A., LI, X., AND SHAH, M. 2004. Contour based object tracking with occlusion handling in video acquired using mobile cameras. *IEEE Trans. Patt. Anal. Mach. Intell.* 26, 11, 1531–1536.
5. SILVEIRA, G. AND MALIS, E. 2007. Real-time visual tracking under arbitrary illumination changes. *IEEE International Conference on Computer Vision and Pattern Recognition*, 1–6.
6. WU, Y. AND FAN, J. 2009. Contextual flow. *IEEE International Conference on Computer Vision and Pattern Recognition*, 33–40.
7. BRADSKI, G. 1998. Real time face and object tracking as a component of a perceptual user interface. *IEEE Workshop on Applications of Computer Vision*, 214–219.
8. F. Heijden. *Image Based Measurement Systems: Object Recognition and parameter Estimation*. Wiley, January 1996