A Real Time Face Tracking System using Rank Deficient Face Detection and Motion Estimation

Vidit Saxena, Sarthak Grover, Sachin Joshi
{vidituec,srthkuec,jsiituec}@iitr.ernet.in
Indian Institute of Technology, Roorkee

***Abstract*-In this paper, we present a novel real-time face tracking system using rank deficient face detection. Real Time computer vision systems need to be fast and efficient to be of practical application. Moreover, the system should be adaptable to diverse real life situations with varied availability of resources. We detect human face in an input grayscale image using a Rank Deficient Face Detection algorithm, which has been shown to be 4 to 6 times faster than unconstrained reduced set systems. The algorithm works on grayscale images and so easily adapts to change in ambient conditions like lighting conditions and different people, and is compatible with both color and grayscale cameras. One major problem in face tracking system used in environments with multiple candidate faces is which face to track and how to keep tracking a heuristically selected face without actually involving face authentication procedures. We have developed a highly flexible yet robust algorithm to deal with these problems. In case of multiple faces in the frame, the most prominent face is identified and automatically selected for tracking, and webcam focused on the desired face. Motion estimation and compensation is then incorporated to ensure robust tracking, to minimize false detections, and for persistent tracking of the desired face. The system showed a robust 10 fps and a false detection rate of 0.6% on a 1.4 GHz Pentium IV workstation using an Intel CS110 webcam. The principle used in this system can be applied to real time tracking of other objects, viz. cars, the human hand, etc by using an appropriate training set. It can also be extended to provide face authentication and subsequent tracking of desired face.**

Keywords: Support Vector Machine (SVM), reduced set vectors, motion vectors, motion compensation.

1. INTRODUCTION

Rank Deficient face detection can be implemented in a visual input system capable of horizontal and vertical motion to develop a novel method of tracking human faces in real time. Real time face tracking is of primary importance in various applications, viz. video surveillance, Human-Computer Interfaces (HCIs), teleconferencing, teleteaching, and unmanned video capture and recording in diverse environments. Some systems incorporate face tracking on the basis of color attributes (e.g. color histograms, etc), [2,3,4]. These systems, however, do not distinguish faces from non faces and so are prone to error and false detections. In such systems, face detection algorithms cannot be used as they take up huge computation power and time and so are unworthy of real time application. Even in existing systems using grayscale images [5,6], the face detection algorithms applied are quite precise and entail large computations, resulting in loss of real time performance. Systems using Eigenfaces, Neural Networks have been developed but they vary with the affine variations of facial image [7]. The novel method of tracking faces in [8], wavelet jet bunch graph approach, tracks faces at 1 fps and so is unsuitable to real time. Wolf Kienzle, et al showed in [1] that face detection using patch classifiers can be speeded up by upto 4 to 6 times using rank deficient face detection. We use this method of face detection to develop a tracking system that operates at 10 fps and shows a false detection rate of approximately 0.60%.

Here we implement the face detection algorithm described in [1] in a face tracking system. The advantages of this algorithm are manifold:

1. Image processing and subsequent face detection is performed on grayscale images, thereby increasing the scope of applications.
2. The detection is robust to changes in ambient conditions as compared to color based systems.
3. Because of the use of rank deficient images for face detection, the algorithm is fast and thus suitable for real time application.

By exploiting these advantages, we have developed this system for face tracking and tested it in various practical situations with varying complexities to obtain an excellent rate of success. For the motion of camera, we have mounted it on a computer controlled motion system capable of moving it in circular motion in horizontal and vertical plane. The motion being circular, components of the magnitude of motion vector along X and Y axes have to be squared to estimate the duration of motion in corresponding directions.

In section 2, we will describe the process of face detection on the intensity images, determination of motion vectors and the algorithms governing the tracking motion. Section 3 describes the hardware setup of the image acquisition and face tracking system. In section 4, we will discuss the results obtained, the possible limitations of the system, and conclude with the scope of applications of the system.

2.1 FACE DETECTION USING RANK DEFICIENT REDUCED SETS

The input obtained from the visual input system is converted to a grayscale image for detection, thereby increasing the speed of detection suitably for our time-critical application. Support Vector Machines (SVMs) have been widely used in robust object detection systems. However, they are seldom used in real time applications because of the computationally expensive decision functions. Object detection can be sped up by reducing the number of expansion points [9,10]. Particularly in [10], Burges introduced the concept of Reduced Set Vectors (RSVs), which improved speed by a factor of 10 to 30 as reported in [10,11,12]. The use of SVM has been shown to outperform Haar wavelets and gradients for face detection in [13].

In [1], the authors have proposed a method of speeding up SVM computations by constraining the RSVs to a special structure, and evaluating them via separate convolutions. This method decreases the complexity of RSV evaluations from O(h.w) to O(r.(h+w)), r being a small number used to tradeoff speed with accuracy of evaluations. The method has been built upon Burge's method of object detection in [10], with the exception that the RSV search space is restricted to the manifold spanned by all image patches that have a fixed, small rank 'r' when viewed as matrices. The rank 'r' is selected by the user and is a measure of the accuracy desired. Selection of a larger 'r' results in more accuracy and thereby lesser false detections. However, it also entails an increase in the computation requirements, and slower speed. The rank has to be selected experimentally depending upon the ambient lighting conditions, background complexity, and the expected level of activity.

Threshold (r)	No. of Faces input	Correct Detections	False Negative
0	2	2	0 %
0	5	5	0 %
0	6	6	0 %
1	8	8	0 %
3	6	6	0 %
3	5	5	0 %
5	5	5	0 %
5	6	6	0 %
5	8	8	0 %
6	8	7	12.5 %
10	2	1	50 %
10	5	3	40 %
10	8	3	62.5 %

Table. 1. False Negative Detections vs Threshold

The rank deficient system of detecting faces showed a false positive ratio of 0.6% for a desired 10 frames per second input. Even in the case of a random false positive, the system is not affected much, because the next detection, if correct, provides a correct estimate of motion vector and thus nullifies the effect of erroneous detection.

Threshold (r)	No. of Faces input	False Detections	False Positive
0	2	0	0 %
0	5	1	20 %
0	6	1	16.67 %
1	8	1	12.5 %
3	6	1	16.67 %
3	5	0	0 %
5	5	0	0 %
5	6	0	0 %
5	8	0	0 %
6	8	0	0 %
10	2	0	0 %
10	5	0	0 %
10	8	0	0 %

Table. 2. False Positive Detections vs Threshold

On applying the face detection algorithm to a data set of 4 images containing 2, 5, 6, and 8 faces, the most optimum results were obtained for a threshold (r) of 5. For a neutral or negative value of threshold, all correct faces were detected but false detections also occurred. As threshold became more negative, error rate due to false positive detections increased. For a positive value of threshold, correct faces were detected and false detections decreased. But increasingly positive values of threshold resulted in false negative detections and at high values number of faces detected decreased drastically.

A disadvantage of using a high positive value of threshold is that higher threshold implies a higher rank (r). We have seen that the complexity of rank deficient matrix systems varies as O(r.(h+w)). Thus a higher r results in accuracy of face detection but a loss in speed of computation. Large values of threshold may be applicable to systems working on still images, but for real-time face detection and tracking system, the time of computation represents an important factor which has to be taken into account. Thus the accuracy and speed must be compromised to get an optimum value which results in least errors and is able to follow faces detected in a laboratory environment satisfactorily.

When a high value of threshold is used for face tracking by the movable camera, the speed of computation is too high and the faces detected move out of the frame before the next image is taken. Due to this time lag, the motion vector for faces in the subsequent frames is very large and eventually the faces detected do not get tracked continuously.

On using a low value of threshold, the computation speed gets increased and thus the camera is easily able to follow

detected faces at an appropriate speed. Though using a low value increases the amount of false detections which occur but the speed of computation makes up for that loss as a face can be detected quickly in the subsequent frame. Thus a lower rank matrix effectively speeds up the program.

2.2 SUBJECT FACE DETERMINATION AND MOTION VECTORS

The grayscale input image used for subject face determination poses 3 basic challenges: (i) It may contain more than one face of varying size (ii) There may be one or more false positives masquerading as human faces (iii) The face detected in frame k might be absent from the input frame k+1 altogether, making it necessary to decide upon a new subject for tracking. Also, while tracking a particular subject, the size of subject face may increase or decrease, and other faces may come in view of input system and be detected as probable candidates for tracking. To accurately track a desired subject face, various face authentication methods may also be used by storing the subject face once identified and using it for subsequent face detections. However, face authentication being a computationally intensive process, it cannot be implemented in a real time system effectively.

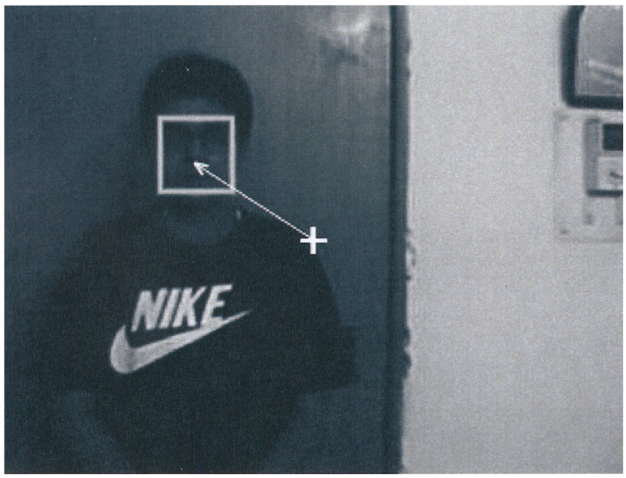

Fig. 1. Motion Vector determination

Therefore, we use a generic algorithm to identify a face for tracking and then 'lock' on to it without actually storing and using the subject face for further face detections. For initial selection of subject face, the first input frame is searched for possible human face candidates, and square estimates of the face location and size are determined. Out of the multiple faces, the face with largest dimensions is selected as subject face. A motion vector for tracking is then constructed by joining the center of the input frame to the center of subject face (Fig 1). Moreover, if there is more than one face of the same largest dimension, the face with smaller motion vector is selected because a smaller motion vector implies that the face is closer to the axis of visual input system. In the rare case of same motion vector magnitudes, the left face in the frame is arbitrarily chosen as the subject face.

2.3 MOTION ESTIMATION AND SUBJECT FACE TRACKING

Once the subject face has been selected, we have to robustly identify the same subject in subsequent frames and estimate the motion vector for its tracking. The motion of the subject relative to the camera can be of three kinds: (a) Parallel to the axis of visual input system, (b) Perpendicular to the axis, or (c) At any arbitrary angle (Fig 2). Due to the practical nature of application, subject motion is mostly be of type (c), but this motion can be broken down into components of the independent (a) and (b) types and analyzed accordingly.

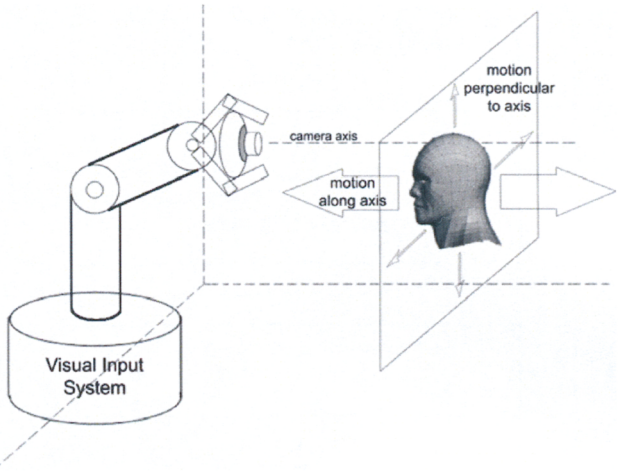

Fig. 2. Components of subject motion parallel and perpendicular to axis

Subject motion parallel to the axis will not have any effect on the motion vector **'M'**; it will only lead to an increase in the dimension 'D' of the subject face (D is the side of square estimated as subject face). Again, motion perpendicular to the axis will have no effect on the dimension of the subject face, but the change in motion vector will reflect subject motion. The following algorithm (Fig 3) is implemented to determine the position of subject face in frame k, given the motion vector $\mathbf{M_{k-1}}$ and dimension D_{k-1} of subject face in frame k-1:

1. Detect the 'n' possible human face candidates in frame k, their dimensions $D_{i,k}$ and construct the motion vectors $\mathbf{M_{i,k}}$ (i = 1 to n).
2. Calculate $P_{i,k} = \alpha (D_{i,k} - D_{k-1}) + \beta |\mathbf{M_{i,k}} - \mathbf{M_{k-1}}|$, for i = 1 to n, where α and β are weighing factors which are determined according to the application environment of the face tracking system. $D_{i,k} - D_{k-1}$ is a directly proportional to the speed of subject parallel to axis and $|\mathbf{M_{i,k}} - \mathbf{M_{k-1}}|$ to the speed perpendicular to axis.
3. If the human traffic is mostly linear and parallel to the axis of visual input system, a higher α/β should be used. However, if the motion of people is expected to

be perpendicular to the axis, a lower value of α/β is required for optimal tracking.

4. Select the face with lowest $P_{i,k}$ as the subject face. Put $\mathbf{M_k} = \mathbf{M_{i,k}}$ and $D_k = D_{i,k}$ for this i, and go back to step 1.
5. In case the value of P_k is greater than a threshold value (determined experimentally), it can be safely concluded that the subject face is no longer present in the frame any more and a new subject should be selected.

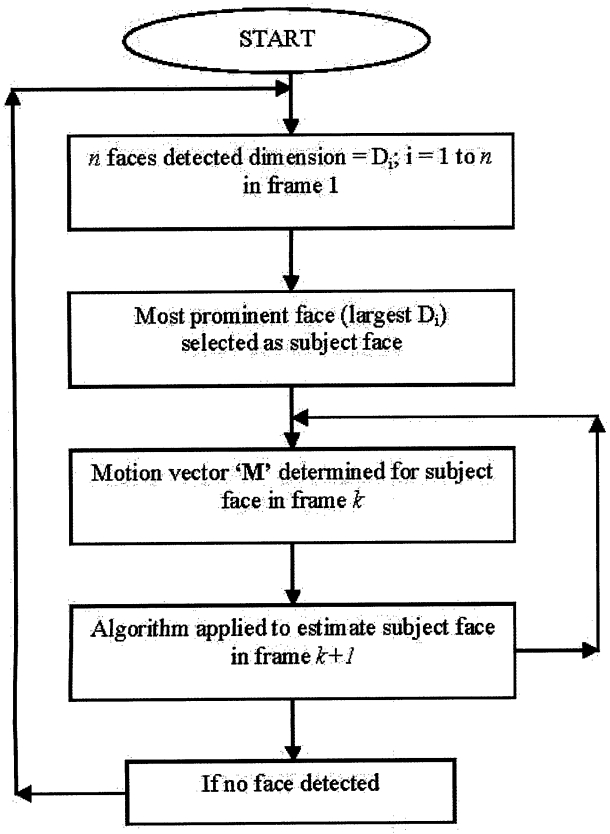

Fig. 3. Flowchart of the applied algorithm.

The motion vector defined above however has a serious flaw: we cannot compare the motion vectors of frame k and k+1 directly, because the motion of camera shifts our axis. Thus we have to use some motion compensation by shifting the origin for frame k+1. The motion of camera is in the direction of motion vector $\mathbf{M_k}$. So, to obtain a motion vector for frame k+1 on the same axes as that of frame k, we just add a fraction of motion vector $\mathbf{M_k}$ to the motion vector $\mathbf{M_{k+1}}$, thus obtaining the compensated motion vectors. This fraction depends on the hardware setup of the system. The algorithm can then be applied as above.

This algorithm is developed making use of the fact that human motion is predictable and non erratic at 10 fps input. This is also the reason that we can reasonably expect P_k to be less than a threshold value. This threshold value varies over different systems, and is directly proportional to the chosen values of α and β. Once the face with the least $P_{i,k}$ is selected as the subject face in frame k, its motion vector can now be used for actual tracking.

3. HARDWARE SETUP

The hardware setup consists of a USB webcam, Intel CS110 mounted on a movable, computer controlled arm (Fig 4). The arm is capable of circular motions in the horizontal plane and the vertical plane. For the tracking motion, the arm can move both horizontally and vertically simultaneously. The arm motion, however, is non linear in the sense that it has to accelerate each time a new motion vector is calculated. To compensate for the non linear motion, the time duration of motion of the arm is made proportional to the square of the magnitude of motion vector.

The input taken was I420_352x288 frames, which were converted to grayscale before analysis by the face detection algorithm. The software has been developed on MATLAB, and arm is controlled using the parallel port of a Pentium IV, 1.4 GHz desktop workstation through MATLAB's data acquisition toolbox. The parallel port is interfaced with the webcam using simple transistor-transistor logic circuits. The transistors used are of high power rating to withstand the high power required to move the robotic arm. Maximum speed of the arm was adjusted to an optimum level to facilitate the 10 fps input rate and allow for relatively less average speed of motion of human subjects.

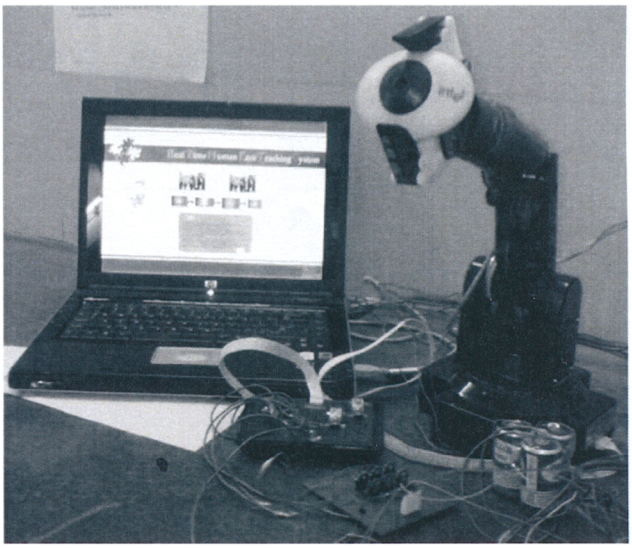

Fig. 4. Hardware setup

4. EXPERIMENTAL RESULTS AND FUTURE WORK

We applied the face tracking system in various common environments and noted the results on varying the values of α and β. In an average laboratory environment, optimum result was obtained for $\alpha = 0.1$ and $\beta = 0.15$. The value of threshold of P_k using these values of α and β was found to be 50 units. The false positive rate on taking rank 'r' of the rank deficient face detection algorithm (described in 2.1) as 5 was found to be approximately 0.6%.

The face detection algorithm was able to robustly track subjects moving at moderate speed (Fig 5(a), Fig 5(b)). The algorithm was initially tested independent of the physically tracking camera system, and showed robust tracking of human subjects during motion parallel to axis (Fig 5(a)) and motion perpendicular to axis (Fig 5(b)) of the stationary camera. The algorithm was then applied on the visual input system capable of motion in horizontal and vertical directions, and rigorous testing yielded encouraging results (Fig 6). We were also able to easily better the tracking for fast moving subjects by decreasing the value of 'r' to 3 and lowering the α/β ratio to 1, though the false positive face detections increased.

Fig. 5(a). Tracking of subject when motion is parallel to the axis of camera

Fig. 5(b). Tracking of subject when motion is perpendicular to the axis of camera

The real time face detection system developed has 2 major drawbacks: (i) A rate of 10 fps, though good for moderate motion of subject, is unsuitable for real life applications with fast moving subjects, Our system has provided very good results for tracking with a false positive rate of only 0.6%. The probable shortcomings in the system and problems in different operational conditions have been identified and can be easily handled so that they do not limit its scope of application.

REFERENCES

1. Wolf Kienzle, Gökhan Bakır, Matthias Franz and Bernhard Schölkop. Face Detection -Efficient and Rank Deficient.
2. S. Bircheld. Elliptical head tracking using intensity gradients and color histograms. In *IEEE Conf. Computer Vision and Pattern Recognition*, CVPR, pages 232-237, 1998.
3. V. Kruger, R. Herpers, K. Daniilidis, and G. Sommer. Teleconferencing using an attentive camera system. In *Int. Conf. on Audio- and Video-based Biometric Person Authentication,* pages 142-147, 1999.
4. Y. Raja, J. McKenna, and S. Gong. Tracking and segmenting people in varying lighting conditions using color. In *Int. Conf. on Automatic Face- and Gesture- Recognition*, pages 228-233, Nara, Japan, April 14-16, 1998.
5. F. d.l. Torre, S. Gong, and S. McKenna. Viewbased adaptive affine tracking. In Proc. *Fifth European Conference on Computer Vision*, volume 1, pages 828-824, Freiburg, Germany, June 1-5, 1998.
6. M. Isard and A. Blake. Condensation conditional density propagation for visual tracking. *International Journal of Computer Vision*, 1998.
7. S. McKenna and S. Gong. Recognizing moving faces. In H. Wechsler et.al., editor, Face Recognition. From *Theory to Applications, NATO ASI Series*, pages 578-588. Springer, 1998.
8. L. Wiskott, J. M. Fellous, N. Kruger, and C. v. d. Malsburg. Face recognition by elastic bunch graph matching. *IEEE Trans. Pattern Analysis and Machine Intelligence,* 19(7):775-779, July 1997.
9. E. Osuna and F. Girosi. Reducing the run-time complexity in support vector machines. In B. Sch¨olkopf, C. J. C. Burges, and A. J. Smola, editors, *Advances in Kernel Methods—Support Vector Learning*, pages 271–284, Cambridge, MA, 1999. MIT Press.
10. C. J. C. Burges. Simplified support vector decision rules. In *International Conference on Machine Learning*, pages 71–77, 1996.
11. C. J. C. Burges and B. Sch¨olkopf. Improving the accuracy and speed of support vector machines. In M. C. Mozer, M. I. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems*, volume 9, page 375. MIT Press, 1997.
12. E. Osuna, R. Freund, and F. Girosi. Training support vector machines: an application to face detection. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 1997.
13. B. Heisele, T. Poggio, and M. Pontil. Face detection in still gray images. AI Memo 1687, MIT, May 2000. CBCL Memo 187.

Fig. 6. Video sequence showing the face tracking of desired subject under various complex motion, introduction of new human subject and partial occlusion. Note that the subject was chosen in first frame by the proposed heuristic of searching for the most prominent face and tracked throughout the rest of the video.