ISSN: 2277-9655
Impact Factor: 1.852

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

Human Tracking System

Mrs. Kavita V. Wagh[*1], Dr.R.K.Kulkarni[2]
[*1]Professor, EXTC, VESIT, Mumbai, India
[2]Student, EXTC, VESIT, Mumbai
Waghkav@gmail.com

Abstract

This paper presents a comprehensive framework for tracking coarse human model from sequences of synchronized monocular grayscale images in single or multiple camera coordinates. It demonstrates the feasibility of an end-to-end person tracking system using a unique combination of motion analysis from sequences of synchronized monocular grayscale images in different camera Coordinates and other existing techniques in motion detection, segmentation, and patter recognition. This human tracking is an important task in many vision applications. The main steps in video analysis are two: detection of interesting moving objects and tracking of such objects from frame to frame. In a similar vein, most tracking algorithms use pre-specified methods for preprocessing.

There are several objects tracking algorithms i.e. Meanshift, Camshift, Kalman filter with different pre-processing methods. The system starts with tracking from a single camera view. When the system predicts that the active camera will no longer have a good view of the subject of interest, tracking can be switched to another camera which provides a better view and requires the least switching to continue tracking. The nonrigidity of the human body is addressed by matching points of the middle line of the human image spatially and temporally, using Bayesian Classification schemes. Multivariate normal distributions are employed to model class-conditional densities of the features for tracking, such as location, intensity, and geometric features.

Keywords: Kalman filter, Principal component analysis, preproccesing , tracking

Introduction

Tracking motion is of interest in numerous applications such as surveillance, analysis of athletic performance, and content-based management of digital image databases. Recently, growing interest has concentrated upon tracking humans using distributed monocular camera systems to extend the limited viewing angle of a single fixed camera. In such a setup, the cameras are arranged to cover a monitored area with overlapping vision fields to ensure a smooth switching among cameras during tracking. Here a comprehensive framework for tracking coarse human models across multiple camera coordinates and demonstrate the feasibility of an end-to-end person tracking system with existing techniques in motion detection, segmentation, and pattern recognition is presented. The nonrigidity of the human body is addressed by matching points of the middle line of the human image, spatially and temporally, using Bayesian classification schemes.

Considered the moving human image as a combination of various blobs. All distributed cameras were calibrated in the world coordinate system, which corresponds to a CAD model of the indoor environment. The blobs of body parts were matched through image sequences using the area, average brightness, and rough 3D position in the world coordinates adopted a similar strategy to construct a 3D environmental model using the voxel feature. The depth information contained in the voxel is obtained using height estimation. Moving humans were tracked as a group of these voxels from the "best" angle of the viewing system. Neither of these methods considered the particular body structure and shape characteristics of a human being. In addition, both need to model the environment in 3D and establish a world coordinate [3]. They are computationally expensive and do not adapt to changes in dynamic environments. Based on studies on human geometric structures, distinguishing moving human figures from other nonhuman objects by modelling the human body. Matching the subject image between consecutive frames involves motion estimation in a spatial-temporal domain under a Bayesian classification scheme. Tracking is done from a single camera view until the system predicts that the active camera soon will no longer have a good view of the subject of interest. But there are methods in which

 ISSN: 2277-9655
Impact Factor: 1.852

tracking switches to the camera that will provide a better view and require the least switching to continue tracking. Thus, the tracking paradigm consists of three basic modules: Single View Tracking (SVT), Multiple View Transition Tracking (MVTT).

Single View Tracking

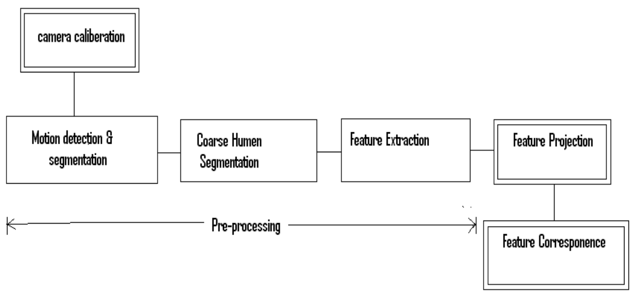

Fig.1: Basic procedure of tracking.

As shown in fig.1 tracking from a single view includes two major components:
Preprocessing and feature correspondence between consecutive frames. Three stages of preprocessing are performed:

1. Segmenting the moving objects from the still background,
2. Distinguishing human subjects from other segmented non background objects, and
3. Extracting features from the segmented human subjects.

Feature correspondence is established by applying a Bayesian classifier to locate the most likely match of the subject image in the next frame. The feature vector consists of location, intensity, and geometric information, Multivariate Gaussian models are formulated to parameterize the class conditional probability density of the feature vector. Thus, tracking is reduced to finding the minimum sum **of** the corresponding Mahalonobis distances of the feature given the estimated feature parameters.

Pre-processing

Pre-processing is critical to the **success** of high-level processing stages. If a moving humen is missed at the pre-processing stage, the system will be unable to track this particular object at later stages.

The major task of preprocessing [19] is to segment human images from the rest of the image objects. There are still no satisfying and robust general solutions. Here, we apply efficient standard motion detection and segmentation techniques to take the advantage of the fact that the viewing system is still more robust and complicated segmentation schemes could be applied if computational cost is not a consideration.

Due to their robustness for matching in different views, ***N*** points belonging to the middle line of the upper body are selected and aggregated as the feature to track. The line segment is extracted by finding the middle points of the blobs. Using multiple feature points instead of a single point [6] makes matching the subject image more reliable. We have elected to use six points based on the trade-off between the need to use fewer points to reduce computation cost and the need to use more points due to the nonrigidity of moving human figures.

To ensure the robustness of the feature matching, we incorporate three types of features: location, intensity, and geometry.

- The location feature is defined as the horizontal and vertical position of the feature points: $X_t = [(u_{1t},v_{1t}), (u_{2t},v_{2t}), \ldots, (u_{nt},v_{nt})]^T$, where ***t*** is the time index.
- We define the intensity feature as $Y_t = [y_{1t}, y_{2t} \ldots y_{nt}]^T$, in which $\mathbf{y_{mt}}$ is the average intensity of the neighborhood of the mth feature points.
- Another type of feature is the image height ratio between consecutive frames (the height of a candidate image in the current frame divided by the subject height in the previous frame) as the geometric feature (g_t), where the image height is computed as the height of the upper body using **a** coarse 2D geometric human model at the segmentation stage. This feature is essential for tracking in narrow corridor scenes where the location and intensity features most likely fail.

Background Subtraction

Object detection can be achieved by building a representation of the scene called the background model and then finding deviations from the model for each incoming frame. Any significant change in an image region from the background model signifies a moving object. Usually, a connected component algorithm is applied to obtain connected regions corresponding to the objects. This process is referred to as the background subtraction. A substantial improvement in background modelling is achieved by using multimodal statistical models to describe per-pixel background color. For instance, use a mixture of Gaussians to model the pixel color. In this method, a pixel in the current frame is checked against the background model by comparing it with every Gaussian in the model until a matching Gaussian is found. If a match is found, the mean and variance of the matched Gaussian is updated, otherwise a new Gaussian with the mean equal to the current pixel color and some initial variance is introduced into the mixture. Each pixel is classified based on whether the matched distribution represents the background process.

Moving regions, which are detected using this approach, along with the background models are shown in Fig.2.

Fig .2. Eigenspace decomposition-based background subtraction [19]:
(a) An input image with objects, (b) Reconstructed image after projecting input image onto the eigenspace, (c) Difference image. Note that the foreground objects are clearly identifiable

Algorithm Used

The algorithm used is a non-parametric method. It provides accurate localization and efficient matching without expensive exhaustive search. The size of the window of search is fixed. It is an iterative process, that is to say, first compute the meanshift value for the current point position, then move the point to its meanshift value as the new position, then compute the meanshift until it fulfill certain condition. For an frame, we use the distribution of the levels of grey which gives the description of the shape and we are going to converge on the centre of mass of the object calculated by means of moments. The flow chart of meanshift in fig.3 described the steps of the algorithm. The number of iterations of the convergence of the algorithm is obtained when the subject is followed within the image sequence.

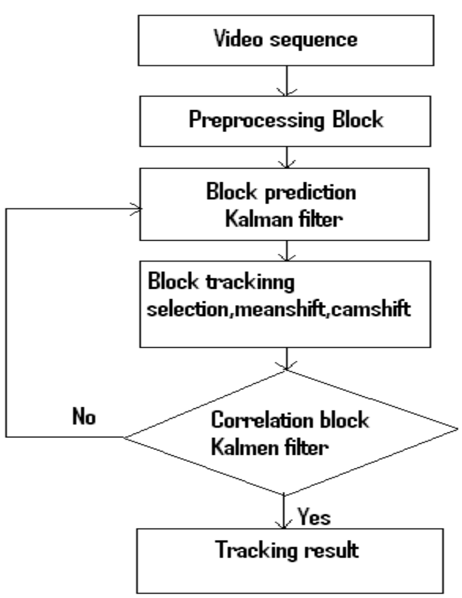

Fig.3 Flow chart of the human tracking system

a. Block Kalman filter

The Kalman filter estimates the position of the object in each frame of the sequence [6], [5]. The input parameters of the Kalman filter, respectively, the position of the object in the image at time *k*, the size of the object and the width and length of the search window of the object which vary due to the mobility of the object during the sequence. These parameters represent the state vector and measurement vector of the Kalman filter [3]. In general, the estimation of parameters followed with a Kalman filter is a process that requires the following steps:
Step 1: The measure is to take the tracking parameters calculated by the algorithm Camshift.
Step 2: The estimate, which updates the position of the object.
Step 3: The prediction, which calculates the position of the object in the next frame.

b. Steps for Calmshift algorithm:

Step1: Set the image as the search area.
Step2: Initialize the size and location of the search window
Step3: calculating the probability distribution of color in the search window.
Step4: Run MeanShift to obtain a new location and size of search window.
Step5: In the next frame of video images, initialize location and size by step3. And Jump to step3 continues to run.

Face Recognition Based on Eigenfaces Generation Using PCA Algorithm

In the project Eigenfaces generation based on PCA algorithm is used which is well studied method of face recognition. PCA is a method of transforming a number of correlated variables into a smaller number of uncorrelated variables. Similar to how Fourier analysis is used to decompose a signal into a set of additive orthogonal sinusoids of varying frequencies, PCA decomposes a signal (or image) into a set of additive orthogonal basis vectors or *eigenvectors*. The main difference is that, while Fourier analysis uses a fixed set of basis functions, the PCA basis vectors are learnt from the data set via unsupervised training. PCA can be applied to the task of face recognition by converting the pixels of an image into a number of eigenface feature vectors, which can then be compared to measure the similarity of two face images.

Training the face detector requires the following steps:

1. Calculate the mean of the input face images
2. Subtract the mean from the input images to obtain the mean-shifted images
3. Calculate the eigenvectors and eigenvalues of the mean-shifted images

ISSN: 2277-9655
Impact Factor: 1.852

4. Order the eigenvectors by their corresponding eigenvalues, in decreasing order
5. Retain only the eigenvectors with the largest eigenvalues (the *principal components*)
6. Project the mean-shifted images into the eigenspace using the retained eigenvectors.

Detailed steps to find out best match :

1. Initially obtain the zero-mean face images.
2. Calculating the eigenvectors and eigenvalues.
3. The output from from previous step is a matrix of eigenvectors.
4. Truncate the eigen vector matrix to maximum principal components
5. The number was selected somewhat arbitrarily.
6. Next step is achieved by achieved by projecting the mean- shifted input images into the subspace defined by truncated set of eigenvectors.
7. Once the face images have been projected into the eigenspace, the similarity between any pair of face images can be calculated by finding the Euclidean distance (y_1-y_2) between their corresponding feature vectors y_1 and y_2.
8. The smaller the distance between the feature vectors, the more similar the faces.
9. It is simply defined by similarity score s (y_1,y_2) based on the inverse Euclidean distance.
10. To perform face recognition, the similarity score is calculated between an input face image and each of the training images. The matched face is the one with the highest similarity, and the magnitude of the similarity score indicates the confidence of the match.

Human face images are very similar in all configuration and they can be described by some "basic face images". According to this idea, one can find the "basic faces" that best account for distribution of face images within the entire face space using the principal component analysis. The "basicfaces" are so-called "eigenfaces".

Let the train set of face images be $\tau_1, \tau_2 \ldots.. \tau_M$.,

The average face of the set is defined by $\Psi = \tau1 + \tau2 + \cdots \tau_M/M$ and different faces from the average face are $\Phi_i = \tau_i - \Psi$ where i=1,2….M. If a face image is considered as a vector whose length is equal to the number of the face image pixels, then the kth eigenvalue then λ_k and the eigenvector of face vectors is chosen such that

$$CU_k = \lambda_k U_k \qquad (1)$$

$$U^t_k U_k = 1 \text{ for } k=n, \quad 0 \text{ for } k \neq n \qquad (2)$$

$$C = \frac{1}{M} \sum_{n=1}^{M} \Phi_n \Phi_n^t = AA^t \qquad (3)$$

$$A = \frac{[\Phi 1, \Phi 2 \ldots\ldots \Phi M]}{\sqrt{M}} \qquad (4)$$

a. Average face : $\Psi = \frac{1}{M} \sum_{t-1}^{M} \tau_l$

b. Different faces : $\Phi_k = \tau_k - \Psi$

c. Compute matrix AA^t where each element is

$a_{mn} = (\Phi_m^t \Phi_n)/M$

d. Compute eigen vectors X_k and eigen values λ_k of the matrix $A^t A$

e. Eigen value U_k are

$U_k = (\sum_{k=1}^{M} \Phi_k X_{kl}) / \sqrt{(\lambda_k M)}$

Result

The tests video sequences was a public in Internet as shown in fig.4 . This video has been adopted by many researchers to test their algorithms, because of its capacity in simulating various tracking conditions, including illumination changes, pose variations, occlusions, and distraction. Since ensemble tracking is a general framework for tracking objects, several object tracking algorithms (Meanshift, Camshift, Kalman filter),and a pre-specified methods for preprocessing (method of histogram color, method of background subtraction and method of detection skin color are evaluated to achieve a good tracking performance. An adaptive first pre-processing the histogram color was tested to the video sequences. The algorithm tracks the object as it moves from one frame to the next one in approximately 0.15 second on a 2.2GHz PC in Matlab. The first expriment is on a video sequence of glass in hand man with 100 frames of spatial resolution 320×240. The tracking target is the moving is the glass. The second experiment is on a sequence with 100 frames of spatial resolution 144 × 176. In this video, we will track a motion of face. As shown in Fig. 3, the trajectories of the glass for the three algorithms tracking by the Camshift algorithm and the estimated trajet location of the Kalman filter in the frame plane.

Fig.4 . Tracking results of 3 sequences video by the Histogram methods.

Conclusion

This paper focuses on simultaneous tracking of human. During the test sequences generated with different methods of pre-processing, we can conclude that the tracking of human differs from one human to another and several parameters can affect the results of tracking. Experimental results show that our algorithm (Camshift and the Kalman filter) is superior in terms of precision, reliability and execution time, compared the various methods presented in the literature. In particular, the use of several methods preprocessing to detect the object in each frame of the sequence, provides satisfactory results in the case of the complex video sequence

Acknowledgement

I would like to take the opportunity to express my feelings towards my guide Mr. Dr.R.K.Kulkarni, college, my colleagues and the staff and family who help me in bringing out this Journal. So I wish to thanks Mr. Dr.R.K.Kulkarni for their help and continuous encouragement and there valuable guidance and support.

References

[1] A. Alper and J. Omar and S. Mubarak, “ Object Tracking: A Survey”. ACM Computing Surveys, vol. 38, no. 45, pp. 1–45. 2006.
[2] D. Chen and F. Bai and B. Li and T. Wang, “ BEST: A Real-time Tracking Method for Scout Robot”. IEEE/RSJ International Conference on Intelligent Robots and Systems, no. 6, pp. 1001–1006. 2009.
[3] J. Wang and F. He and X. Zhang and G. Yun, “ Tracking Object throuth Occuusions Using Improved Kalman Filter”. IEEE 2010, no. 6, pp. 223-228. 2010.
[4] N.A Gmez, “ A Probabilistic Integrated Object Recognition and Tracking Framework for Video Sequences”. Universitat Rovira I Virgili, PHD thesis, Espagne, 2009.
[5] X. Sun and H. Yao, and S. Zhang, “ A Refined Particle Filter Method for Contour Tracking”. Visual Communications and Image Processing, no. 8, pp. 1–8. 2010.
[6] D. Comaniciu, and V. Ramesh, and P. Meer , “ Kernel-Based Object Tracking”. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 25 no. 14, pp. 564–577. 2003.
[7] L. Ido, and L. Michael, and R. Ehud, “ Mean Shift tracking with multiple reference color histograms”. Computer Vision and Image Understanding, no. 9, pp. 400–408. 2010.
[8] G. bradski, and R. Gary, and L.M. Olga, “ Computer Vision Face Tracking For Use in a Perceptual User Interface”. Intel Technology Journal Q2’98, vol. 10, no. 15, pp. 1–15. 1998.
[9] G. Bradski, and T. Ogiuchi, and M. Higashikubo, “ Visual Tracking Algorithm using Pixel-Pair Feature”. International Conference on Pattern Recognition, no. 4, pp. 1808–1811. 2010.
[10] Y. Ruiguo, and Z. Xinrong, “ The Design and Implementation of Face Tracking in Real Time Multimedia Recording System”. IEEE Transaction, no. 3, pp. 1–3. 2009.
[11] E. David, and B. Erich, and K. Daniel, and S. Anselm, “ Fast and Robust Camshift Tracking”. IEEE Transaction, no. 8, pp. 1–8. 2010.
[12] S. Avidan, “ Support vector tracking”. In IEEE Conference on Computer Vision and Pattern Recognition, no. 8, pp.184–191. 2001.
[13] S. Franois, B. and R.J. Alexandre, “ Camshift Tracker Design Experiments”. IMSC, no. 11, pp. 1–11. 2004.
[14] L. Zhang, and D. Zhang, “ Robust Object Tracking Using Joint Color Texture Histogram”. International Journal of Pattern Recognition and Artificial Intelligence, vol. 23 no. 19, pp. 1245–1263. 2009.
[15] X. Song, and R. Nevatia, “ Camshift Tracker Design Experiments with Intel OpenCV and Sai”. International Conference on Pattern Recognition, IEEE Computer Society, no. 4, pp. 1–4. 2004.
[16] S. Afzal, and N. Vaswani, and A. Tannenbaum ,A. and Yezzi, “ Object Tracking: A Survey”. IEEE, no. 6, pp. 4244–4249. 2009.
[17] A. Iyad, and H. Mahmoud, “ Smart Human Face Detection System”. International Journal of Computers, vol. 5, no. 8, pp. 210–221. 2011.
[18] C. Abbas, and C. Joan, and C. Kevin, and M.K. Paul “ A Skin Tone Detection Algorithm for an Adaptive Approach to Steganography”. , no. 23, pp. 1–23. 2011.

ISSN: 2277-9655
Impact Factor: 1.852

[19] Alper Yilmaz, Omar Javed and Mubarak Shah *University of Central Florida* “Object Tracking: A Survey” ACM Computing Surveys, Vol. 38, No. 4, Article 13, Publication date: December 2006.
[20] http://developer.com/technology/itj.
[21]http://ftp.pets.rdg.ac.uk/PETS2001/DATASET1/TESTING/.
[22] http://developer.com/technology/itj.
[23]http://ftp.pets.rdg.ac.uk/PETS2001/DATASET1/TESTING/.