

Partial Swarm Optimization for Minimizing Occlusion Problem during Multi-Face Recognition

M. A. Archana¹, Dinesh Kumar .T², Dr. C. Parthasarathy³

¹Research scholar, SCSVMV University, Kanchipuram, Tamilnadu, India

²Assistant Professor, SCSVMV University, Kanchipuram, Tamilnadu, India

³Assistant Professor-II, SCSVMV University, Kanchipuram, Tamilnadu, India

Abstract: *Visual surveillance in crowded scenes, especially for humans, has recently been one of the most active research topics in machine vision because of its applications such as deter and response to crime, suspicious activities, terrorism or other illegal activities. One of the most important problems in multiple face tracking is the occlusion problem. The occlusion problem can be overcome by using the partial swarm optimization (PSO) as a tracker in addition to the kalman filter. Kalman filter is used for filtering the frames and also for removing the Gaussian noise. For face tracking in a video sequence, various face tracking algorithms have been proposed however, most of them have a difficulty in finding the initial position and size of a face automatically. In this, we present a fast and robust method for fully automatic multiple face detection and tracking. Using a small number of critical rectangle features selected and trained by various algorithms, we can detect the initial position, size and view of a face correctly. Face tracking and detection will be done for images and also for the recorded videos but not for live video systems. We are extending the face tracking system for the live videos also with the help of new tracker and the new version tools.*

Keywords: Partial Swarm Optimization, Video, image processing, Spatial transformations, occlusion, face tracking,

1. Introduction

An image may be defined as a two-dimensional function, $f(x, y)$, where x and y are spatial (plane) coordinates, and the amplitude of f at any pair of Coordinates (x, y) is called the intensity or gray level of the image at that point. When x, y , and the amplitude values of f are all finite, discrete quantities, we call the image a digital image. The field of digital image processing refers to processing digital images by means of a digital computer. Note that a digital image is composed of a finite number of elements, each of which has a particular location and value. These elements are referred to as picture elements, image elements, pels, and pixels.

Pixel is the term most widely used to denote the elements of a digital image. Vision is the most advanced of our senses, so it is not surprising that images play the single most important role in human Perception. However, unlike humans, who are limited to the visual band of the electromagnetic (EM) spectrum, imaging machines cover almost the entire EM spectrum, ranging from gamma to radio waves. They can operate on images generated by sources that humans are not accustomed to associating with images. These include ultrasound, electron microscopy, and computer-generated images. Thus, digital image processing encompasses a wide and varied field of applications. There is no general agreement among authors regarding where image processing stops and other related areas, such as image analysis and computer vision, start sometimes a distinction is made by defining image processing as a discipline in which both the input and output of a process are images. We believe this to be a limiting and somewhat artificial boundary. For example, under this definition, even the trivial task of computing the average intensity of an image (which yields a single number) would not be considered an image processing operation. On the other hand, there are fields such as computer vision whose

ultimate goal is to use computers to emulate human vision, including learning and being able to make inferences and take actions based on visual inputs. This area itself is a branch of artificial intelligence (AI) whose objective is to emulate human intelligence. The field of AI is in its earliest stages of infancy in terms of development, with progress having been much slower than originally anticipated. The area of image analysis (also called image understanding) is in between image processing and the computer vision. There are no clear-cut boundaries in the continuum from image processing at one end to computer vision at the other. However, one useful paradigm is to consider three types of computerized processes in this types we have low-level processes, mid-level processes, and high-level processes.

2. Video & Image Processing

The Video and Image Processing Blackest software is a tool used for the rapid design, prototyping, graphical simulation, and efficient code generation of video processing algorithms. Video and Image Processing Block set blocks can process images or video data. These blocks can import streaming video into the Simulink environment and perform two-dimensional filtering, geometric and frequency transforms, block processing, motion estimation, edge detection and other signal processing algorithms. We can also use the block set in conjunction with Real-Time Workshop to automatically generate embeddable C code for real-time execution. Video and Image Processing Block set blocks support floating-point, integer, and fixed-point data types. To use any data type other than double-precision and single-precision floating point, you must install Simulink Fixed Point software

3. Image Processing Toolbox

Image Processing Toolbox provides a comprehensive set of reference-standard algorithms and graphical tools for image processing, analysis, visualization, and algorithm development. You can perform image enhancement, image deblurring, feature detection, noise reduction, image segmentation, geometric transformations, and image registration. Many toolbox functions are multithreaded to take advantage of multicore and multiprocessor computers. Image Processing Toolbox supports a diverse set of image types, including high dynamic range, gigapixel resolution, embedded ICC profile and tomographic. Graphical tools let you explore an image, examine a region of pixels, adjust the contrast, create contours or histograms, and manipulate regions of interest (ROIs). With toolbox algorithms you can restore degraded images, detect and measure features, analyze shapes and textures, and adjust color balance.

3.1 Key Features

- Image enhancement, filtering, and deblurring
- Image analysis, including segmentation, morphology, feature extraction, and measurement
- Spatial transformations and image registration
- Image transforms, including FFT, DCT, Radon, and fan-beam projection
- Workflows for processing, displaying, and navigating arbitrarily large images
- Modular interactive tools, including ROI selections, histograms, and distance measurements
- ICC color management
- Multidimensional image processing
- Image-sequence and video display
- DICOM import and export

3.2 Analyzing Images

Image Processing Toolbox provides a comprehensive suite of reference-standard algorithms and graphical tools for image analysis tasks such as statistical analysis, feature extraction, and property measurement.

4. Statistical Functions

Statistical functions let you analyze the general characteristics of an image by:

- Computing the mean or standard deviation
- Determining the intensity values along a line segment
- Displaying an image histogram
- Plotting a profile of intensity values

5. Computer Vision System Toolbox

Computer Vision System Toolbox provides algorithms and tools for the design and simulation of computer vision and video processing systems. The toolbox includes algorithms for feature extraction, motion detection, object detection, object tracking, stereo vision, video processing, and video analysis. Tools include video file I/O, video display, drawing graphics, and compositing. Capabilities are provided as MATLAB functions, MATLAB System object,

and Simulink blocks. For rapid prototyping and embedded system design, the system toolbox supports fixed-point arithmetic and C code generation.

5.1 Key Features

- Feature detection, including FAST, Harris, Shi & Tomasi, SURF, and MSER detectors
- Feature extraction and putative feature matching
- Object detection and tracking, including Viola-Jones detection and CAM Shift tracking
- Motion estimation, including block matching, optical flow, and template matching

5.2 Surveillance of Face

Segmentation and tracking of objects, especially humans in video sequences is one of the most important researches for its applications, such as visual surveillance and human-computer interaction. There have been many studies carried out on this subject in the literature, a few of them introduced here in this paper. Isard and Mac Cormack used a multi-blob likelihood function with particle filter and made a Bayesian multi blob tracker. In Zhao and Nevatia used the Markov chain Monte Carlo (MCMC) technique for their multiple human tracking algorithms and proposed other algorithms for multiple objects tracking when different objects are merged or split. Some other approaches used a kind of background subtraction or comparison between the pixel color of frames and learned models of the stationary background. When the tracker uses background subtraction for segmentation, humans may be merged into a single blob. Haritaoglu et al. proposed a method to segment a big blob into multiple humans.

For human tracking, body details are not needed, therefore body models such as multiple rectangles could be used. In most of the mentioned activities, occlusion is not complete. In other words, their algorithms work when they do not have any full occlusion conditions for several frames. But this paper proposed one of the recent particle swarm optimization (PSO) in as a tracker, in addition to the Kalman filter to solve the problem of some situations with severe occlusion in several frames.

6. Proposed System

The project features a web camera, installed in house premises, which is operated by an advanced information and communication system based on Internet and mobile phone monitoring. The web camera is connected within house and software module is designed to be devoted to analyze the signals which are generated by it. The camera detects falling of a person in front of the camera dimensions

In this mentioned method has been extended to track objects in difficult situations. In the last section a constant and limited number of particles with a constant covariance matrix and certain mean vector which was last state of the object, were released into multidimensional state space and then the particle were updated. Hence, to update the particles, object should not have any full occlusion and the tracker should have information regarding the object.

In this method, covariance matrix was constant and dependent on problem conditions and the mean vector was often about the last object state. However, when extreme or full occlusion occurred, it was necessary to find a way for robust tracking because the tracker did not have any information about the object. Thus, the state vector in the last frame is not accessible and the tracker should search a larger space because, we are not informed about the next object state exactly. Before presenting the solution, at first, the tracker should discover the occlusion time. So suppose that Y is the object feature vector with j Elements in which these features can be pixels color. For each object and each feature a Gaussian membership function has been provided with a mean \bar{y}_j^k and standard deviation σ_j^k produced with feature history in previous frames for the object k .

$$\mu_j^k(y_j) = \begin{cases} a_j^k \exp \left[-\left(\frac{y_j - \bar{y}_j^k}{\sigma_j^k} \right)^2 \right] & k = 1 \dots M \\ 1 - \max_{k=1}^M (\mu_j^k(y_j)) & k = 0 \end{cases}$$

Tracking of video objects is one of the major issues in video surveillance systems. To analyze the behavior of a specific person, the target's position needs to be correctly located in consecutive frames. Several approaches have been proposed to deal with the tracking problems.

7. Tracking

The following briefly shows the three major categories of tracking methods:

- **Point tracking:** The target is expressed as a point in the frame, and the previous target state is utilized to make the association between targets and points. Kalman filter and particle filter are widely used in this category.
- **Kernel tracking:** Targets are tracked by computing the motion of the kernels which represent the appearance of the targets. Mean shift tracker is a kind of kernel tracking.
- **Silhouette tracking:** Given the target model, the target is tracked by matching the contour region in each frame.
- **Tracking by using PSO:**

As mentioned above, our tracker is PSO which is one of the newest methods in object tracking. It is initially introduced in. Suppose that $x =$

$$\{x_{t,k}^{i,n}, i = 1, \dots, N, k = 1, \dots, M\}$$

is set of particles that introduces states and $x_{t,k}^{i,n}$ represents the i th particle for the k th object in the t th frame on the n th iteration. N is the number of particles and M is the number of objects. Every particle is a vector with a couple of elements that represents a situation of template.

For example it can be

$$x = (x, y, w_x, w_y, \theta).$$

Respect to X ,

$$O = \{o_{t,k}^{i,n}, i = 1, \dots, N, k = 1, \dots, M\}$$

is a set of observations that are reached by particles set in the frame. In this method each particle and each observation is updated frequently. Then after number of iterations, all of the particles are changed to the same vector, which is the final goal and it represents the new place of the object. We introduce p^i as the best state which appeared during

iterations for the i th particle and also 'g' as the best state that all of the particles had in their iterations. The particles can be updated in each stage of the iteration as follows:

$$p_{t,k}^i = \begin{cases} x_{t,k}^{i,n}, & \text{if } f(x_{t,k}^{i,n}) > f(p_{t,k}^i) \\ p_{t,k}^i, & \text{else} \end{cases}$$

$$g_{t,k} = \arg \max_{p_{t,k}^i} f(p_{t,k}^i)$$

Where function returns the similarity between observation corresponding to the particle and object appearance model. It is noteworthy to mention that for every object and each frame, a constant number of particles are released with the Gaussian distribution and constant covariance matrix on the last object's state into the state space

8. Results

We evaluate the performance of the proposed methodology Partial Swarm Optimization is used to minimize the occlusion problem. The experiment will test the real-time characteristic of the face tracking system. The environment includes outdoor, office room and a dark room with controllable lighting. A System objects to estimate direction and speed of object motion. The program for multi face tracking is written and executed using MATLAB. A sample live video frame is illustrated in the figure.



Figure 1: Shows the Face Recognition with Eliminated Occlusion Problem

9. Conclusion

Based on the obtained results from proposed method, it is possible to make the following conclusions Occlusion time determination when the face has a full occlusion with the background. Face tracking when the tracker does not have any information about faces for the number of frames by the Kalman filter. Considering the probability of changing the pathway when the face is hidden behind the background by increasing the search space without need for re-detection.

10. Future Work

In future, we shall investigate more efficient optimization procedures of the constrained clustering and matching

problems and incorporating the simultaneous face clustering and linking into an overall system for video summarization. Apart from this, there are two critical problems; one is light condition changing and the number of clusters in the k-means clustering, which are not solved here. To cope with the lighting condition changes, our tracking system must be able to Quickly update its color model. Another problem is that we may need to guess the proper number of clusters in the absence of prior information. A formal approach to this problem is to devise some measure of goodness of fit that expresses how well a given k-cluster description matches the sample data. These two problems are our further works.

References

- [1] A. F. Abate, M. Nappi, D. Riccio, and G. Sabatino, "2D and 3D face recognition: a survey," *Pattern Recognition Letter*, vol. 28, no. 14, pp. 1885–1906, 2007.
- [2] A. Mittal and L. Davis, "M2tracker: A multi-view approach to segmenting and tracking people in a cluttered scene using region-based stereo," *Int'l J. of Computer Vision (IJCV)*, 2003.
- [3] A. K. Jain, R. P. W. Duin, and J. C. Mao, "Statistical pattern recognition: a review," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 1, pp. 4–37, 2000.
- [4] D.A.Ross, J.Lim, R-S.Lin and M-H.Yang, "Incremental Learning for Robust Visual Tracking," in *Proc. International Journal of Computer Vision*, vol. 77, no. 1-3, pp. 125- 141, Aug 2007.
- [5] Greg Welch and Gary Bishop, An Introduction to the Kalman Filter, TR-95-041, Department of Computer Science, University of North Carolina at Chapel Hill, updated in 2002.
- [6] J. Berclaz, F. Fleuret, and P. Fua, "Robust people tracking with global trajectory optimization," in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2006.
- [7] J. Black, T. Ellis, and P. Rosin, "Multi view image surveillance and tracking," in *IEEE Workshop on Motion and Video Computing*, 2002.
- [8] J. Kang, I. Cohen, and G. Medioni, "Tracking people in crowded scenes across multiple cameras," in *Asian Conf. on Computer Vision*, 2004.
- [9] M. Grgic, and K. Delac, "Face recognition homepage." [online] Available: <http://www.face-rec.org/general-info>.
- [10] O. Javed, Z. Rasheed, O. Alatas, and M. Shah, "Knight: A real time surveillance system for multiple overlapping and nonoverlapping cameras," in *4th Int'l Conf. on Multimedia and Expo*, 2003.
- [11] R. Chellappa, C. L. Wilson, and S. Sirohey, "Human and machine recognition of faces: a survey," *Proc. IEEE*, vol. 83, no. 5, pp. 705-740, 1995.
- [12] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: a literature survey," Technical Report CAR-TR-948, Center for Automation Research, University of Maryland (2002).
- [13] X.Zhang, W.Hu, Wei Qu, and S.Maybank, "Multiple Object Tracking Via Species-Based Particle Swarm Optimization," in *Proc. IEEE Trans. Circuits and Systems for Video Technology*, vol. 20, no. 11, pp. 1590-1602, Nov 2010.

Author Profile



M. A. Archana received the B.E. degree in Electronics and Communication Engineering from Kanchi pallavan Engineering College, Anna University, Chennai, Tamil Nadu, India and the M.E. degree in Applied Electronics from C.Abdul Hakeem College of Engineering, Anna University, Chennai, Tamil Nadu, India in 2007 and 2011. Presently she is a Research Scholar in Electronics and Communication Engineering at SCSVMV University, Kanchipuram, Tamilnadu, India.



T. Dineshkumar received the B.E. degree in Electronics and Communication Engineering from Ranipettai Engineering College, Anna University, Chennai, Tamil Nadu, India and the M.E. Communication Systems from Ranipettai Engineering College, Anna University, Chennai, Tamil Nadu, India in 2008 and 2010. Presently he is working as Assistant Professor in Electronics and Communication Engineering at SCSVMV University, Kanchipuram, Tamil Nadu, India.



Dr. C.Parthasarathy received the Ph.D degree in Information and Technology from SCSVMV University in 2012. Presently he is working as Assistant Professor in Information Technology at SCSVMV University, Kanchipuram, Tamilnadu, India.