Proceedings of the 2007 IEEE Symposium on Computational
Intelligence in Image and Signal Processing (CIISP 2007)

Bare Bones Strategy for Human Detection and Tracking

M. N. Siddiqui and B. Yousaf
National University of Computer and Emerging Sciences
Islamabad, Pakistan
naweedsiddiqui@yahoo.com, bilalyousaf@yahoo.com

Abstract

We present a scaled down version of a Human Detection and Tracking System designed to run on relatively low-end machines of developing countries. The system uses sequence of monocular images of a single, fixed surveillance camera to extract data of moving objects. A Linear Motion Model coupled with Pre-computed Average Color Intensities is used to track the detected subjects across a series of frames. Head Detection algorithm is applied to form multiple hypotheses so as to accurately detect individuals in case of occlusion caused by people overlapping with each other. A final Shape Fitting algorithm is applied on the detected form to verify each hypothesis. Experiments conducted on real world data show the robustness of the algorithm, the speed of the process and its potential in lightweight, economical real- time applications.

1. Introduction

Speed of execution and robustness of the procedure are the two trade-offs between which all image-processing techniques have to decide. Particularly in the case of tracking objects in real time where data is fed at a constant rate and is the subject of compound mathematical models, the speed to complexity ratio must be kept in check so that the application performance in terms of both speed (measured in FPS, frames per second) and accuracy does not deteriorate. Most tracking algorithms are tailored for the laboratory and comprise of complex models and recurring techniques that aim to increase the precision with which objects of interest are detected and tracked. Since these algorithms are processor intensive, they require specialized, high-performance, dedicated hardware in order to achieve a decent frame-rate with a greater degree of robustness.

However this hardware is usually expensive and out of reach of most developing nations for them to introduce it extensively. Unfortunately it is these countries that are worst hit by a lack of security, particularly at public places. Increase in crime rate and terror activities are a concern to authorities particularly in areas that are under-monitored. In such areas surveillance cameras have proven to be quite effective in preventing crimes and capturing offenders.

Still a computerized security system can add to the preventive measure by eliminating human error and reducing response time. Systems like the ADVISOR project [1] couple a detection and tracking subsystem with a behavior analysis subsystem in order to interpret human behavior and make necessary decisions accordingly. Our aim is to provide the lightweight human detection and tracking subsystem that could provide both speed and a fair degree of accuracy on non-dedicated, low-end machines of developing countries. We achieve this by reducing redundancies in existing techniques, employing extensive preprocessing and using simpler models for motion and shape analysis. The behavior subsystem can later be added in order to complete the overall system.

It should be noted that besides its obvious use for security reasons, a plug-able, lightweight human tracking system has applications in many fields particularly in robotic vision.

1.1 Previous Work

Human modeling can be broadly divided into three main categories with increasing model complexity [1].

1) The most basic technique uses Region or Blob based detection. Procedures derived from this method use a model free system [2, 5, 6, 9]. The advantage of using this system is that it is simple, easier to implement and fast. However, it performs no contouring to adjust form and is usually coupled with complex mathematical models such as multi-variate Gaussian models [2] or Kalman filtering [6], in order to provide more accuracy.

2) 2D appearance models are the next step in increasing model complexity and accuracy [1, 3, 8]. The inherent detail in model detection reduces the intricacy of the applied mathematical models. Nevertheless, systems relying on such models employ mathematical techniques similar to those used with blob based detection to further increase tracking robustness [3].

3) The most complicated human modeling procedure is through 3D modeling. A 3 dimensional form is superimposed onto the image. Although this method is largely accurate, it is extremely slow and unsuitable for real-time applications. It is also constrained to specific conditions.

2. Our Approach Overview

In the design of the people tracker, our primary concern was CPU speed and processing power as disk space is cheap and frames can afterwards be discarded. Our aim was to develop a robust real-time tracking application. Most mathematical models used by contemporary tracking systems are complex and require extensive processing. Exponent power calculations and multiple multiplications for each possible set of detected regions is CPU and time expensive. We have forgone this complexity in favor of a simpler linear model relying on color intensities and object dimensions and velocities.

The 2D appearance model approach has been used so as to maintain a fair degree of accuracy and not compromise speed of execution.

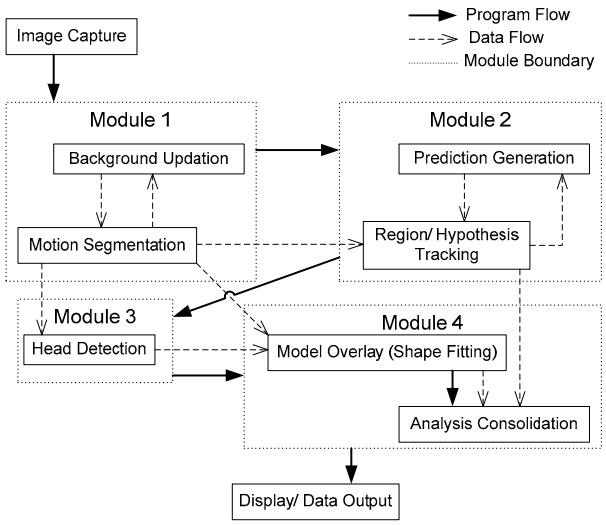

Fig. 1. System diagram showing module interrelations and dependencies.

The basic architecture of Reading People Tracker (RPT) [1] has been employed. However we have skinned the model to get rid of excessive redundant techniques that the system can do without and optimized others in order to help speed frame processing. RPT suggests a distributed architecture with each module utilizing a different processor. However since we have proposed running all the modules on a single workstation, hence they are to be executed linearly one after the other. Parallel processing on a single workstation can be used to improve responsiveness to the user, but removing it results in performance gain for the system. Figure 1 shows the modified program and data flows from that proposed in RPT by Siebel [1].

3. Pre-Processing

All pertinent data about the image is calculated beforehand so as to lower overhead of subsequent processing.

The human shape model used for active shape analysis in the final consolidation module consists of a pre-computed B-Spline. It approximates the form taken by a human being in an upright position (stationary or mobile). The B-Spline model is superimposed on the detected object according to the head position so that the two head centers align. The size of the shape is then adjusted relative to the bounding box calculated in Siebel's head detection procedure [1]. Next a proximity analysis of the shape plot pixels is performed to determine whether they lie on or near object boundary points. At least 50% of the pixels must lie in close proximity to the boundary points for the object to successfully qualify as a human being. This percentage (representing accurate plot of shape pixels) could be adjusted to produce the desired level of accuracy. Whereas systems using shape analysis [1, 5] usually compute models from boundary points for every significant blob in the frame, this technique has the advantage of calculating the basic shape model only once (i.e. at system startup). This produces a significant boost in performance.

4. Illumination-Invariant Detection

All systems employing computer vision are susceptible to discrepancies caused by changes in the external environment particularly luminance. Changing light intensities have a profound effect on techniques involving color information. Lighting changes can either be gradual or sudden. Gradual changes are brought about by time of day and are most influential in outdoor environment image analysis. However in well-lit indoor surroundings, these changes have a confined effect causing only a bounded alteration in the color tone. This effect can be diminished by performing an update of the static pixels of a periodic frame within the background image.

Sudden changes in luminance on the other hand, are caused by flashes of light such as glare from a source or reflection into the camera. The frames captured during such an event can greatly disturb readings as they interfere with object recognition in the tracking module. This exposes a potential weakness of the system whereby the camera is blinded by light shone directly towards the lens. We have implemented the following method in order to determine and remove the defective (bad) frames before they corrupt the system. The method also allows the system to recover from the loss of these frames.

4.1. Bad Frame Elimination

This procedure is used to retain flow of the system in the presence of bad frames. The effect of these frames if left uncurbed is propagated throughout the system clouding both the tracker and the head detection modules. This results in loss of information on detected objects causing the system to make faulty hypothesis. This problem has been tackled by determining and eliminating the bad frames at the root, thus preventing defective data from circulating within the system. The placement of this vital step also saves time.

The process takes place at the start of the first module (the Object Detection Module). The system calculates a simple ratio of the number of pixels that are dynamic to the total number of pixels of the frame. Each thresholded image is defined by the following equation:

$$H(I) = H(B) + H(W) \quad (1)$$

where $H(x) = \sum^{m} \sum^{n} x$: m and n represent dimensions, and B and W are stationary and moving pixels. The ratio can then be represented as:

$$r = H(W) / H(I) \quad (2)$$

The outcome of this ratio 'r' lies between 0 and 1. If this ratio is outside a predefined (and pre-tested) range, the image is discarded. Further processing on the frame is stopped and the module returns. It is then called again for the succeeding frame. The system executes as if the defective frame never occurred.

The calculated ratio is basically an indication of the number of pixels that are white in the thresholded image. A high ratio relates to a higher proportion of white pixels in the detected image. The white pixels are the dynamic pixels that form part of a moving object. Generally, this ratio is minute tending towards 0. This means that the number of moving pixels is small as compared to the total number of pixels.

Following a sudden change of luminance, pixels in the foreground image show a marked deviation from corresponding pixels in the background image. Hence image differencing and thresholding results in a ratio that has more white pixels as compared to black ones. The thresholded image thus shows objects where there are none and results in the ratio exceeding the limit.

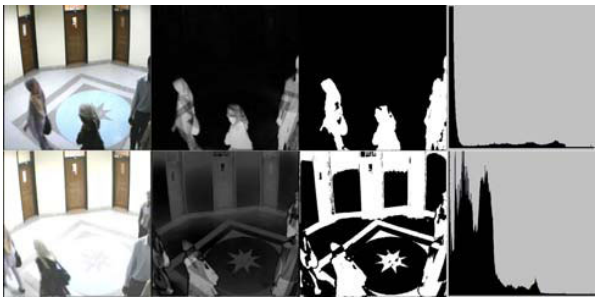

Fig. 2. Comparison of data obtained from coherent (top) and defective (bottom) frames. From left to right, the line contains original frame, differenced image, thresholded image and image histogram. (Histogram axes are not to scale)

This information can be easily obtained and inferred from an analysis of the image histogram. These essentially map the pixel intensity with the frequency of occurrence. If a higher proportion of pixels occur above the threshold intensity, then the image is defective. Figure 2 shows an image histogram of a defective frame (bottom right). Notice the dispersed nature as opposed to the coherent frame histogram.

5. Object Tracking

Correlation between blobs in previous and new frames is established by means of a linear motion model. Predictions of position and displacement are made by using accumulated data from previous frames. In addition, predictions of the dimensions of the new blob are also made. Dimension predictions take into account the position of the object and the camera in a 3-Dimensional world. Thus when an object is detected to be moving away from the camera, its size will be predicted to diminish whereas an approach towards the camera results in a reinforcement of the hypothetical dimensions. However it should be noted that no camera calibration is required in this process as the predictions are made by models relying on differences between previous frames. The position of the center and the dimensions are predicted in terms of pixels. This makes the system more dynamic with regard to input, as it eliminates the need to calibrate and train the system.

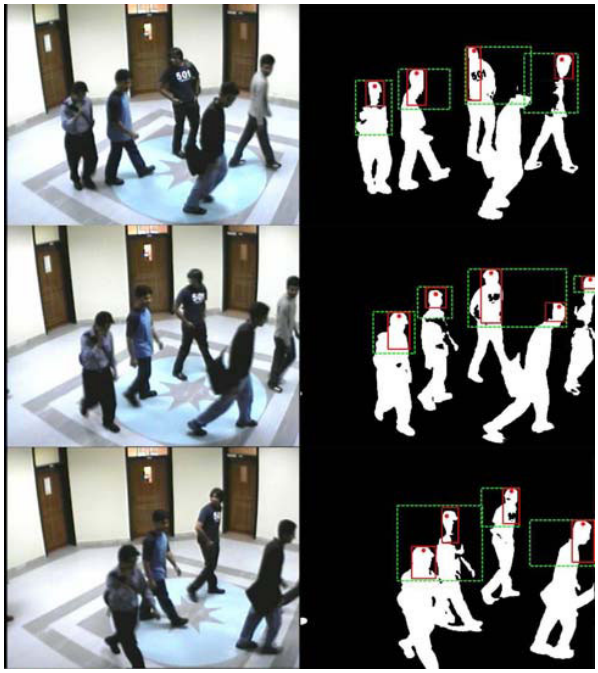

Fig. 3. Results of the head detection algorithm. The red rectangles in the thresholded (right) images represent the detected head whereas the green rectangles demarcate the search area for the head.

Tracking is based on the transition between the objects of separate frames connected by a mapping. A mapping is a correlation between 2 objects detected in different frames. On first entry into the image, an object is initially predicted to

remain stationary; however with subsequent frames, the predicted model for the object refines allowing more accurate insights into future object location and size. The stationary object case is taken in order to prevent any inaccurate mapping that may arise due to erroneous predictions. System startup also follows a similar procedure in which all objects within the image are new and are yet to be assigned mappings. All new objects are hence eliminated from assignment of mappings.

Siebel's equation [1] for determining closest match for an object from among the set of different hypothesis has been used. The equation, as shown below, calculates a value for the absolute weighted difference in the object's position and dimension from the prediction.

$$\delta(\text{obj, pred}) = \alpha_1\Delta_x + \alpha_2\Delta_y + \alpha_3\Delta_w + \alpha_4\Delta_h \qquad (3)$$

Δ_x and Δ_y denote the absolute positional difference in x and y coordinates of the center whereas Δ_w and Δ_h correspond to the absolute differences in width and height respectively. The equation shows that the calculated value is a function of the object detected in the new frame and prediction of an object in the old frame.

The smallest value obtained represents the least error in the object's next frame prediction. This means that the respective prediction is accurate and hence a mapping can be created between the owner of the prediction (object in old frame) and the object detected in the new frame. If the value of the error is substantial, then no precise prediction for the object exists. The mapping of this object will then be deferred to after other objects have been matched by the motion model. Finally the tracked outcome of the unmapped object will be determined by average color intensity matching (if enough unassigned objects are present in the new frame) or/and by region splitting and merging procedure. If a suitable match is still not found, then the object is eliminated as having left the camera's field of view.

Occlusion from objects coming in the way of others, and coalescing of distinct objects has been accomplished by employing region splitting and merging technique [1, 3, 5]. In this procedure, objects in the new and preceding frames are linked by ownership. This is essentially the same as mapping. Each object is the owner of its mapping in the succeeding frame. Ownership of a combined object is shared by all the objects of the preceding frame that have joined together. The system keeps track of these objects until finally they segregate into their component objects. Here average color intensity matching is used to determine which object has departed from the combined Mega object. Ownership in the Mega object is reduced as the component object regains its mapping or ownership.

Average Color Intensity is an approximate value for the chromaticity of an object. It is a number between 0 and 2^n-1 where n represents the number of bits making up one component of a projected color. In ordinary conditions of white light, average color intensities for an object has been found to vary only slightly from frame to frame. Even in conditions of an overlying shadow, the basic shade changes (becomes grave) by a very small amount. Hence average color intensities can be used for a robust 2nd order tracking criterion. In addition, matching with average color intensities is executed within a reduced subset of hypothesis. This diminishes chances of a correlation to another object with similar coloration.

Figure 3 shows the results of the head detection algorithm applied within module 3. Figure 4 shows the output for the same frame subsequence. In the first frame, 2 persons (cyan) are merged. Since the shape model cannot be applied onto the merged blob, it cannot be verified as a human. When the blob splits into its components (in the third frame), the shape model fits onto the separate individuals and confirms the hypothesis. Figure 5 demonstrates more tracking results of the system.

Fig. 4. System Output for the frames in figure 3.

Fig. 5. More hypothesis verification results.

6. Results and Conclusion

We have measured the performance of the application on a test machine that contains a 1.7 GHz Celeron processor with MS-Windows XP SP2 installed. The system is not dedicated and all visual effects have been retained. A system with the above specifications is easily available and is affordable for mass usage by developing countries.

The application performance has been measured by providing a unit that calculates the number of frames that have been processed and the time elapsed so far. The FPS is then calculated by division and extrapolation. Multiple runs of the application have been conducted on data consisting of at least 500 sequential frames. The data set used itself is from 2 different sources and consists of 5 different scenarios.

The frame rate is particularly dependent on the nature and amount of data present within the images. It is also highly reliant on the nature and number of other processes running on the test machine since it is not a dedicated system. For our experiments we assume minimum processes on the test machine (for low CPU and memory utilization) and a random data set. Tests show that the frame rate achieved is between 14 and 23 FPS. The average frame rate is however more erratic and understandably varies between data sets.

Qualitative analysis of the tracking module has been accomplished in accordance with the evaluation performance metrics set out in Keifer[12]. The techniques outlined in [12] suffer from some fundamental problems. Firstly, these techniques are focused towards moving object/pixel detection which is accomplished in the first module. The process (differencing and thresholding) used within this module, is essentially the same as used in RPT and other people trackers and is very dependent on external parameters set during the application calibration such as the value of the threshold and the size of frame. The evaluation techniques fail to give importance to the other modules in determining the quality of the tracked output. The techniques also fail to recognize the inherent nature of procedures involving B-Spline approximation for person approximating in which the system tries to exact a shape for the human subject so that it covers most of the moving pixels. Another problem arises in many ratios where overlapping of objects can result in limits being exceeded.

However in the absence of better quality evaluation criteria, these metrics have been employed. Table 1 shows the results of a randomly chosen video sub-sequence with frame size 640x480 that contains between 1 and 5 people. ξ is fixed at 0.75. Table 2 shows the overall results of different ratios on the video sequence. All calculations have been done with the aid of MATLAB.

Ground Truth Objects refer to the real world objects and Ground Truth Pixels are the actual pixels of these objects. True Positives are those pixels that are successfully identified as ground truth pixels. False Positives are pixels that are mistakenly classified as ground truth pixels. False Negatives are all those pixels which are not detected, although labeled in ground truth. Precision ratios measures how well the algorithm minimizes false alarms whereas the Recall ratios estimate how well the algorithm output covers the ground truth. The calculated ratios of Table 2 indicate a high precision for the detected objects and a satisfactory recall value. This means that the algorithm aptly covers only the ground truth objects. However a low fragment value indicates that the detected objects are usually split up into more than one part.

This paper presents a minimal model that should serve as the basis of monocular indoor human detection and tracking frameworks. Experimental results show that the technique suggested is sufficiently accurate and efficient for use in real-time surveillance and other applications. The implementation of the human detection and tracking framework and estimations of the quality and speed of the output have led to the conclusion that the current complexities in most algorithms produces less gain in terms of tracking robustness (quality) at the cost of speed of execution.

Frame #	Number of Objects	True Positives	False Positives	False Negatives	Pixel Based Precision	Pixel Based Recall	Average Object Fragmentation	Object Based Precision	Object Based Recall	Localized Object Based Precision	Localized Object Based Recall
74	5	8251	1178	2386	0.88	0.78	0.90	1.09	0.78	5.00	4.00
75	5	8081	542	1392	0.94	0.85	1.47	1.17	0.85	5.00	5.00
76	5	7975	496	1503	0.94	0.84	1.13	1.18	0.84	5.00	4.00
77	5	7454	286	900	0.96	0.89	1.00	0.97	0.89	5.00	5.00
78	4	6261	1067	699	0.85	0.90	1.62	1.15	0.90	4.00	3.00
79	4	6667	824	827	0.89	0.89	1.62	1.19	0.89	4.00	4.00
81	4	6272	276	1072	0.96	0.85	3.37	1.92	0.86	4.00	3.00
83	4	5399	417	1087	0.93	0.83	1.16	1.87	0.83	4.00	3.00
85	3	2759	361	1164	0.88	0.70	1.93	2.74	0.67	3.00	1.00
87	3	3353	293	1078	0.92	0.76	1.93	2.77	0.76	3.00	2.00
89	3	3493	195	733	0.95	0.83	1.93	2.86	0.84	3.00	2.00
91	2	2438	93	718	0.96	0.77	1.00	1.93	0.79	2.00	1.00
94	1	937	21	683	0.98	0.58	1.00	0.00	0.58	1.00	0.00
96	1	580	6	756	0.99	0.43	1.00	0.00	0.43	1.00	0.00
99	2	2202	67	1019	0.97	0.68	0.59	0.00	0.68	2.00	0.00

Table 1. Precision and Recall ratio calculations

Quality Metric	Value
Overall Precision (Pixel Based)	0.9218
Overall Recall (Pixel Based)	0.8183
Overall Fragmentation	0.4243
Overall Precision (Object Based)	1.4035
Overall Recall (Object Based)	0.8131
Overall Local Object Precision	1.6452
Overall Local Object Recall	0.7255

Table 2. Overall Ratios for the video sequence

Future work could be extended towards increasing the robustness and speed of the overall process. Techniques such as the aging factor of Ant Colony algorithm could be incorporated within the system to filter out transient glitches by placing more emphasis on those hypotheses that have recently been verified as true. Additionally there is a need for better standardized quality metrics for human (or hypothesis) detection that can give a realistic and comparable idea of the accuracy of the system and include all aspects of the detection and tracking procedure.

References

[1] N.T. Siebel, "Design and Implementation of People Tracking Algorithms for Visual Surveillance Applications", *Ph.D. thesis*, Department of Computer Science, University of Reading, Reading, UK, 2003.

[2] Q. Cai and J. Aggarwal, "Tracking human motion using multiple cameras", *In Proc. ICPR*, Vienna, 1996

[3] T. Zhao, R. Nevatia, and F. Lv. "Segmentation and tracking of multiple humans in complex situations". *In Proc. of USC Computer Vision*, 2001

[4] N.T. Siebel and S.J. Maybank, "The Application of Colour Filtering to Real-Time Person Tracking", *Proceedings of the 2nd European Workshop on Advanced Video-Based Surveillance Systems*, Kingston upon Thames, UK, 2002

[5] T.J. Ellis and M. Xu, "Object detection and tracking in an open and dynamic world", In Proc. *2nd IEEE International Workshop on Performance Evaluation on Tracking and Surveillance,* Kauai, Hawaii, 2001

[6] O. Masoud and N. Papanikolopoulos, "A robust real-time multi-level model-based pedestrian tracking system", *In Proc. ITS America Seventh Annual Meeting*, Minneapolis, June 1997

[7] Y. Song, X. Feng, and P. Perona, "Towards detection of human motion", *In Proc. IEEE Conference on Computer Vision and Pattern Recognition*, *Vol. 1*, Hawaii, 2000

[8] A. M. Baumberg. "Learning Deformable Models for Tracking Human Motion",. *PhD thesis,* School of Computer Studies, University of Leeds, Leeds, UK, October 1995.

[9] I. Haritaoglu, D. Harwood and L. S Davis. W4: Real-time surveillance of people and their actions, *"In Proc. IEEE Transactions on Pattern Analysis and Machine Intelligence*", August 2000.

[10] D. M. Gavrila and L. S. Davis, "Tracking of humans in action: A 3-D model based approach", *In Proc. ARPA Image Understanding Workshop*, Palm Springs, USA, 1996.

[11] H. Sidenbladh. "Probabilistic Tracking and Reconstruction of 3D Human Motion in Monocular Video Sequences", *PhD thesis*, Institutionen for Numerisk analys och datalogi, Kungliga Tekniska Högskolan (KTH), Stockholm, Sweden, November 2001.

[12] C. Keifer. "Qualitative and Quantitative Evaluation of the Reading People Tracker", *Diploma Thesis*, Eidgenössische Technische Hochshule (ETH), Zürich, Sweden, March 2004.