Multi-constraints Face Detect-Track System

Hazar Mliki, Mohamed Ali Hammami, Hanêne Ben-Abdallah

▶ To cite this version:

Hazar Mliki, Mohamed Ali Hammami, Hanêne Ben-Abdallah. Multi-constraints Face Detect-Track System. 11th International Conference on Computer Information Systems and Industrial Management (CISIM), Sep 2012, Venice, Italy. pp.224-235, 10.1007/978-3-642-33260-9_19 . hal-01551727

Multi-constraints face detect-track system

Mliki Hazar[1] Hammami Mohamed[2] and Ben-Abdallah Hanêne[1]

[1] MIRACL-FSEG, University of Sfax, 3018 Sfax, Tunisia
mliki.hazar@gmail.com
hanene.benabdallah@fsegs.rnu.tn
[2] MIRACL-FS, University of Sfax, 3018 Sfax, Tunisia
mohamed.hammami@fss.rnu.tn

Abstract. This paper presents a new system to achieve face detection and tracking in video sequences. We have performed a combination between detection and tracking modules to overcome the different challenging problems that can occur while detecting or tracking faces. Our proposed system is composed of two modules: Face detection module and face tracking module. In the face detection module, we have used skin color and motion information to extract regions of interest and cut off false positive face. This filtering step has enhanced the next face tracking processing step, as it helps to avoid tracking false positive faces. Regarding tracking module, we have used face detection results to keep the face tracker updated. In order to carry on tracking face we have used particle filter technique which was adapted to track multiple faces. Moreover, each tracked face was described by a defined state: tracked, occluded, entered, left or stopped. The performance of our detect-track system was evaluated using several experiments. This evaluation proved the robustness of our face detection-track system as it supports automatic tracking with no need to manual initialization or re-initialization and reaches best performance to deal with different challenging problems.

Keywords: face detection, face tracking, particle filter.

1 Introduction

Detecting and tracking faces with any view is an important problem since it has been observed that there is nearly 75% of faces in video are non-frontal [1]. This requires insuring permanent face spotting through the three degree of freedom of human head pose: yaw, pitch and roll. Thereby, detecting and tracking faces with any point of view is a very useful task mainly in visual surveillance applications, human computer interaction, facial biometric security solutions and driving assistance systems.

Actually, both automatic face detection and tracking have to deal with different problems. Concerning face detection task, it has to deal with various challenging problems such as: variation in pose, illumination, scale, and facial expression. With regard to tracking task there is no doubt that using pure tracking technique cannot be applied in real world scenarios since it reveals some deficiencies namely:

- Initialization problem: as soon as the new face enters the scene, we ought to provide this information to the tracker in order to start a new track.
- Lost track problem: when a face has left the scene, we have to update the tracker.
- Occlusion problem: this is a major problem since people tend to walk and interact in groups with others which increase the chances of having partial or complete occlusion.

Many previous studies have been reported on tracking face through video frames. Relying on the face representation, we can classify tracking methods into three main approaches: point-based approach, kernel-based approach and silhouette-based approach. Point-based approach represents the detected face as a set of relevant points. The association of the points is based on the previous face state [2, 3]. This approach requires face detection in every frame which is not always possible since the head is naturally moving within its three degree of freedom, this can made the detection task rough. Nevertheless, it can be used when the face is moving in certain narrow degree of rotation. The category of kernel-based approach is based on face appearance representation, namely templates and density-appearance models. In fact, Template matching is a brute force method which goes over the whole frame, looking for a similar region to the object template defined in the previous frame [4]. Instead of templates, other face representations can be used like color histograms [5] or mixture models [6] which can be computed by using the appearance of pixels inside the rectangular or ellipsoidal face region. The chief goal of a silhouette-based tracker is to find the face region in each frame using a face model generated in the previous frames. This face model can take the form of face shape or face contour. At respect to these face models, silhouette trackers can be divided into two sub-categories, namely, shape matching approaches and contour tracking approaches. Shape matching approaches search for the face silhouette in the current frame [7, 8]. However, the contour tracking approaches raise an initial contour to its new position in the current frame [9, 10].

In this paper, we propose an integrated detect-track system that can overcome the foremost problems usually emerging while detecting and tracking faces. Our proposed approach is composed of two main steps: Face detection step which was widely detailed in [18] and tracking step which will be depicted in this paper. Both of these two steps are running continually in each frame so that they complement each other very well to improve face localization in video sequences. In fact, when detection fails to detect face, tracking will secure face location; conversely when the tracker fails to update the tracker, the detection step will keep the tracker informed if a face has entered, left or reappeared. Concerning detection step, our proposed approach was based on the affluent combination of spatial (skin) and temporal (motion) information in video to filter out and restrict regions of interest. Regarding Tracking step we have used particle filter technique which is well known for its capability to deal naturally with problems arising from non-linearity and non-normality of the motion model

under many tracking scenarios. Moreover, we have classified each tracked face into five states: tracked, stopped, entered, left, or occluded, this will help understanding person behavior in the scene. To evaluate our proposed system for face detection and tracking in video, we have appraised several experiments. Our main contributions can be abridged in two points: adapting the system to detect and track multiple faces without scarifying real time performance and handling multiple complex challenging problems related to detection and tracking tasks.
The rest of the paper is organized as follows: We begin in the next section, by detailing our proposed detect-track system. The results of several experiments and performance evaluation are presented in section 3. We conclude this paper with remarks on potential extensions for future work.

2 Proposed approach

Figure 1 illustrates our proposed approach for automatic detection and tracking faces in video. It is composed of two successive modules: face detection module and face tracking module.

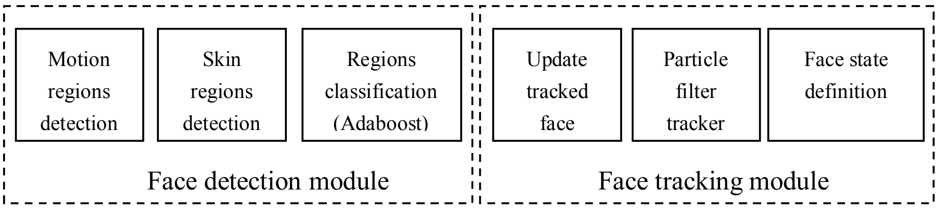

Fig. 1. The proposed detect-track approach

2.1 Face detection module

Regarding face detection module, we have proposed in [18] a hybrid approach which benefits from the affluent combination of spatial (skin) and temporal (motion) information in video to improve face detection. In fact, our approach starts by detecting regions in motion using approximate median background subtraction technique. Then, we go over these moving regions to find out pixels having skin color. Skin color detection task was performed using our proposed skin color model HSCr which was developed using a data-mining process. This restriction of regions of interest, by applying motion and skin color filtering, helps to reduce regions of interest and cuts out earlier false positive which can appear jointly with decor or non-skin regions. Finally, we classify these moving-skin color regions into face and non-face using Adaboost algorithm. In [18] we have proved the effectiveness of integrating spatial and temporal information as it helps to avoid tracking false positive faces.

2.2 Face tracking module

This module is composed of three main stages: update tracked face stage; particle filter tracker stage and face state definition stage.

Update Tracked face. The update tracked face stage seeks to update not only the face tracked structure but also the number of tracked faces either there is a new detection or not.
To start this stage, we have to record a first detection shot in order to initialize the tracker. Henceforward, we just applied the update tracked face stage which aims to keep the tracker informed with the detection module results.
Initialization step consists of assigning to the tracking face structure the new measures out came from the detected face structure. Notice here that a face structure (detected or tracked) is defined by its center coordinates (x, y), height, width, distance between the face and the camera and its current state (Leaving, Entering, Stopped, Occluded, Tracked).
In the case when the face detection module fails to detect faces in the scene, we update the tracked face structure and the number of tracked faces in the scene with the previous ones.
Otherwise, when we record a new detection, we have to identify if this new detection belong to an old tracked face or it is for a new entering face. Therefore, we go over each detected face and check if its center belongs to an old tracked face, if it fits this condition we just update the old tracked face structure with the new detected face structure; If not, we verify if there is an old occluded tracked face in the scene. If this is the case, we have to understand that this occluded face has just reappears again and then we update the occluded tracked face structure with the detected face structure without increasing the number of tracked faces.
Otherwise, if we do not find an old face tracked which fits the new detected face and there is no previous occluded face in the scene, we conclude that there is a new face which has just entered to the scene. Thus we have to update the tracked face list by adding the new detected face structure and incrementing the number of tracked faces in the scene.

Particle-Filter Tracker. At time t, each face is described by a state vector S_t. The goal of face tracking is to estimate the face state S_t at time t using a set of observations Z_t. In other words, it consists of constructing the posterior probability density function (pdf) defined as $p(S_t | Z_t)$.
To derive the prior pdf of the current state $p(S_t | Z_{t-1})$, the recursive Bayesian filtering provides the theoretically optimal solution which makes use of the dynamic equation and the already computed pdf $p(S_{t-1} | Z_{t-1})$ of the face state at time $t-1$ in the prediction step. Afterward, in the update step, it employs the face likelihood function $p(Z_t | S_t)$ of the current observation to compute the posterior pdf $p(S_t | Z_t)$.
Formally, it can be written as follows:

$$p(S_t|Z_{1:t}) = \frac{p(Z_t|S_t) \int p(S_t\ |S_{t-1}) p(S_{t-1}|Z_{1:t-1})\ dS_{t-1}}{\int p(Z_t\ |S_t\ ) \int p(S_t|S_{t-1}) p(S_{t-1}\ |Z_{1:t-1})\ dS_{t-1}\ dS_t} \quad (1)$$

As it showed in equation above, the computation of the posterior probability $p(S_t|Z_{1:t})$ using Bayesian filtering requires a high dimensional integrals computation and can deal with the non-linearity and non-normality of the motion model. High-dimensional integrations cannot be easily computed in a closed analytical form; hence a particle filter tracker was adopted to estimate the posterior probability.

Actually, it is widely accepted that the particle filter is superior to the traditional Kalman filter in terms of tracking performance [12], since the particle filter provides a robust object tracking without being restricted to a linear system model and proves performance to deal with limitations arising from non-linearity and non-normality of the motion model under many tracking scenarios [13, 14].

To perform tracking we have used particle filter technique with adaptive resampling and adjusted it for multiple faces tracking context.

In fact, the particle filter tracker maintains a probability distribution over the tracked face state (location, scale, etc.). Particle filters represent this distribution as a set of weighted particles samples. Each particle is an instantiation illustrating one potential location of the tracked face. Particles having high value of weight reveal the locations where the tracked face is more likely to be. This weighted distribution is propagated through time using the Bayesian filtering equations, the trajectory of the tracked face can be found by using the highest weighted particles. In our case, we have defined the face state as follows: S_t = {x, y, s, w, h, histo}, where (x, y) are the face coordinates center, s is the face scale, w and h are respectively the width and the height of the current face, histo is an histogram model of the current region.

The first step in particle filter tracker is initializing distribution which consists of creating N particles on each detected face. Then for each particle, we sample a new face state using second-order autoregressive dynamical model which is a type of random process often used to model and predict an output of a system based on the two previous outputs. Particle weight assignment was performed using Bhattacharyya distance between the predicted face region and the measured one. In fact, particles which have predicted correctly the face location will have high value of weight and conversely the particles which have fail to predict properly the face will have weak value of weight.

Actually, the algorithm consists in making evolve the particle set $P_t = (S_t^n, W_t^n)_{n = 1,\dots,N}$, where S_t is hypothetical state of the tracked face and W_t its weight. However, after some iterations, most of particles will have a weak weight close to zero; this is known as the degeneracy problem. The goal of resampling is to focus on high weighted particles and get rid of weak weighted particles by replacing the N particles according to their importance weights.

Unlike the existing works [17, 11] which use particle filter technique for tracking a single object, we have adopted this technique for tracking multiple faces by activating a different set of particles for every new detected face.

Face State definition. The state definition stage is a significant task since it contributes to update successfully faces, as well as understanding their behavior in the scene. Hence, we have differentiated five states: Tracked, Occluded, Stopped, Entered or Left.

- *Stopped Face State.*

A face is seen as stopped face if the Euclidean distance between its center in the current frame and the previous one does not exceed two pixels. Furthermore, the current state of the face should not be occluded since an occluded face has often a small amount of movements. Moreover, the distance separating the face from the camera should not be greater than one centimeter among the previous and the current frame (Figure 2).

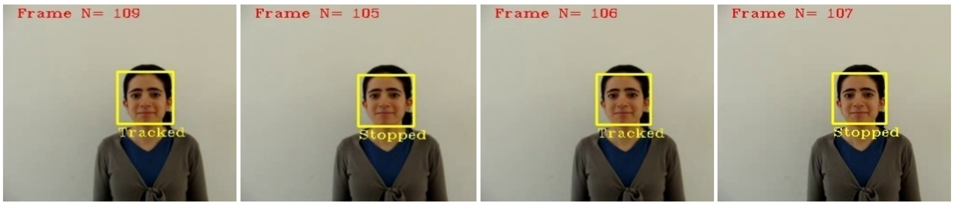

Figure 2.Stopped Face State

- Leaving and Entering Face State.

To identify leaving and entering face, we have defined a window of track as an area where the face could appear clearly in the scene. If the tracked face center is out of this window of track, we declare that it is either entering or leaving the scene. This window of track is defined as follows:

$$\begin{cases} X_{TrackWindowv} = X_{Frame} - 10 \\ Y_{TrackWindow} = Y_{Frame} - 10 \\ Height_{TrackWindow} = Height_{Frame} - 20 \\ Width_{TrackWindow} = Width_{Frame} - 20 \end{cases} \tag{2}$$

A face is perceived as a leaving face if its center goes beyond the window of track and the distance between the center of current tracked face and the frame center is larger than the distance between the center of previous tracked face and the frame center (i.e. the face is going far from the frame center)
Once we have identified a leaving state we have to update face tracked structure, thus we delete the left face structure from face tracked list and decrease the number of tracked faces in the scene (Figure 3).
A face is defined as an entered face, if its center is out of the window of track and the distance between the center of current tracked face and the frame center is lower than the distance between the center of previous tracked face and the frame center (i.e. the face is moving toward the frame center).

Figure 3.Leaving Face State

- Occluded Face State.

A face is classified as an occluded face in two cases: when there is a faces interaction constraint or there is a one-face occlusion. In fact, when the distance between the centers of two tracked faces is lower than the sum of their widths divided by two, we declared that we have a faces interaction case. However, when the number of skin color pixel in the tracked region is lower than 30% of the total face tracked region we asserted that we have a one-face occlusion case (Figure 4).

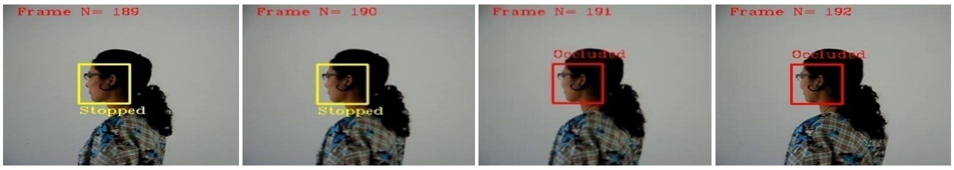

Figure 4.One-Face Occlusion Case (Total Head Rotation)

In the first instance, when faces interaction occurs, we have to identify which face occludes the other. Therefore, we compare the distance between each interacted face and the camera. The face having a larger distance than the interacted one will be classified as an occluded face. With regard to the face hiding the occluded one, it will be classified as a tracked face (Figure 5).

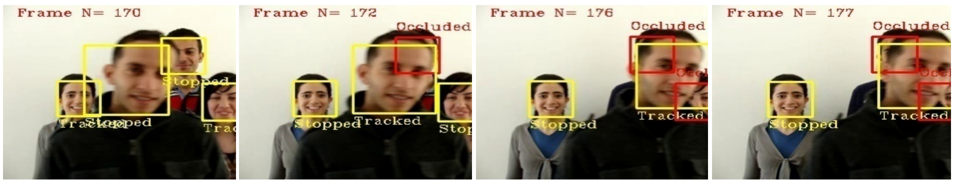

Figure 5. Faces Interaction Case

- Tracked Face

The tracked face state is the default state. In other words, when a face is not occluded, or stopped, nor entering or leaving, it is obviously moving in the scene and hence it is perceived as a tracked face.

3 Experimental results

In order to evaluate the performance of our proposed detect-track system, we have carried out two series of experiments. The first one dealt with the evaluation of the proposed detect-track system and the second one seeks to evaluate the performance of our adapted particle filter tracking technique.

3.1 First serie of experiments

This set of experiments aims to evaluate the performance of our proposed approach for face detection and tracking in video, as well as traces its behavior when only face detection module is applied.
Experimental study was performed on two video databases sequences, La Casia database [15] and our own recorded database. In fact, we have picked 10 sequences from La Casia database (5 sequences under uniform light and 5 sequences under varying light), also we have captured 15 sequences which deal with various challenging problems that can occur while detecting or tracking faces. Both video databases sequences description are summarized in table 1.

Video Databases	Constraints		# Sequences	# Frames	#Faces	Duration (seconds)
La Casia Database	Uniform Light		5	985	985	30
	Varying Light		5	985	985	30
Our Data-base	One Face	Free Head Rotation	2	266	266	8
		In/Out Face	2	194	141	6
		Partial/Total Occlusion	2	257	257	8
		Total Head Rotation	2	557	321	18
	Multiple Faces	Faces Interaction	3	989	2765	32
		In/ Out Faces	2	509	720	16
		Free Heads Rotation	2	347	694	11
TOTAL			25	5089	7134	129

Table 1. Video databases sequences description

Table 2 illustrates the obtained results while applying only face detection module and the recorded results by performing our face tracking module jointly with face detection module. To ensure evaluation, the familiar rates of precision and recall were used.
The table 2 chows that with tracking module, both of recall and precision rates were improved. In fact we record height rate of precision rate while applying only detection module or running the whole detect-track system. This was expected, since we process with a filtering stage while detecting faces using spatial (skin) and temporal (motion) information. This filtering stage helps to get rid of false positive detection in an earlier step. Moreover, we notice that even when there is a total head rotation, we

succeed to keep following it, so we jump from 61.9% of recall rate, to 100% thanks to our combination of detection and tracking modules

Video Databases	Constraints		Recall Rate		Precision Rate	
			Only Detection	Detect-Track	Only Detection	Detect-Track
La Casia Database	Uniform Light		82.5 %	100 %	100 %	100 %
	Varying Light		100 %	100 %	100 %	100 %
Our Data-base	One Face	Free Head Rotation	83.3 %	100 %	100 %	100 %
		In/Out Face	100 %	100 %	100 %	100 %
		Partial/Total Occlu-sion	73.6 %	100 %	100 %	100%
		Total Head Rotation	61.9 %	100 %	100 %	86.0 %
	Multiple Faces	Faces Interaction	93.4 %	100 %	100 %	97.8 %
		Free Heads Rotation	69.5 %	99.3 %	100 %	92.2 %
		In/Out Faces	100 %	100 %	100 %	100 %

Table 2. Precision and recall rates of the detection module alone and the whole detect-track system

Nevertheless, we perceive that the precision rate decrease little bit while tracking an occluded face. In fact, this precision decrease is due to the fast movement of the occluded face, as the face is not only occluded but also moves quickly. Although, this has affected slightly the precision rate but it does not touch the general performance of our detect-track system since a new detection will update naturally the face tracker.

3.2 Second serie of experiments

With regard to tracking technique evaluation, we have compared our adapted particle filter face tracker technique with the referenced tracking technique CamShift [16]. To insure such evaluation, we suggest a new distance measure to appraise the performance of each face tracker technique. This distance measure is defined as the difference between X and Y ground truths trajectories and the X and Y tracked trajectories averaged over all frames in each video sequence. Seeing that this distance is an average measure, we have rounded it to the nearest whole unit. In addition, since we have dealt with large databases we cannot display all the computed results, so only one example of each constraint will be exposed. Table 3 sums up this comparative study.
Table 3 depicts the obtained results with our adapted particle filter and CamShift techniques. In fact, these results reveal the sensibility of CamShift tracker when some constraints arise. In particular, when the face is occluded the Camshift tracker fails to follow it and drift away with 22 pixels from the Y ground truth trajectory however the adapted particle filter keeps pursuing correctly this occluded face (only 3 pixels away from the Y ground truth trajectory).

Video Databases	Constraints			Adapted Particle Filter		CamShift	
				Dist_X	Dist_Y	Dist_X	Dist_Y
La Casia Database	Uniform Light (Seq: jam1)			3	2	4	4
	Varying Light (Seq: jal5)			5	3	39	6
Our Database	One Face	Free Head Rotation (Sequence number 2)		3	2	10	5
		In/Out Face (Sequence number 2)		6	1	3	7
		Partial/Total Occlusion (Sequence number 2)		3	3	3	22
		Total Head Rotation (Sequence number 2)		8	1	23	17
	Multiple Faces	Faces Interaction (Sequence number 2)	Face 1	4	2	9	6
			Face 2	5	2	17	10
			Face 3	3	2	12	14
			Face 4	7	1	15	11
		In/Out Faces (Sequence number 1)	Face 1	3	1	3	4
			Face 2	2	2	3	2
		Free Heads Rotation (Sequence number 2)	Face 1	4	3	9	6
			Face 2	5	4	10	5

Table 3. Performance comparison of Particle filter and CamShift trackers Techniques

Furthermore, our proposed particle filter seems to be more faithful to the ground truth trajectory then CamShift tracker in varying illumination constraint since CamShift tracker discards with 39 pixels from the X ground truth trajectory while the particle filter tracker keep being close (5 pixels) to the X ground truth trajectory. Moreover, regarding faces interaction constraint which is a complex challenge since faces are moving one next the other in such way that they hide each one the other, our adapted particle filter tracker proves more performance than CamShift tracker seeing that particle filter trajectory along X and Y is more closer to the ground truth trajectory than CamShift trajectory. Based on this report, we can conclude that CamShift tracker usually drifts away and in some times looses tracking when tracking error propagates through video frames. This problem is especially relevant when partial or total face occlusion occur or nearby faces appear with similar distributions to the target face.

Figure 6 displays the X-trajectory of the second video sequence under faces interaction constraint. Through this figure, we can see that the second and the third faces enter successively to the scene at the frame number 41 and 102 respectively; then we can notice how the fourth face enters at the frame number 165 and pass in front of the three faces and occludes them; however our face detect-tracker system succeed to keep following them although they are really occluded.

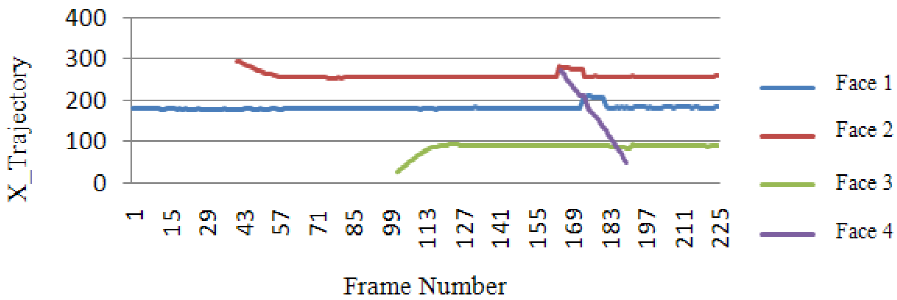

Figure 6.X-Trajectories of faces in interaction

This experimental study has highlighted the robustness and the effectiveness of our proposed approach for face detection and tracking in video. In fact, the combination of the detection and tracking modules supports automatic tracking with no need to manual initialization or re-initialization and reaches best performance to deal different challenging problems which can arise while face detection or tracking stages. This is achieved without affecting the quality of tracking or its computational complexity.

4 Conclusion

A face detect-track system was proposed in this paper. The face detection module makes use of motion information as well as skin color to earlier get rid of false positive. Based on detection modules results we update the tracking module. Then we adapt the particle filter technique for multiple faces track scenario without affecting its performance. Finally we define a specific state for each tracked face. This state definition step helps to understand each face behavior in the scene; hence we have defined five face states: Tracked, Occluded, Stopped, Entered or Left.

The experimental results have proved the effectiveness and the performance of our proposed combination of face detection and tracking modules to deal with the various face detection and tracking problems. In fact, the detection module helps to handle tracking problems and conversely, the tracking module provide an intelligent way to handle detection challenges.

In our future works, we seek to estimate continually the pose of the tracked faces. Such works will help to capture the best frame having the most reliable pose for a further face recognition task.

References

1. Kuchinsky, A., Pering, C., Creech, M.L., Freeze, D., Serra, B., Gwizdka, J. Fotofile: Consumer multimedia organization and retrieval system, Proc. ACM SIG CHI99 Conf, (1999)

2. Xiaokun, L., Chiman, K., Gang, M., Baoxin, L. : A Generic Approach to Object Matching and Tracking, Third international conference, ICIAR, (2006)
3. Adipranata., R, Ballangan, C.G., Rostianingsih, S., Ongkodjodjo, R.P. :Real-Time Human Face Tracker Using Facial Feature Extraction, International Conference on Soft Computing (2007)
4. Romero, M., Bobick, A. : Tracking Head Yaw by Interpolation of Template Responses", CVPRW '04 Proceedings of the 2004 Conference on Computer Vision and Pattern Recognition Workshop (2004)
5. Comaniciu, D., Ramesh, V., Andmeer, P.: Kernel-based object tracking. IEEE Trans. Patt. Analy., 564–575 (2003)
6. Swaminathan, G., Venkoparao, V., Bedros, S.: Multiple appearance models for face tracking in surveillance videos, In Proceedings of AVSS, 383-387 (2007)
7. Hou,Y., Sahli, H., Ilse,R., Zhang, Y., Zhao,R.: Robust Shape-Based Head Tracking, Proceedings of the 9th international conference on Advanced concepts for intelligent vision systems (2007)
8. Kohsia, S. H., Mohan, M.T.: Robust Real-Time Detection, Tracking, and Pose Estimation of Faces in Video Streams, the 17th International Conference on Pattern Recognition (2004)
9. Sobottka, K., Pitas, I.: A novel method for automatic face segmentation, facial feature extraction and tracking, Signal Processing: Image Communication, 263-281 (1998)
10. Gunn, S. R., Nixon, M.S.: Snake Head Boundary Extraction using Global and Local Energy Minimization, Proceedings of the 13th ICPR, 581-585 (1996)
11. Perez, P., Hue, C., Vermaak, J., Gangnet, M.: Color-Based Probabilistic Tracking, European Conference on Computer Vision, 661–675 (2002)
12. Chang, C., Ansari, R., Khokhar, A.: Multiple object tracking with kernel particle filter, in: Proc. IEEE Conf. on Computer Vision and Pattern Recognition (2005)
13. Li, P., Zhang, T., Pece, A.E.C.: Visual contour tracking based on particle filters, Image Vis. Comput. 21, 111–123 (2003)
14. Okuma, K., Taleghni, A., Freitas, N.D., Little, J. J., Lowe, D.G. : A boosted particle filter: multitarget detection and tracking, in: Proc. European Conf. on Computer Vision, 28–39 (2004)
15. La-Cascia,M., Sclaro_,S., Athitsos,V.: Fast, Reliable Head Tracking under Varying Illumination: An Approach Based on Registration of Texture-Mapped 3D Models. IEEE Transactions On Patterns Analysis and Machine Intelligence (2000)
16. Bradski, G.R.: Computer Vision Face Tracking For Use in a Perceptual User Interface, in Proc. IEEE Workshop on Applications of Computer Vision, 214–219 (1998)
17. Andrew, B., Zulfiqar, H.K., Irene, Y.H.G.: Robust Object Tracking Using Particle Filters and Multi-region Mean Shift, Circuits and Systems for Video Technology, IEEE Transactions on, 74 –87 (2011)
18. Mliki, H., Hammami, M., Ben-Abdallah, H.: Real time face detection based on motion and skin color information, Appeared in the 6th International Conference on Multimedia and Ubiquitous Engineering (2012)