Robust Facial Feature Tracking

Fabrice Bourel, Claude C. Chibelushi, Adrian A. Low
School of Computing, Staffordshire University
Stafford ST18 0DG
F.Bourel@staffs.ac.uk
C.C.Chibelushi@staffs.ac.uk
A.A.Low@staffs.ac.uk

Abstract

We present a robust technique for tracking a set of pre-determined points on a human face. To achieve robustness, the Kanade-Lucas-Tomasi point tracker is extended and specialised to work on facial features by embedding knowledge about the configuration and visual characteristics of the face. The resulting tracker is designed to recover from the loss of points caused by tracking drift or temporary occlusion. Performance assessment experiments have been carried out on a set of 30 video sequences of several facial expressions. It is shown that using the original Kanade-Lucas-Tomasi tracker, some of the points are lost, whereas using the new method described in this paper, all lost points are recovered with no or little displacement error.

1 Introduction

Motion tracking based on points is often important in many applications requiring time-varying image analysis. Applications may be object tracking, motion understanding, navigation, automatic speechreading, and facial feature tracking. This paper presents a new robust technique based on the Kanade-Lucas-Tomasi [9, 18, 17] tracker. The technique focuses on the automatic recovery of points lost between frames. It is specialised for robust tracking of human facial features. This new method is intended for use in robust recognition of facial information such as the identity or facial expression of a person. Such recognition applications are important components of future machine interfaces, for example. Instead of trying to improve tracking performance through the automatic selection of better features, as proposed in [19], here we exploit the knowledge that the tracker is working on a human face. Several constraints are applied during the initial selection and tracking of feature points.

Several general-purpose point trackers have been proposed. Lucas and Kanade [9] have worked on the tracking problem and proposed a method for registering two images for stereo matching based on a translation model between images. From the initial work of Lucas and Kanade, Tomasi and Kanade [18] developed a feature tracker based on the 'sum of squared intensity differences (SSD)' matching measure, using a translation model. Then, Shi and Tomasi [17] proposed an affine transformation model. Over small inter-frame motion, the translation model has higher reliability and accuracy than the affine model. However, the affine method is preferable and more adequate over a longer

BMVC 2000 doi:10.5244/C.14.24

time span. Tommasini *et al.* [19] proposed a robust tracker based on the work of Shi and Tomasi [17] by introducing an automatic scheme for rejecting spurious features.

Some point tracking techniques focus on solving the correspondence problem for incomplete trajectories in an image sequence. Some features may enter or leave the field of view or become occluded. Many methods, possessing different capabilities and linking strategies, have been proposed for finding correspondences. Examples are the algorithms of Chetverikov and Verestoy [1], Sethi and Jain [16], Hwang [3], Salari and Sethi [15], Rangarajan and Shah [14]. A comparison of these methods is given in [20].

Many trackers have been investigated within the framework of specific applications. Verestoy *et al.* [21] present a paper where several feature point trackers are used for particle image velocimetry. Among those trackers are the IPAN tracker [1] and the Kanade-Lucas-Tomasi tracker [9, 18, 17]. Point trackers have also been used to track facial features. McKenna *et al.* [11] proposed an approach to track rigid and non-rigid facial motion based on a point distribution model (PDM) and Gabor wavelets. Petajan [12] uses facial feature tracking to track the eyes and the nostrils. Huang and Huang [2] also use a PDM approach to extract facial features. Their method measures the variation of the position of each point. Lien *et al* [8] also use point tracking to recognise facial actions corresponding to expressions. Many lip trackers are variants of the snake method of Kass and Terzopoulos [5] or of the deformable template technique of Yuille [22]. Luettin [10] uses an active shape model.

Many of the techniques do not address the problem of recovering points lost during the tracking. A point may be lost because of variation in lighting conditions, head motion (rigid and non-rigid), temporary occlusion, or because of gradual tracker drift away from the correct position. These problems need to be solved to achieve robust tracking. The techniques proposed in this paper constitute a step in this direction.

2 Description of the tracker

The proposed facial feature tracker is based on the Kanade-Lucas-Tomasi (KLT) tracker. The new tracker is specialised to work on human faces. The nostrils are used as main ”anchor” points. In addition, constraints imposed by the configuration of human faces, are exploited during tracking. In the main, the recovery of lost points is based on determining a search window around the region of interest.

The method described here is concerned with several issues that are considered to be important for robust recognition of facial expressions. These issues are rigid and non-rigid motion, variation of lighting conditions, head orientation, head tilting, and real-time tracking. The proposed point tracking scheme aims to tackle all those issues. The technique is designed to be robust and to recover points lost during tracking. The main constraint is for the nostrils to be visible. The tracker works on grey level image sequences of any length. The general tracking process is shown in Figure 1.

2.1 Initialisation

The tracker uses a set of 12 points: one per nostril, one for each mid-point of the upper and lower lip, one for each mouth corner, three for each eyebrow (Figure 2). In the work presented herein, these 12 points are manually set in the first image of a sequence.

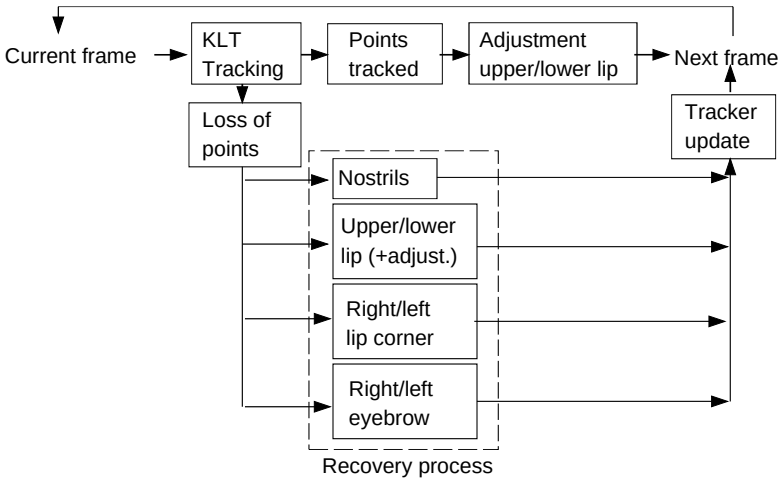

Figure 1: Facial feature point tracker diagram.

Figure 2: The 12 facial points.

2.2 Nostril tracking and recovery

The tracker is anchored on the nostrils. The latter provide a reliable way of tracking human faces and give some information about the orientation and size of the face [12]. When the other facial features have to be recovered, the nostrils constitute the base of a polar coordinate system used for determining search windows for points to be recovered. Even though the nostrils can be easily tracked with the KLT tracker alone, they may still be lost during tracking. Hence, in the proposed enhanced KLT tracker, the nostrils are recovered using information in the frame just before they disappeared. If both nostrils are lost, a search window is computed. Its centre is the average of the previous position of the nostrils and its size is chosen heuristically to be big enough to include the possible new position. The search window size is proportional to the Euclidean distance between the last known nostril positions. This has the effect of automatically scaling the window

if the person gets closer or further from the camera.

The nostrils are assumed to be the darkest areas in the search window. Therefore, automatic thresholding, with a threshold corresponding to the lower 5 percent of the local histogram for the window, is applied. This isolates the two nostrils and possibly other small areas corresponding to the valleys between the cheeks and the nose. After thresholding, a simple flood filling is applied to the search window and the small dark areas are removed. The centre of the nostrils is then calculated by averaging all coordinates in the region assumed to represent a nostril. Figure 3 illustrates the thresholding method in a search window. The recovered nostril centres are used for tracking the points in the next frame. As in [12], this algorithm, is fairly insensitive to noise, illumination, nostril shape, head orientation, scale and tilting.

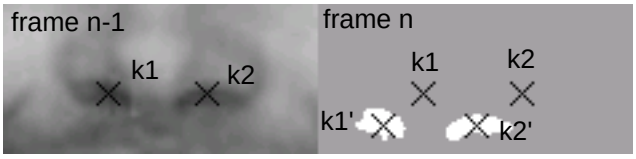

Figure 3: Nostril recovery (points $k1'$ and $k2'$) inside a search window.

2.3 Mouth tracking and recovery

Tracking the mouth is a bit more complicated than tracking and recovering the nostrils, but it uses the same kind of ideas. There are four points associated with the mouth. One at each mid-point of the top and bottom of the lips, and one at each mouth corner (Figure 2). The method proposed divides tracking and recovery of the top/bottom and left/right side of the lips in two independent steps, one for the lip mid-points, and another for the mouth corners.

2.3.1 The upper and lower lip features

Detecting and tracking the upper and lower lip is not straightforward. The upper lip is easily detectable and trackable as feature information is usually strong, but the lower lip may cause some problems. In an image, the ”signature” (colour, gradient magnitude, texture) of the lower lip is sometimes so small that it is unusable. Several solutions to this problem have been proposed [7, 10, 6, 13], some are based on colour information [6, 13]. The method proposed in this paper, uses the KLT tracker coupled to some post-processing to ensure robust and consistent tracking of the upper and lower lip points.

When the upper and lower lips are being tracked, a problem that may arise is that the points may slowly drift away. Therefore, a scheme for automatically realigning lip mid-points has been developed. The adjustment of lip mid-points during tracking is implemented as follows: the points are adjusted to lie on the perpendicular line passing between the nostrils. A lip mid-point before and after the realignment lies on a circle centred on the middle of the nostrils, with radius equal to the distance to the lip mid-point. Figure 4 illustrates the adjustment of lip mid-points to counter tracker drift.

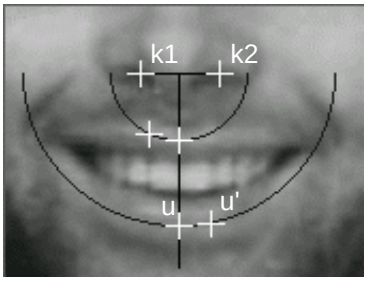

Figure 4: Adjustment of upper and lower lip mid-points.

To recover a point that got lost from frame $n-1$ to n, the position of that point in frame $n-1$ is used. For instance, if the lost point is the lower lip point v, then the Euclidean distance from m to v (named d_v) is calculated (see Figure 5). Thereafter, in frame n, the distance d_v is applied from m, along the perpendicular to the inter-nostril line. This yields the recovered mid-point v'. Only a small displacement along the constraining vertical line may be lost by reusing the previous position from frame $n-1$. This will be improved in the future to remove the displacement error that may be introduced if the lips move between frame $n-1$ and n.

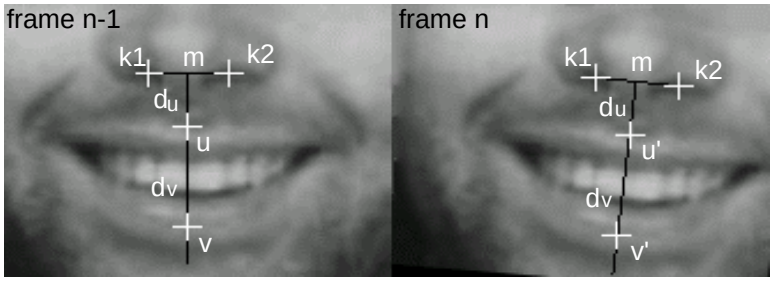

Figure 5: Recovery of the upper and lower lip mid-points.

2.3.2 The mouth corners

Similar to the procedure used for the nostrils, recovery of lost mouth corners is based on the automatic detection of dark zones inside search windows. Search windows are thresholded and small blobs are eliminated as described in Section 2.2. The new mouth corner positions are the extremities of the dark zones. Figure 6 illustrates the thresholding and mouth corner detection operations.

Here however, a search window is determined differently from the technique used for the nostrils. The positions used are the last known positions of the nostrils and mouth corners. The search window size is equal to the distance separating nostrils (Figure 7),

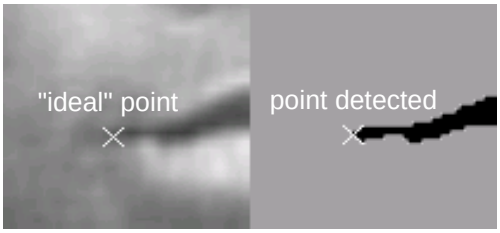

Figure 6: Recovery of a mouth corner.

but it may be multiplied by a scaling factor if it is too small. For instance, let us consider that the right mouth corner u_n is lost during tracking in frame n. Therefore, to recover this point, we have to find a search window of centre s_n in frame n, before applying the automatic algorithm for detecting dark zones. The polar coordinates (R, D), of u_{n-1} relative to the right nostril N, are calculated in frame $n-1$. Then, in frame n, the centre s_n of the search window is determined, relative to the right nostril N. s_n is the position of u_{n-1} in frame n. Once the search window is defined, the point u_n is recovered (see Figure 7) using the method illustrated in Figure 6.

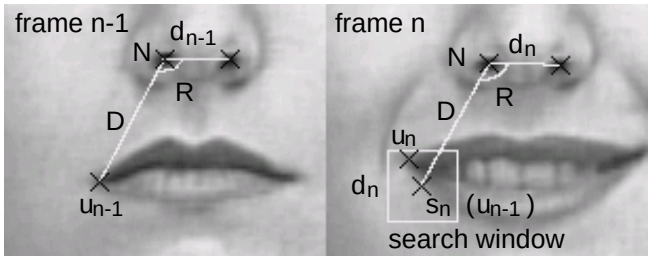

Figure 7: Determining the search window to recover the right mouth corner.

2.4 Eyebrow tracking and recovery

To recover lost points, search windows are determined using an approach similar to the one described earlier (Section 2.3.2). Unlike the thresholding method used for the recovery of the nostrils and the mouth corners, grey levels in the search window are inspected to find out if the point has drifted from its position. A simple block matching operation is performed using 9x9 blocks. All possible blocks contained in the search window of frame n are matched to the block corresponding to the last known position of the point in frame $n-1$. The centre of the best matching block from the current search window is then used as new position for the feature point.

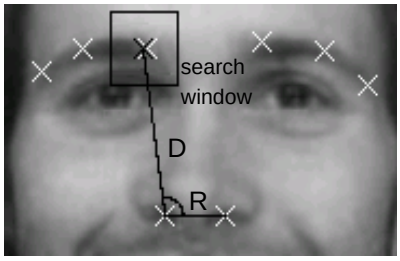

Figure 8: Recovery of a point on one of the eyebrows.

3 Performance assessment

Experiments were conducted using a set of 30 grey level image sequences [1] (image dimensions: 640x480 pixels) of facial expressions of people from different ethnicities. The length of the sequences ranged from 30 to 70 frames. First, the KLT tracker alone was tested. Then, the KLT coupled to the recovery method proposed in this paper was assessed. The latter tracker is hereafter referred to as enhanced KLT (EKLT).

3.1 KLT tracker

Table 1 shows, for 30 image sequences, the number of sequences in which some feature points were lost, temporarily occluded or drifted. For instance, the upper lip point got lost during tracking in 2 sequences out of 30. Tracking the nostrils proved to be very robust and reliable. Some of the facial features were lost by the KLT tracker, sometimes early during the tracking, and others drifted away slowly from their correct position. Some points were lost for no visually apparent reason, others got lost due to motion blurring in certain frames. Many lost points were on the lower lip. Points that drifted away from their target were mainly the upper and lower lip mid-points. Mouth corners were lost a couple of times. The eyebrows were rarely lost.

3.2 Enhanced KLT tracker

The performance improvement by the EKLT tracker over the KLT tracker was significant. *All* the facial points were tracked and recovered in *all* image sequences. The points on the upper and lower lips that drifted horizontally in the KLT tracker were suitably readjusted. The tracking error was also minimal when the lower lip feature disappeared, due to the person being angry for instance. The EKLT tracker displayed substantial insensitivity to motion, head tilting and lighting conditions.

Figure 9 gives an estimate of the tracking accuracy of the EKLT tracker. These measurements were taken for 5 frames equally spread across each of the 30 image sequences. This gave a total of 150 measurements per tracked facial point. Manually determined positions of facial points in the 150 frames were used as reference for measuring displacement error. The error was calculated as the Euclidean distance between the reference points and

[1] Permission to use the CMU-Pittsburgh AU-Coded Facial Expression database [4] is gratefully acknowledged.

	Event during tracking		
Facial feature	Loss (*)	Loss (occlusion)	Drift
Nostril	0	0	0
Upper lip	2	0	3
Lower lip	11	4	7
Mouth corner	3	0	0
Eyebrow	1	0	0

(*) Due to motion, changes in texture, lighting variation.

Table 1: Number of sequences where the KLT tracker lost or drifted away from individual facial points.

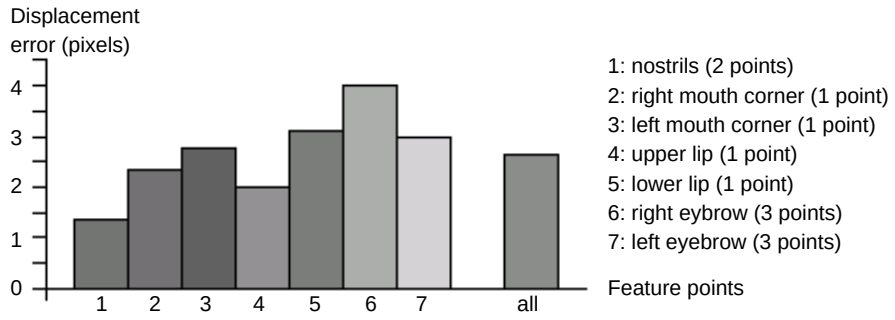

Figure 9: Average Euclidean distance between facial points tracked by the EKLT tracker and their ideal positions.

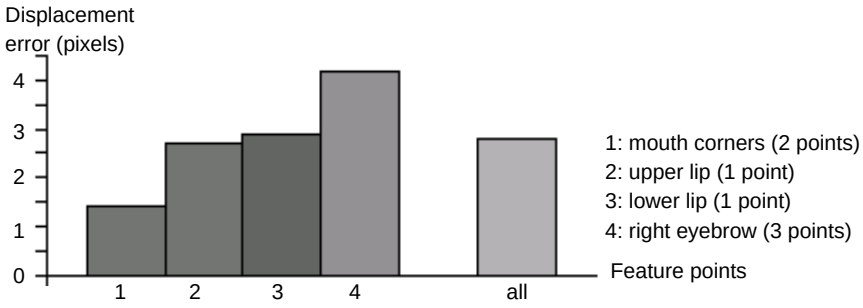

Figure 10: Average recovery error, for the EKLT tracker, measured for frames where the KLT tracker got lost.

the points returned by the tracker. Image scale was approximately constant across all sequences, hence the measured displacement errors (in pixels) are nearly proportional to spatial error measurements in the physical world. The overall average error of nearly 2.5 pixels (see Figure 9) shows that the EKLT tracker is quite accurate. As expected, for the 12 feature points used, the nostrils have the smallest tracking error. Figure 10 shows a low average tracking recovery error for the EKLT tracker, thereby illustrating the effectiveness of the enhancements added to the KLT tracker.

Some aspects of the tracker still have to be improved. An important one, is sensitivity to occlusion of the nostrils, which are used as anchor for position estimation. Another parameter that currently affects the performance of this technique is the frame rate of the camera.

4 Conclusion

An enhanced tracker for facial feature points, derived from the Kanade-Lucas-Tomasi tracker, has been presented. Automatic recovery, which uses the nostrils as a reference, is performed based on some heuristics exploiting the configuration and visual properties of faces. Experimental assessment has shown that the EKLT tracker outperforms the KLT tracker, in terms of recovering lost or drifting points. Twelve facial feature points have been tracked correctly throughout the 30 image sequences in the test set. Our current work focuses on automatic facial feature initialisation, and self-reinitialisation in case of total occlusion. Future work will incorporate the tracker into a robust system for the recognition of facial expressions.

References

[1] D. Chetverikov and J. Verestoy. Tracking feature points: A new algorithm. *Proc. International Conf. on Pattern Recognition*, pages 1436–1438, 1998.

[2] C. Huang and Y. Huang. Facial expression recognition using model-based feature extraction and action parameters classification. *Journal of Visual Communication and Image Representation*, 8(3):278–290, 1997.

[3] V. Hwang. Tracking feature points in time-varying images using an opportunistic selection approach. *Pattern Recognition*, 22:247–256, 1989.

[4] T. Kanade, J. F. Cohn, and Y. L. Tian. Comprehensive database for facial expression analysis. *Proc. 4th IEEE International Conference on Automatic Face and Gesture Recognition (FG'00)*, pages 46–53, 2000.

[5] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, 1987.

[6] R. Kaucic and A. Blake. Accurate, real-time, unadorned lip tracking. *Proc. 6th Int. Conf. Computer Vision, Bombay, India*, pages 370–375, 1998.

[7] R. Kaucic, B. Dalton, and A. Blake. Real-time lip tracking for audio-visual speech recognition applications. *Proc. ECCV'96, Cambridge, UK*, pages 376–387, 1996.

[8] J. J. Lien, T. Kanade, J. F. Cohn, and C. Li. Subtly different facial expression recognition and expression intensity estimation. *Proc. International Conference on Computer Vision and Pattern Recognition, Santa-Barbara, CA*, pages 853–859, 1998.

[9] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. *Proc. International Joint Conference on Artificial Intelligence*, pages 674–679, 1981.

[10] J. Luettin, A. N. Thacker, and S. W. Beet. Locating and tracking facial speech features. *Proc. ICPR'96, Vienna, Austria*, 1:652–656, 1996.

[11] S. McKenna, S. Gong, R. P. Wurtz, J. Tanner, and D. Banin. Tracking facial feature points with Gabor wavelets and shape models. *Proc. Int. Conf. on Audio- and Video-Based Biometric Person Authentication, Crans-Montana, Switzerland*, pages 35–42, 1997.

[12] E. Petajan and H. P. Graf. Robust face feature analysis for automatic speechreading and charcter animation. *Proc. Second International Conference on Automatic Face and Gesture Recognition, Killington, Vermont*, pages 357–362, 1996.

[13] M. U. Ramos Sanchez, J. Matas, and J Kittler. Statistical chromaticity models for lip tracking with B-spline. *Proc. Int. Conf. on Audio- and Video-Based Biometric Person Authentication, Crans-Montana, Switzerland*, pages 69–76, 1997.

[14] K. Rangarajan and M. Shah. Establishing motion correspondence. *CVGIP: Image Understanding*, 54:56–73, 1991.

[15] V. Salari and I.K. Sethi. Feature point correspondence in the presence of occlusion. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 12:87–91, 1990.

[16] I.K. Sethi and R. Jain. Finding trajectories of feature points in a monocular image sequence. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 9:56–73, 1987.

[17] J. Shi and C. Tomasi. Good features to track. *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 593–600, 1994.

[18] C. Tomasi and T. Kanade. Detection and tracking of feature points. *Carnegie Mellon University Technical Report CMU-CS-91-132, Pittsburgh, PA*, 1991.

[19] T. Tommasini, A. Fusiello, M. Trucco, and V. Roberto. Making good features track better. *Proc. IEEE Int. Conf. on Computer Vision and Pattern Recognition, Santa Barbara, CA*, pages 178–180, 1998.

[20] J. Verestoy and D. Chetverikov. Comparative performance evaluation of four feature point tracking techniques. *Proc. 22nd Workshop of the Austrian Pattern Recognition Group, Illmitz, Austria*, pages 255–263, 1998.

[21] J. Verestoy, D. Chetverikov, and M. Nagy. Digital PIV: A challenge for feature based tracking. *Proc. 23nd Workshop of the Austrian Pattern Recognition Group*, pages 165–174, 1999.

[22] A. Yuille. Feature extraction from faces using deformable templates. *International Journal of Computer Vision*, 8(2):99–111, 1992.