Feature point tracking combining the Interacting Multiple Model filter and an efficient assignment algorithm

David Marimon, Yousri Abdeljaoued, Bruno Palacios and Touradj Ebrahimi

Signal Processing Institute
Swiss Federal Institute of Technology (EPFL)
CH-1015 Lausanne, Switzerland

ABSTRACT

An algorithm for feature point tracking is proposed. The Interacting Multiple Model (IMM) filter is used to estimate the state of a feature point. The problem of data association, i.e. establishing which feature point to use in the state estimator, is solved by an assignment algorithm. A track management method is also developed. In particular a track continuation method and a track quality indicator are presented. The evaluation of the tracking system on real sequences shows that the IMM filter combined with the assignment algorithm outperforms the Kalman filter, used with the Nearest Neighbour (NN) filter, in terms of data association performance and robustness to sudden feature point manoeuvre.

Keywords: feature point extraction and tracking, Interacting Multiple Model filter, assignment algorithm, track management

1. INTRODUCTION

In recent years, there has been a growing interest in the problem of matching natural features, with applications such as video summarization,[1] augmented reality[2] or mosaicing,[3] to mention a few.

Research in this area aims at resolving two main issues, namely, state estimation and data association. Feature Points (FP) are tracked throughout an image sequence. The position of a feature point (FP) is predicted and this prediction is corrected with the associated FP detected in each frame. Most feature point tracking techniques are based on the motion smoothness constraint.[4,5] That is, the motion of a feature point does not undergo abrupt changes. However, these constraint is very restrictive due to the wide variety of real video sequences.

In this paper, we present a combination of a state estimator, the Interacting Multiple Model (IMM) filter,[6] and a data association algorithm, the assignment solution proposed by Jonker and Volgenant (JV),[7] derived from the Hungarian algorithm.[8] We tailor this combination to the problem of tracking feature points in real sequences where motion can change rapidly (i.e., different motion models) and FPs are extracted in textured regions (i.e., FPs closely located). FPs in real scenarios are best tracked with different motion models filtering at the same time. Moreover, feature points are extracted in textured regions. Consequently, it is likely that these feature points are located close to each other. In such environments, the assignment algorithms such as JV are best suited. Therefore, this approach shows robustness in front of different manoeuvres and dense FP environments. IMM-JV has been successfully used for pure multi-target tracking,[9] and video summarisation and manipulation.[1]

This paper is structured as follows. First, Section 2 describes the state-of-the-art. The techniques involved in the proposed combination, namely IMM and JV, are presented in Sections 3 and 4. The core of our tracking technique is deeply dealt with in Section 5. Several experiments and their results are explained in Section 6. Conclusions and future paths of research are finally discussed.

---

e-mail: {david.marimon, yousri.abdeljaoued, touradj.ebrahimi}@epfl.ch

2. RELATED WORK

The feature point tracking techniques proposed in the literature can be classified into heuristic and statistical methods.[5]

Most of the heuristic techniques are based on the motion smoothness constraint, which prevents its application to real sequences.[5] Sethi and Jain[4] expressed this motion smoothness constraint by the cost function of assigning a FP (i.e. the measurement) in a certain frame to a particular track in the previous frame. They developed an iterative greedy exchange algorithm for the data association step. This algorithm has a number of drawbacks. Its complexity is of $O(KM^2)$, where K is the number of frames and M is the number of tracks. Because the number of tracks M is assumed to be constant, track initiation and track deletion are not considered. Salari and Sethi[10] improved the FP tracking system of[4] by extending its track management capabilities to track initiation and deletion. No track continuation is provided by this algorithm. Chetverikov and Verestoy[11] developed an algorithm which includes track initiation, deletion and continuation. For each FP, candidate measurements are searched in the previous and next frame. Search areas are determined by a maximal speed. Data association is the result of a competitive hypothesis testing process. Veenman *et al*[5] also formulated the problem of data association as an assignment problem. They proposed to use the Hungarian algorithm.[8] A number of experiments show that this algorithm outperforms all the other presented heuristic algorithms in terms of tracking performance and parameter sensitivity. The main drawback of all the presented heuristic algorithms is assuming the motion smoothness constraint.

Most of the statistical algorithms are based on the Kalman filter (KF). Yao and Chellappa[12] used an extended KF (i.e. the extension of the KF to a nonlinear system) for the prediction of the FP in the next frame. Then, a set of tests composed of a validation process, a similarity test based on Gabor responses, and a motion smoothness criterion similar to,[4] was used to select the best measurement-track association. Track deletion is handled by means of data association probabilities from a Probabilistic Data Association filter (PDA). FPs are extracted only at the first frame and when the track estimate becomes unreliable. Because of the local smoothness constraint, manoeuvreing FPs are not considered. It is also important to note that the problem of track management is not rigorously addressed. Cox and Hingorani[13] proposed an efficient implementation of the original Multiple Hypothesis Tracking (MHT)[14] filter for the tracking of FPs. This implementation has a polynomial computational complexity. The validation is supplemented with an intensity-based cross-correlation test. The MHT filter has an efficient build-in track management process. Due to the lack of a mechanism dealing with manoeuvreing FPs, different process noise settings are chosen for the different test sequences. In order to make the MHT filter less sensitive to the parameters choice, Tissainayagam and Suter[15] suggested to use the Interacting Multiple Model filter (IMM) instead of the KF. They consider different velocity and acceleration motion models, and good tracking results have been reported. IMM-JV has been successfully used for pure multi-target tracking.[9] To the best of our knowledge it has only been used for video processing in our previous work for video summarisation and manipulation.[1]

In the last few years, new techniques have emerged that perform FP tracking by detection. This is the case of the SIFT descriptor[16] and the key point detector of Lepetit *et al*.[17] These techniques concentrate their efforts on invariance to viewpoint (scale, rotation, appearance) leading to improved FP recognition. The main drawback of these approaches is the lack of track management, being the number of FPs fixed.

3. IMM FILTER

The IMM filter uses several possible models for the target's motion and a probabilistic switching between these models. During each sampling period (i.e. between two consecutive frames), we assume that the target's motion is described by one of the models. The IMM is implemented with multiple parallel filters, where each of the filters corresponds to one of the assumed models. Because of the switching between models, there is an exchange of information between the filters. During each sampling period, all the filters of the IMM are in operation. The overall state estimate is a combination of the estimates from the individual filters.

A generic IMM filter using a Kalman filter for each model is briefly presented in this section. The set of equations involved in one iteration of the IMM filter can be divided into four steps: interaction/mixing, filtering, model probability update, and combination. For a derivation of these equations, we refer to.[18]

Let us denote by $L^1, L^2, \cdots, L^r$ the r models of the IMM filter, and by $L^j(k)$ the event that model L^j is in effect during the sampling period ending at frame k.

1. **Interaction/Mixing**: For the event $L^j(k+1)$, the mixed estimate $\hat{\boldsymbol{x}}^{0j}(k|k)$ and the covariance $\boldsymbol{P}^{0j}(k|k)$ are computed as

$$\hat{\boldsymbol{x}}^{0j}(k|k) = \sum_{i=1}^{r} u^{i|j}(k|k)\hat{\boldsymbol{x}}^i(k|k) \tag{1}$$

$$\boldsymbol{P}^{0j}(k|k) = \sum_{i=1}^{r} u^{i|j}(k|k)\left\{ \boldsymbol{P}^i(k|k) + [\hat{\boldsymbol{x}}^i(k|k) - \hat{\boldsymbol{x}}^{0j}(k|k)] \cdot [\hat{\boldsymbol{x}}^i(k|k) - \hat{\boldsymbol{x}}^{0j}(k|k)]' \right\}. \tag{2}$$

The *conditional model probabilities* $u^{i|j}(k|k)$ are given by

$$u^{i|j}(k|k) = Pr\{L^i(k)|L^j(k+1), Z^k\} = \frac{1}{u^j(k+1|k)}p_{ij}u^i(k|k), \tag{3}$$

where $Pr\{.\}$ denotes the probability of an event, Z^k is the measurement sequence up to k, and the *predicted model probability* $u^j(k+1|k)$ is computed by

$$u^j(k+1|k) = Pr\{L^j(k)|Z^k\} = \sum_{i=1}^{r} p_{ij}u^i(k|k). \tag{4}$$

The model switching process (usually Markov) is specified by the following *model transition probabilities*

$$p_{ij} = Pr\{L^j(k+1)|L^i(k)\}. \tag{5}$$

2. **Filtering**: This step corresponds to a prediction and update of the Kalman filter when the event $L^j(k+1)$ is in effect.

3. **Model probability update**: The *likelihoods* $\Lambda^j(k+1)$ (detailed further in Section 5.1) and the *predicted model probabilities* are used to compute the *updated model probability* $u^j(k+1|k+1)$ for $L^j(k+1)$:

$$u^j(k+1|k+1) = Pr\{L^j(k+1)|Z^{k+1}\} = \frac{1}{c}u^j(k+1|k)\Lambda^j(k+1), \tag{6}$$

where the normalisation factor c is

$$c = \sum_{i=1}^{r} u^i(k+1|k)\Lambda^i(k+1). \tag{7}$$

4. **Combination**: The overall state estimate $\hat{\boldsymbol{x}}(k+1|k+1)$ and the associated covariance $\boldsymbol{P}(k+1|k+1)$ are given by:

$$\hat{\boldsymbol{x}}(k+1|k+1) = \sum_{j=1}^{r} u^j(k+1|k+1)\hat{\boldsymbol{x}}^j(k+1|k+1) \tag{8}$$

$$\boldsymbol{P}(k+1|k+1) = \sum_{j=1}^{r} u^j(k+1|k+1)\cdot \left\{ \boldsymbol{P}^j(k+1|k+1) + \right.$$
$$\left. \left[\hat{\boldsymbol{x}}^j(k+1|k+1) - \hat{\boldsymbol{x}}(k+1|k+1)\right] \cdot \left[\hat{\boldsymbol{x}}^j(k+1|k+1) - \hat{\boldsymbol{x}}(k+1|k+1)\right]' \right\} \tag{9}$$

4. JV ASSIGNMENT ALGORITHM

In the case of multi-target tracking, the problem is to associate the $M(k)$ measurements (position of FPs) received at the current frame k to the $N(k-1)$ previously established tracks. Only feasible assignments are considered, that is, each track is assigned at most to one measurement and each measurement is assigned at most to one track. A measurement m, which is not assigned to any of N existing tracks, is associated with a dummy track $n = 0$. Further, a track, which is not updated with any of the M measurements, is assigned to a dummy measurement $m = 0$. Among all the feasible assignments, the most likely assignment is the one that minimises a global cost function.

Let us define cost of assigning the measurement m to the track n as a binary assignment variable $\chi(n, m)$:

$$\chi(n, m) = \left\{ \begin{array}{ll} 1 & \text{if m is assigned to n} \\ 0 & \text{otherwise,} \end{array} \right. \tag{10}$$

One can visualise an assignment as a matrix A, where the row index corresponds to the track index and the column index to the measurement index. The elements of the matrix are the binary assignment variable. For example:

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{11}$$

By using the binary assignment variables, the assignment problem is formulated as a *linear problem*:

$$min \sum_{n=0}^{N(k-1)} \sum_{m=0}^{M(k)} c(n, m)\chi(n, m), \tag{12}$$

subject to the feasibility constraints:

$$max \sum_{m=1}^{M(k)} \chi(n, m) = 1, \qquad n = 1, \cdots, N(k-1)$$
$$max \sum_{n=1}^{N(k-1)} \chi(n, m) = 1, \qquad m = 1, \cdots, M(k), \tag{13}$$

where $c(n, m)$ is the cost of assigning the measurement m to the track n.
An efficient implementation to solve the *linear problem* is given by Jonker and Volgenant (JV algorithm).[9]

5. FEATURE POINT TRACKING

A typical FP tracking system predicts current state x_k (position and speed of a FP) provided the history of measurements $Z^{k-1} = \{z(1), ..., z(k-1)\}$ and the association to the current measurement $z(k)$. Measurements are provided by a FP extraction step. For our system, we use Harris corner detector, an intensity-based operator.[19] Some tracks are hence updated, created or terminated at each frame. Track creation occurs when an extracted FP cannot be associated with any existing track. Generally, tracks are terminated when no measurement is available for continuation. Performing the track update asks for three steps: defining a validation region with the IMM filter, a photometric test on the measurements inside this region, and finally an assignment of the measurement with least cost. Track update is described in further details in the next sections.

5.1. Validation region with the IMM filter

For each track n, r validation regions ($V_{n,j}$) are estimated by the r models in the IMM filter. These validation regions provide a group of measurements that are candidates for association. The solution proposed in[20] was adopted for this purpose. More specifically, the union of all the measurements falling inside the r validation regions ($V_{n,IMM} = \{z : z \in \bigcup_{i=1}^{r} V_{n,i}\}$) is considered. This is especially useful in manoeuvreing phase, where the displacement of the feature point is relatively large. Each model j of the IMM filter is implemented with a Kalman filter. At frame k, the KF forms the predicted measurement $\hat{z}(k|k-1)$ and the associated measurement covariance matrix $S_j(k)$. The *validation region* is the region in the measurement space that fulfills:

$$V_{n,j}(k, \gamma) = \{z(k) : \nu_j'(k)S_j^{-1}(k)\nu_j(k) < \gamma\}, \tag{14}$$

where ν is the innovation (difference between measurement and predicted measurement), and γ is the *validation threshold* which determines the probability that the measurement $z(k)$ is inside the region.

5.2. Photometric Test

A photometric test is performed on the elements of $V_{n,IMM}$. For computational efficiency and assuming small changes in illumination and perspective, the *Normalised Cross Correlation* (NCC) was chosen. For each measurement $z \, \epsilon \, V_{n,IMM}$ an intensity neighbourhood of 5x5 pixels is correlated with the previous 3x3 neighbourhood of track t_n, resulting in a validated subset $V_{n,IMM+NCC}$. This is similar to an approach proposed by Cox and Hingorani.[13]

5.3. Combining IMM with JV

In order to solve the assignment problem a cost measure for each assignment is needed. As a prediction of the state is available, likelihood of a certain association measurement-to-track can be obtained. Hence, the cost $c(n,m)$ of assigning the measurement m to the track n is equal to $-\log(\Lambda_{nm})$ when $m \, \epsilon \, V_{n,IMM+NCC}$, and ∞, otherwise. The likelihood Λ_{nm} is calculated as follows. For each model j of the track n, a likelihood $\Lambda_{nm,j}(k)$ is computed. We assume that the measurement $z(k)$ conditioned on Z^{k-1} is Gaussian distributed:

$$\Lambda_{nm,j}(k) = p(z(k)|Z^{k-1}) = N[z(k); \hat{z}_j(k|k-1), S_j(k)] = N[z(k) - \hat{z}_j(k|k-1); 0, S_j(k)] = N[\nu_j(k); 0, S_j(k)], \tag{15}$$

where $p[.]$ denotes a Probability Density Function (PDF), $N[z; \mu, \Sigma]$ denotes the Gaussian PDF (with argument z, mean μ, and covariance matrix Σ), and ν is the innovation. The likelihood of each model is then weighted with the *predicted model probability* of the IMM filter.[20] The solution to the assignment problem, using these costs, produces the final unique candidate $z_{n,IMM+NCC+JV}$. Once a measurement is associated with a track, the update process of the IMM filter is carried out.

5.4. Track Continuation

The update process presented before assumes that there is at least one measurement $z_{n,IMM+NCC}$ and that the JV algorithm assigns it to a track. However, due to occlusions and image blurring (usually due to motion), one can assume situations where no measurement $z_{n,IMM+NCC+JV}$ can be found. In order to deal with such cases, a virtual measurement can be fed into the system. We use the prediction $\hat{x}(k|k-1)$ of the filter. Given the measurement model of a KF one obtains:

$$z(k) = H(k)x(k) + w(k) = H(k)\hat{x}(k|k-1) + w(k) \tag{16}$$

where $H(k)$ is the measurement matrix and $w(k)$ is a realisation of the measurement noise.

5.5. Track quality indicator

In many applications, it is useful that the multi-target tracking system provides an indication about the quality of its tracking performance. For instance in the estimation of the camera motion parameters by using the correspondence between feature points, specifically, the best tracks are selected to provide a more reliable estimate.

Let us compute, for a single KF in the IMM filter, the joint PDF of the measurement sequence up to time k $Z^k = \{z(1), \cdots, z(k)\}$ associated with a particular track:[21]

$$p[Z^k] = p[z(k), Z^{k-1}] = p[z(k)|Z^{k-1}]p[Z^{k-1}] = \prod_{i=1}^{k} p(z(i)|Z^{i-1}). \tag{17}$$

Using the *likelihood function* (15) in (17), with an explicit expression for a gaussian PDF, one has

$$p[Z^k] = \left[\prod_{i=1}^{k} \frac{1}{\sqrt{2\pi S(i)}} \right] \exp[-\frac{\lambda(k)}{2}], \tag{18}$$

where

$$\lambda(k) = \sum_{i=1}^{k} \nu'(i)S^{-1}(i)\nu(i) = \lambda(k-1) + \nu'(k)S^{-1}(k)\nu(k) \tag{19}$$

is the *modified log-likelihood function*. The individual terms $\nu'(i)S^{-1}(i)\nu(i)$ are each χ^2_M distributed with M degrees of freedom, where M is the dimension of the measurement vector $z(i)$. It follows that $\lambda(k)$ is also χ^2_{kM} distributed with kM

degrees of freedom. Using these results, a quality threshold can be established by using the χ^2_{kM} distribution and $\lambda(k)$ can serve as a track quality indicator. The track quality indicator (TQI) is obtained by weighting the *modified log-likelihood function* of each model with its *updated model probability* (6):

$$TQI(k) = \sum_{j=1}^{r} u^j(k|k)\lambda^j(k). \tag{20}$$

When TQI exceeds the quality threshold, it indicates either incorrect data association or an incorrect state estimator model. For tracks with synthetic measurement (continued tracks), another penalisation factor is introduced. The *validation threshold* γ introduced in Section 5.1 is used, forcing the worst case:

$$\lambda(k) = \lambda(k-1) + \gamma. \tag{21}$$

6. EXPERIMENTS

In order to asses the performance of our proposed technique several experiments have been conducted. Extensive evaluation on synthetic data has already been presented in our previous work.[1]

A video sequence has been captured by a camera describing a U-shaped manoeuvreing parallel to a map stuck to the wall. FPs are tracked from this sequence. In the first 113 frames, the motion is slow. During the fast manoeuvre (last 16 frames) direction and motion intensity change abruptly. The purpose of this particular motion is to show the response under both a slow motion and a fast manoeuvre.

First, the performance of the state estimator is analysed. We compare our results with KF. The IMM filter uses two constant velocity models with two different levels of process noise: for slow motion, $q_s = 3500\ pixels/sec^2$; and for fast manoeuvreing, $q_f = 27000\ pixels/sec^2$. Figure 1 shows the response to an abrupt change. We only overlap the trajectory of one track during the whole sequence for a better visualisation. The circles represent the position of the FP in the current frame. During the phase with slow motion, KF(q_s) performs better that KF(q_f). In this phase, measurements are heavily trusted by KF(q_f), resulting in noisy estimates. On the other hand, during the fast manoeuvreing, KF(q_s) looses track while KF(q_f) does not. The probabilistic switching between models enables the use of the best model according to the current motion. The visual comparison of trajectories alone is not sufficient for track evaluation. Therefore, the Root Mean Square Error (RMSE) has been selected as a scalar performance measure. It is defined as follows:

$$RMSE = \sqrt{\frac{1}{N}\sum_{k=1}^{N} \|\boldsymbol{z} - \hat{\boldsymbol{z}}(k|k)\|^2}, \tag{22}$$

where $\boldsymbol{z}$ is the true position of the feature point (extracted manually) and N the number of frames. The RMSE (between manually extracted coordinates and the estimates) is calculated at the end of the first part (where the three filters still track the FP). Table 1 presents the RMSE calculated and shows that IMM outperforms the other two filters in terms of estimation accuracy.

Filter	RMSE [pixels]
Kalman q_f	2.0600
Kalman q_s	1.2165
IMM	1.1863

Table 1. RMSE for the two KFs and IMM filter.

The second test evaluates the data association performance of the JV assignment algorithm. In this experiment, the Nearest Neighbour (NN) data association is compared with the JV algorithm. Figure 2 shows the performance of IMM-JV and IMM-NN for the oldest tracks during the slow motion phase. A miscorrelation is said to occur when a measurement from a given FP is assigned to a track that was last updated by a measurement generated by a different FP. Even though a few miscorrelations occur in IMM-JV, the proposed solution shows, globally, more robustness in front of miscorrelations than IMM-NN. See for instance, the black circle in Figure 2(a), where IMM-NN merges two FPs in a single track while IMM-JV distinguishes them (Figure 2(b)).
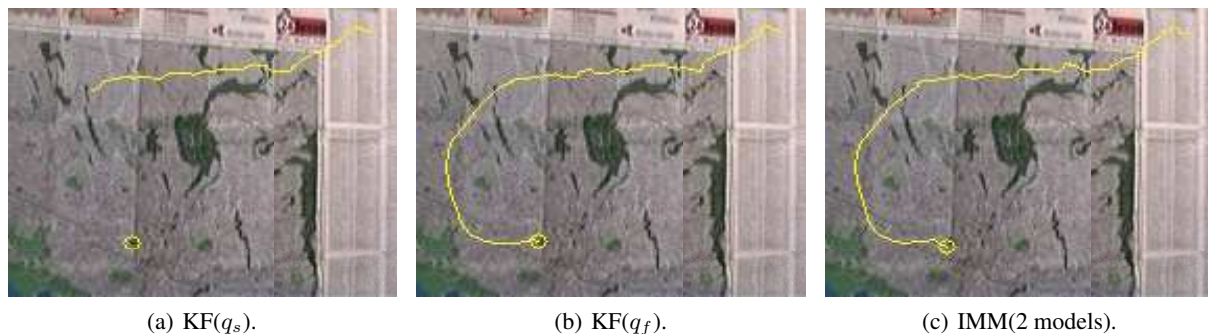
(a) KF(q_s).   (b) KF(q_f).   (c) IMM(2 models).

Figure 1. Performance of different techniques tracking a single FP changing direction and motion intensity abruptly.
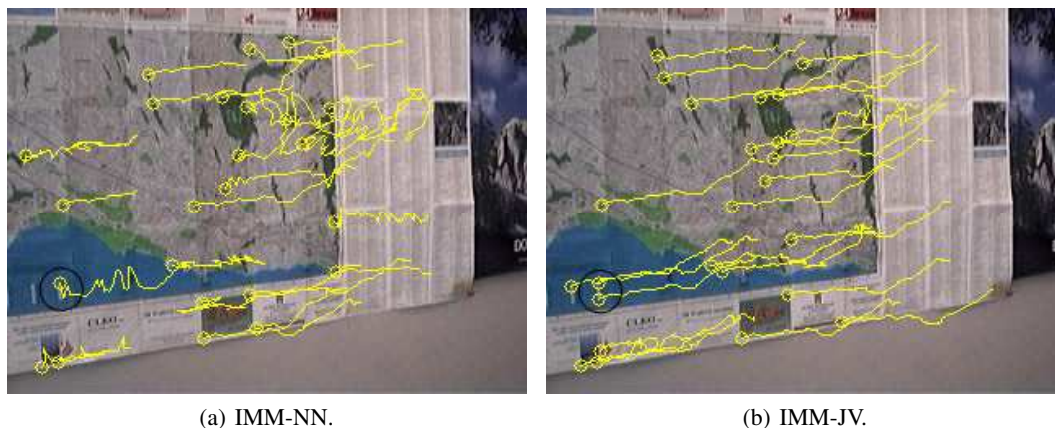

(a) IMM-NN.   (b) IMM-JV.

Figure 2. Data association performance comparison.

7. CONCLUSIONS

A novel approach to feature point tracking in real sequences has been presented. The technique presented deals with closely located FPs, and with FP describing manoeuvres. The results show that the combination of the IMM filter with the JV assignment algorithm performs better than state-of-the-art techniques, such as single model KF combined with NN. A framework for this combination in visual tracking has been detailed, together with track management techniques applied to this combination. In particular, a track continuation method and a track quality indicator have been presented.

Future research will focus on different velocity and acceleration motion models in the IMM filter instead of different process noise levels. Another axis of investigation will concentrate on dynamic tuning of hyper-parameters (variance of process and measurement noise). The performance of the proposed track management method will also be tested on real sequences.

REFERENCES

1. Y. Abdeljaoued, *Feature point extraction and tracking for video summarization and manipulation*. PhD thesis, EPFL, 2001.
2. U. Neumann and S. You, "Natural feature tracking for augmented reality," *IEEE Transactions on Multimedia* **1**, pp. 53–64, Mar 1999.

3. M. Irani, P. Anandan, J. Bergen, R. Kumar, and S. Hsu, "Mosaic representations of video sequences and their applications," *Signal Processing* **8**, pp. 327–351, May 1996.

4. I. K. Sethi and R. Jain, "Finding trajectories of feature points in a monocular image sequence," *IEEE Trans. PAMI* **9**, pp. 56–73, January 1987.

5. C. J. Veenman, M. J. T. Reinders, and E. Backer, "Resolving motion correspondence for densely moving points," *IEEE Trans. PAMI* **23**, pp. 54–72, January 2001.

6. H. A. P. Blom and Y. Bar-Shalom, "The interacting multiple model algorithm for systems with markovian switching systems," *IEEE Trans. Automatic Control Systems* **33**, pp. 780–783, August 1988.

7. R. Jonker and A. Volgenant, "A shortest augmenting path algorithm for dense and sparse linear assignment problems," *Computing* **38**, pp. 325–340, 1987.

8. H. W. Kuhn, "The Hungarian method for the assignment and transportation problems," *Naval Research Logistics Quarterly* **2**, pp. 83–97, 1955.

9. M. Yeddanapudi, Y. Bar-Shalom, and K. R. Pattipati, "IMM estimation for multisensor air traffic surveillance," *Proceedings of the IEEE* **85**, pp. 80–94, Jan 1997.

10. V. Salari and I. K. Sethi, "Feature point correspondence in the presence of occlusion," *IEEE Trans. PAMI* **12**, pp. 97–91, Jan 1990.

11. D. Chetverikov and J. Verestoy, "Feature point tracking for incomplete trajectories," *Computing* **62**, pp. 321–338, 1999.

12. Y. Yao and R. Chellappa, "Feature correspondence using probabilistic data association," in *Proc. Int. Conf. Acoustics, Speech, and Signal Processing*, pp. 157–160, 1993.

13. I. Cox and S. Hingorani, "An efficient implementation and evaluation of reid's multiple hypothesis tracking algorithm for visual tracking," in *ICPR*, pp. A:437–442, 1994.

14. D. Reid, "An algorithm for tracking multiple targets," *IEEE Trans. Automatic Control* **24**, pp. 423–432, Dec 1979.

15. P. Tissainayagam and D. Suter, "Visual tracking and motion determination using the IMM algorithm," in *ICPR*, pp. 289–291, 1998.

16. D. G. Lowe, "Object recognition from local scale-invariant features," in *International Conference on Computer Vision (ICCV)*, pp. 1150–1157, September 1999.

17. V. Lepetit, J. Pilet, and P. Fua, "Point matching as a classification problem for fast and robust object pose estimation," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2004.

18. Y. Bar-Shalom and X. R. Li, *Estimation and Tracking: Principles, Techniques and Software*, Artech House, 1993.

19. C. Harris and M. Stephens, "A combined corner and edge detector," in *Alvey Vision Conf.*, pp. 147–151, 1988.

20. Y. Bar-Shalom and X. R. Li, *Multitarget-Multisensor Tracking: Principles and Techniques*, YBS Publishing, 1995.

21. Y. Bar-Shalom and T. E. Fortmann, *Tracking and Data Association*, Academic Press, 1988.