International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 1 Issue 5, July - 2012

Object detecting using PCA image reconstruction and Optical flow

Sajjad einy
Mtech spatial information technology
JNTU university

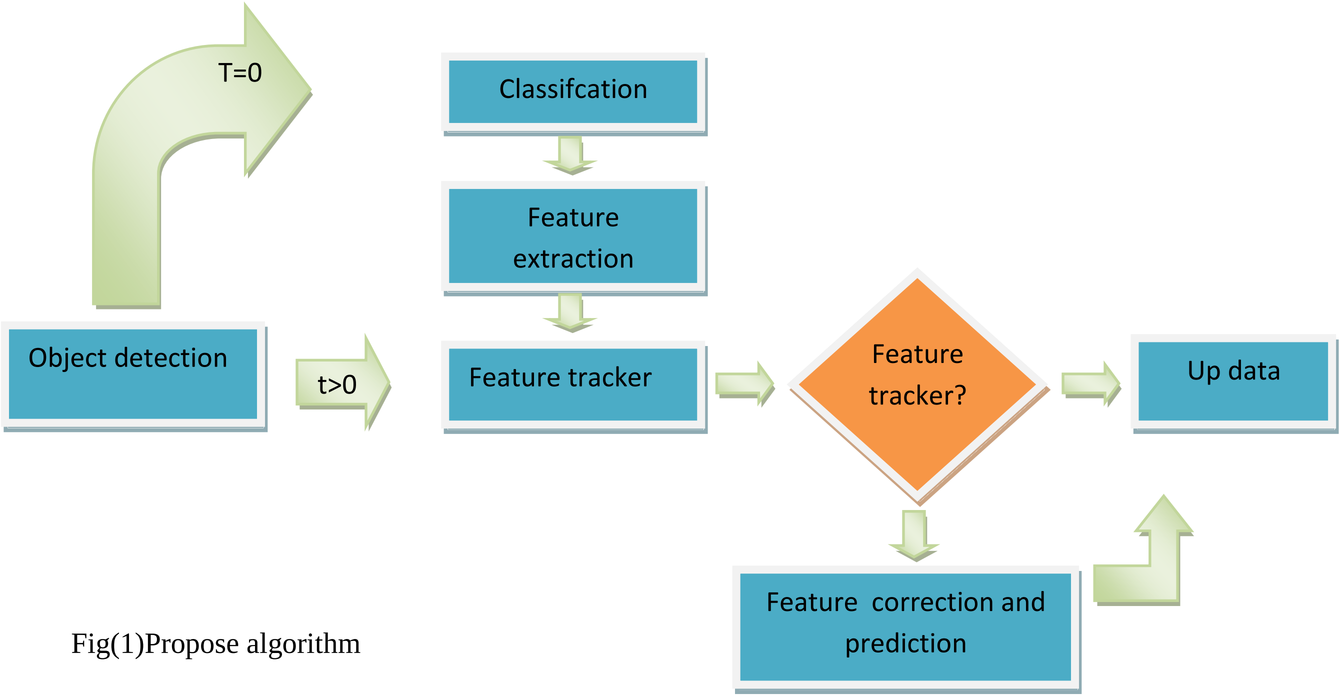
Anoop M. Namboodiri
Assistant Professor IIIT hyderabad

Abstract:

This paper addresses the issue of moving object tracking from video. Two steps of processing are PCA reconstruction for classification the area and Optical flow-based tracking of feature points. Optimal flow-based tracking algorithm predicts and restores the feature of area in the real time object tracking. The proposed system is computationally more efficient for learning of dynamic object and object tracking to shape information. PCA based method compare to other methods is powerful algorithm in object detection and the Optical flow-based tracking algorithm could be used for decreasing the false point tracking and remove the error of the motion tracking in real time system tracking. The proposed algorithm tracks a set of feature points; during the tracking, feature is restored inside the predicted region. One important contribution of this work is to provide a restoration process for missing feature points, which occurs at almost every frame under realistic, noisy environment.

Introduction

Principal Component Analysis is a popular technique for data compressor and has been successfully used as an initial step in many computer vision tasks, including face recognition and object recognition and feature extraction. In this paper we are using PCA in image reconstruction for extraction and classification objects. We present an object tracking system to detect pedestrians in gray level images, with assuming the correction and restoration system. The system works as follows: feature extraction using by PCA reconstruction system second tracking the Objects using by optimal flow-based tracking algorithm. When a new pattern needs to be classified we compare the reconstruction made by training sets of principal components (PCs). In order to improve the performance of the classifier we can use the training set of PCA classifier and feature point for correction and restoration the information of image. Additionally, we show that the performance of the system can be improved by combining the classifier based on PCA reconstruction with a feature points using a Support Vector Machine. By definition, PCA looks for the set of PCs that best describe the distribution of the data that are being analyzed. Therefore, these PCs are going to preserve better the information of the images from which PCA was performed, or of those that are similar. Thus, if we have a set of PCs that were obtained from a set of pedestrian images only, these must reconstruct better the images of other pedestrians than any other type of images, and vice versa, if we have a set of PCs obtained from images of anything except pedestrians, the reconstruction of the pedestrian images will not be as good [1]. in the fig1 we show the implementation algorithm of object tracking using by feature extraction in this algorithm after classification of the image using PCA then extraction of them used feature point for correction and up data the feature that are tracks. After detecting moving objects from background, we extract a set of feature points inside the object and predict the corresponding feature points in the next frame. We keep checking and restoring any missing feature points during the tracking process. If over 60% of feature points are restored, we decided the set of feature points are not proper for tracking, and redefine new set of points [7].



Fig(1)Propose algorithm

1 Image reconstruction with PCA:

The formulation of standard PCA is as follows. Consider a set of m images, each of size r · c. Each image Ii is represented by a column vector v_i of lengthr_c. The mean object of the set is defined by

$$\mu = \frac{1}{m}\sum_{i-1}^{m} v_i$$

C, the covariance matrix, is given by

$$c = \sum_{i-1}^{m} (v_i - \mu)(v_i - \mu)^t$$

The principal components are then the eigenvectors of C. These eigenvectors can be computed in several ways. Perhaps the easiest one is to solve the generalized eigenvector problem using the QZ algorithm or its variants [1,2].It is also common to formulate the problem as that of finding the basis vectors that minimize the reconstruction error and then solve it using standard least-squares techniques In our system we compute these eigenvectors using the implementation provided by Matlab, which is based on the QZ algorithm [1,2].

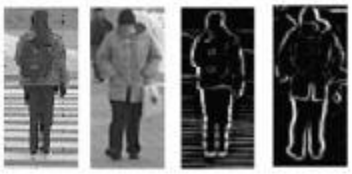


Fig2.Image Reconstruction with deferent sets of PCs

1.2 Classification using reconstruction

From this fact we can create a classifier based on image reconstruction with PCA, which decides if an image belongs or does not belong to the pedestrian class. The algorithm to do this classification is the following:

Before doing any classification:

1. Perform PCA on the set of pedestrian gray level images to obtain the projection matrix p_{gp} and the mean μ_{gp}

2. Perform PCA on the set of pedestrian edge images to obtain the projection matrix $p_{gp}$and the meanμ_{ep}.

3. Perform PCA on the set of non-pedestrian gray level images to obtain the projection matrix p_{gp} and the meanμ_{gn}.

4. Perform PCA on the set of non-pedestrian edge images to obtain the projection matrix Pen and the mean μ_{en}.[1]

$$class\ (g)\begin{cases} pedestrian & d \geq 0 \\ non-pedestrian & d < 0 \end{cases}$$

1.3 feature extraction using by clustered

If a window is identified correctly as a pedestrian, then it is very likely that there are no pedestrians either above or below it, and if there are pedestrians beside it, they cannot be too overlapped. This heuristic allows us to eliminate nearby detections. With this purpose, we define a region around a detection which we are going to use to eliminate any detection whose centroid is inside this region. The size of the region was defined empirically as 1.4 times the detection height upwards and downwards from the centroid and between 0.5 and 0.75 times the detection width towards each side of the centroid. We know that when we have multiple detections we must choose only one to keep, but how do we make this decision A reasonable way to choose is to maintain the grouped detection composed by the most original detections; nevertheless, we observe that usually the biggest detections were the correct ones, due to the fact that arms, legs and head are often confused with pedestrians, so when we need to decide among a set of detections that are in the same region, we must consider the number of times that they have been detected originally as well as the size of the detected regions. To achieve this, the detections that compose a grouped detection are weighted by their height, then the grouped detection with the greatest Preference, according to the following formula is chosen [1].

Preference =Detections .Weight (height)

where Detections is the number of original detections that compose the grouped detection that we are evaluating and Weight is a function that determines the value that each detection has, according to the height of the grouped detection, and is given by the formula: Weight(height)=(height- 50)(height-50) There are very few cases where this heuristic does not work, and thus it allows to eliminate many false detections when the classifier confuses the arms, the legs, or some other object with a pedestrian[1]

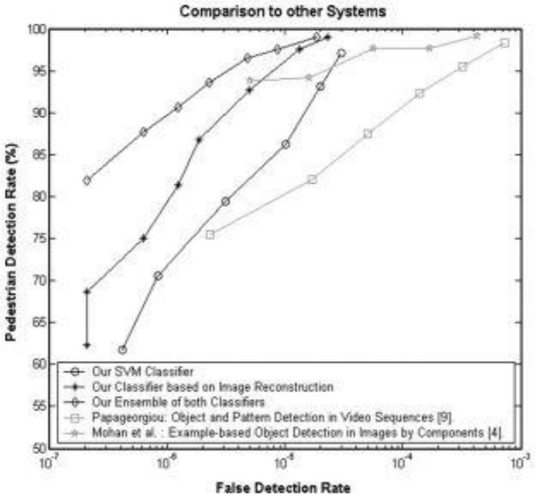


fig3.False detection rate

Fig. 3. ROC curves comparing the performance of our classifiers versus the best reported in the literature. The detection rate is plotted against the false detection rate measure on logarithmic scale.

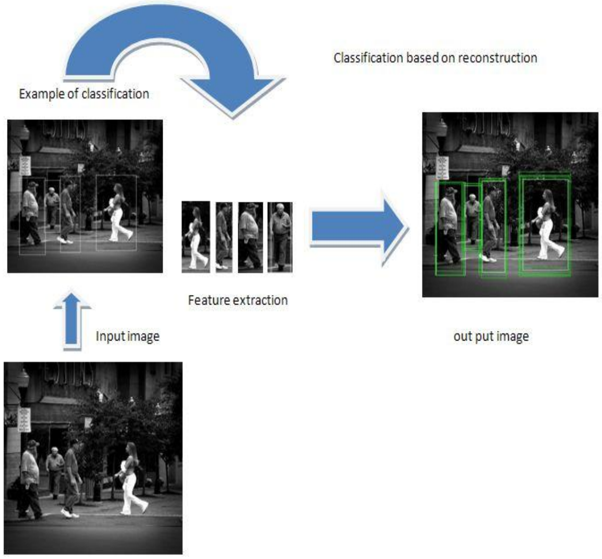

Example of classification
Classification based on reconstruction
Feature extraction
Input image
out put image

Fig4.object classification algorithm

2 Optical flow-based tracking of feature points

Optical flow is approximated with the displacement of features between two consecutive frames . It is interesting to see how such an approximation is good for motion generated by hand held cameras and when otherwise it does lead to serious errors.

2.1 Motion-based object detection

For the detection of an object from background, we use optical flow as an initial clue. The fundamental condition of optical flow is that intensity of a point on the object does not change during the sufficiently small duration. Let $s_c(x,y,t)$ represent the distribution of intensity in a continuous frame, the optical flow condition can be defined as[8,9,7]

$$\frac{ds_c(x,y,z)}{dt} = 0 \qquad (1)$$

By applying the chain rule to (1), we have that[2,1]

$$\frac{\partial s_c(x,y,z)}{\partial x} v_x(x,y,z) + \frac{\partial s_c(x,y,z)}{\partial y} v_y(x,y,z) + \frac{\partial s_c(x,y,z)}{\partial t} v_t(x,y,z) = 0 \qquad (2)$$

Where $v_x(x,y,z) = dx/dt$ and $v_y(x,y,z) = dy/dt$ From (2), we can evaluate optical flow as $\varepsilon_{of}\big(\upsilon(x,y,t)\big) = \{\nabla\, s_c(x,y,t), \upsilon(x,y,t)\} + \frac{\partial s_c(x,y,t)}{\partial t}$ =0 (3)

Where $\nabla\, s_c(x,y,t) = [\frac{\partial s_c}{\partial x} * \frac{\partial s_c}{\partial y}]^t$, $\upsilon(x,y,t) = [\upsilon_1(x,y,t)\, \upsilon_2(x,y,t)]^t$

And <0.0> denotes vector inner product[13]. Optical flow that satisfies the constraint in (3) is prone to noise because the difference approximation between adjacent pixels is used for evaluating derivatives. To reduce noise amplification,Horn–Schunk’s and Lukas–Kanade’s methods are widely used in the literature[10,11,7]. Error from the optical flow constraint can be measured as

$$E=\sum_{(x,y)} e_{of}^2$$

Where R represents a neighboring region [8].In minimizing (4), we ignore trivial motion vectors so as to reduce error as long as making noise[7].In order to separate a moving object and noise in low-gradient conditions, we used the following measure, which is called the normalized difference:

$$d_i=\frac{1}{n_i}\sum_{(x,y)} |i_t\,(x,y) - i_{t-1}(x,y)|$$

n_i The total number of pixels in R. If the normalized difference is smaller than a pre-specified threshold denoted by T_d,the corresponding region is considered to be noise. As a result, the region of moving objects can be extracted, and the corresponding region is then labeled based on motion direction [8].

2.2 Feature point extraction

After detection of an object from background, we extract a set of feature points inside the object by using the BOUOGUT tracking algorithm [12,13]. Due to the nature of motion estimation, motion-based object detection algorithms usually extract the object slightly larger than the real size of the object, which results in false extraction of feature points outside the object Let the position of a feature point at frame t be v_t^i , where I represents the index of feature points and v_t^i=$[x_t^i, y_t^i]$,These outside feature points are removed by considering the distance between feature points given as

$$v_t^i=\begin{cases} v_t^i & d=<t \\ 0 & d > t \end{cases}$$

Where d=$\sum_{t=0}^{k-1} \sqrt{(x_t^{i-1} - x_t^i)}(y_t^{i-1} - y_t^i)$

t represents the index of frames, and i the index of feature points. Heredi represents the sum of distance betweent th frame and t+1st frame with respect to I th feature point. In general, the moving distance of a feature point in the background (outside object) is further less than that of a feature point in the tracked object[7]in the fig 1 we present the effect of t amount in the propose algorithm.

2.3 Feature point prediction and correction

Sometimes, a tracking algorithm may fail to track a proper feature point in the next frame. A feature point is defined as untracked when an error value within small window is over a pre-defined threshold. Specifically, a threshold value is determined by distance between average vectors predicted by a spatio-temporal prediction. After the spatio-temporal prediction, reinvestigation is performed. Then, both tracked and untracked feature points are updated in a list. In many real-time, continuous video tracking applications, a feature-based tracking algorithm fails due to the following reasons: (i) self or partial occlusions of an object and (ii) feature points on or outside the boundary of the object, which are affected by changing back-ground. In order to deal with the tracking failure, we should correct the erroneously predicted feature points by using the location of the previous feature points and inter-pixel relationship between the predicted points. A temporal prediction is suitable for deformable objects while a spatial prediction is good for non-deformable objects. Both temporal and spatial prediction results can also be combined with proper weights. Although users can control the value of the number of frames, K, the value of 7 was used for all test sequences. The larger the value of Kis, the better the performance of the algorithm is. Because of trade-off between processing time and accuracy, the value around 7 was found to be reasonable for temporal prediction[7].

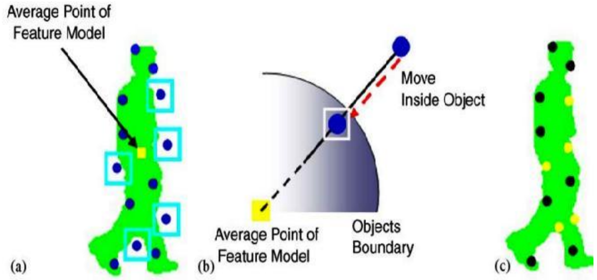

Average Point of Feature Model
(a)
Move Inside Object
Average Point of Feature Model
Objects Boundary
(b)
(c)

Fig5:(a) feature model (b) relocation (c) result of feature reconstruction

Conclusion:

This method is suitable for dynamic background. The experimental results have shown that the proposed method has successfully tracked moving objects in most cases. Applying the PCA reconstruction algorithm caused better realization of information related to the interest area by using both intensity and edge image.

References:

[1]Object detection using image reconstruction with PCA Luis Malago´n-Borja a, Olac Fuentesb,* Image and Vision Computing (2007)

[2]Edgar Osuna, Robert Freund, Federico Girosi, Training support vector machines: an application to face detection, in: IEEE Confer-ence on Computer Vision and Pattern Recognition, 1997, pp. 130

[3] Constantine Papageorgiou, Object and pattern detection in video sequences, Master’s thesis, M.I.T., Cambridge, MA, 1997.

[4] John C. Platt, Sequential minimal optimization: a fast algorithm for training support vector machines, in: Bernhard Scholkopf, Christo-pher J.C. Burges, Alex J.

Smola (Eds.), Advances in Kernel Methods:Support Vector Learning, MIT Press, Cambridge, MA, USA, 1999,pp. 185–208.

[5] C.B. Moler, G.W. Stewart, An algorithm for generalized matrix eigenvalue problems, SIAM Journal on Numerical Analysis 10 (2)(1973) 241–256

[6] Massimiliano Pontil, Alessandro Verri, Support vector machines for 3D object recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 20 (6) (1998) 637–646

[7] Optical flow-based real-time object tracking using non-prior training active feature model Jeongho Shin a , Sangjin Kim a, Sangkyu Kang b, Seong-Won Lee c, Joonki Paika,Besma AbidId, Mongi Abidida

[8]Haritaogˇlu I, Harwood D, Davis L. W-4: Real-time surveillance of people and their activities. IEEE Transactions on Pattern Analysis and Machine Intelligence 2000;22(8):809–30.

[9] Amer A. Voting-based simultaneous tracking of multiple video objects. Proceedings of the SPIE Visual Communication Image Processing 2003;5022:500–11.

[10] Wren C, Azerbayejani A, Darrel T, Pentland A. Pfinder: Real-time tracking of the human body. IEEE Transactions on Pattern Analysis and Machine Intelligence 1997;19(7):780–5.

[11] Comaniciu D, Ramesh V, Meer P. Real-time tracking of nonrigid objects using mean shift. Proceedings of the IEEE International Conference on Computer Vision, Pattern Recogni-tion 2000;2:142

[12] Comaniciu D, Ramesh V, Meer P. Kernel-based object tracking.IEEE Transactions on Pattern Analysis and Machine Intelligence2003;25(4):1–14.

[13] Baumberg AM. Learning Deformable Models for Tracking Human Motion. PhD dissertation, School of Computer Studies,University of Leeds, UK, October 1995.