Tracking of Object with SVM Regression

Weiyu Zhu, Song Wang, Ruei-Sung Lin, Stephen Levinson
Dept. of Electrical and Computer Engineering, Univ. of Illinois at Urbana-Champaign
Beckman Institute, 405 N. Mathews, Urbana, IL 61801, USA

Abstract

This paper presents a novel feature-matching based approach for rigid object tracking. The proposed method models the tracking problem as discovering the affine transforms of object images between frames according to the extracted feature correspondences. False feature matches (outliers) are automatically detected and removed with a new SVM regression technique, where outliers are iteratively identified as support vectors with the gradually decreased insensitive margin ε. This method, in addition to object tracking, can also be used for general feature-based epipolar constraint estimation, in which it can quickly detect outliers even if they make up, in theory, over 50% of the whole data. We have applied the proposed method to track real objects under cluttering backgrounds with very encouraging results.

1 Introduction

Tracking 3-D objects in real scenes is a challenging computer vision problem. Traditional tracking methods usually work under the assumption that the time-intervals between frames are short enough so that the motion of object images is highly predictable [1][2] or the object location in images is bounded in a small region [3][4]. Unfortunately, this assumption does not hold in many applications. For example, the robot we built is required to constantly collect data from its large-numbered peripheral sensors and perform actions from now and then. Consequently, high video frame rates can hardly be guaranteed due to the limitations of the communication bandwidth and the mechanical delays of the actuators and sensors.

Feature-matching based method offers a way to track objects with large and unpredictable image changes over frames. A basic idea of feature-matching based tracking is to discover the image transforming properties in consecutive frames according to the detected feature correspondences. In order to obtain a valid estimate of the transforms, correct detection of feature correspondences is essential, which is, however, not easy in practice due to three factors: (1) the similarities, such as similar intensities and shapes, shared among features; (2) the occlusions and (3) the noise, which might also drive data away from where they should be. The uncertainties in feature matching impose great challenges on the estimation of transforming properties. Therefore, developing a robust feature matching technique is especially important.

A straightforward strategy for robust feature matching is to utilize the geometric constraints among features. Specifically, a set of candidate matches for each feature in the template is detected in the target image and a corresponding matching likelihood is computed for each candidate match according to some geometric constraints among the neighboring features. Then, the candidate matches are "relaxed" according to a certain updating algorithm, such as the well known *winner-take-all*, as used in [5] and [6], *loser-take-nothing* [7], and *some-winners-take-all* [8]. The geometric-constraint based methods are usually quite computationally expensive since each pair of the potential matches needs to be inspected independently. In addition, this method performs poorly when the constraint information is incomplete or incorrect, such as when occlusion exists.

More advanced robust matching technique includes the so-called *regression diagnostic* method [9] and *robust regression* method [10]. The regression diagnostic method detects false matches, called the *outliers*, by measuring the influence on the solution when the corresponding data are removed. Shapiro *et al.* and Torr *et al.* exploited this scheme for epipolar geometry estimation in [11] and [12], respectively, where false matches were detected if their removal would largely reduce the regression residuals. The regression diagnostic scheme works well when the percentage of outliers is small, which might not be met if the difference between images is large.

To cope with large percentages of outliers, a number of the so-called *robust regression* methods have been proposed, among which the Maximum Likelihood Type Estimators (M-estimators) [13], Least Median of Squares (LMS) [10] and Random Sample Consensus (RANSAC) [14] are the most commonly used. The M-estimator, which was used by Huber [13] and Olsen [15], works in a similar way as the least squared method except that it replaces the squared residuals by some other functions for suppressing the effect of the data far away from the underlined fitting

0-7695-1272-0/01 $10.00 © 2001 IEEE

functions. The LMS method, unlike the M-estimators, works against noise by minimizing the median of the squared residuals over the data set. Zhang *et al.* [8] have used this method for fundamental matrix estimation, in which outliers were allowed to make up as much as 50% of the data without jeopardizing the correct result. Both the M-estimators and LMS method are powerful for outlier detection in theory; however, their solutions are generally hard to obtain due to the nonlinear nature (true for M-estimators if the residual function is non-linear) of the formulated optimization problem. Practically, these methods are usually implemented in approximate forms with the aid of the RANSAC method.

RANSAC is a Monte Carlo typed statistical method for parameter estimation. In RANSAC, samples are randomly selected from the data set and used for parameter estimation. The results are then evaluated with the entire data set according to a certain criterion, such as the least mean squared error metric, and the one with the best performance (or exceeding a given threshold) is selected as the final estimate. The RANSAC method offers a generic framework for assisting other methods, such as the M-estimators and LMS method, to solve, in statistics, nonlinear optimization problems that cannot be solved in closed forms.

In this paper, we present a novel robust feature-matching method, in which false feature matches are effectively detected and removed with a special iterative Support Vector Machine (SVM) regression technique. Compared with the approaches in literature, our method possesses two advantages: (1) Low computational cost, which is similar to solving a linear programming problem, and (2) Large tolerance for outliers, which are allowed to account for, theoretically, over 50% of the whole data while the underlined function could still be discovered. The proposed method can be used for general epipolar constraint estimation or tracking of a rigid object in an image sequence, in which the tracking problem, as will be described in the next section, is modeled as discovering the affine transform of object images in two frames.

The rest of this paper is organized as follows. Section 2 describes the proposed method and its usage for rigid object tracking. Section 3 discusses two implementation issues: the performance analysis and computational complexity estimation. Experiments are presented in Section 4, followed by the conclusion in Section 5.

2. Method Proposed

2.1 Formulation of the object-tracking problem

For simplicity, we assume the object to be tracked "thin" enough relative to the distance to the camera so that the weak-perspective camera model is valid for object imaging. Under the weak perspective assumption, it is easy to prove that the object images in two frames would satisfy some affine transform property given the motion of the camera/object and the structure of the scene are *general* (A study of the *non-general* cases, when the epipolar geometry between two images does not exist, is given in [16]). Therefore, tracking a rigid object in an image sequence could be formulated as searching for the affine transforms of object images over frames. The extension to the general perspective camera model is straightforward, where the tracking problem could be modeled as searching for the fundamental matrix **F** in consecutive images.

The affine transform of object images is formulated as

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix} \qquad (1)$$

where the coefficients (a, b, e, c, d, f) are the affine parameters and (x', y') / (x, y) are the points in the target and template images, respectively. The proposed method first uses corner detectors to select a set of feature points in the template and searches for the correspondences in the target image with a two-stepped matching algorithm. Then, a special SVM regression technique is applied to detect and remove outliers in feature matching iteratively. Finally, the affine parameters are computed with the remaining feature correspondences by minimizing the mean squared error.

2.2 Detection of feature correspondences

The proposed tracking system uses "corners" on the object of interest as feature points, which are extracted according to the operator [17]:

$$R(x, y) = \det[C] - k \times \mathrm{trace}^2[C] \qquad (2)$$

where k is a constant and the matrix C is defined as

$$C = \begin{bmatrix} \sum E_x^2 & \sum E_x E_y \\ \sum E_x E_y & \sum E_y^2 \end{bmatrix}$$

where $[E_x, E_y]$ are the spatial image gradients on the x and y directions and the sums are taken over the neighborhood of the point to be checked. The operator R in (2) is computed over all corner candidates, which are heuristically defined as all edge points obtained with the Candy edge detector [18], and thresholded for feature selection.

Feature matching is done by first applying the same corner detector to extract candidate features in the target image. Then, a maximum-likelihood edge template matching technique, proposed by Olson [19], is exploited to find candidate matches based on the similarities of edge distribution in the windows centered at the feature points in the template and the candidate correspondences in the target

image. The matching likelihood of any feature pair $p_{template}$ and p_{target} is defined as

$$L(p_{template}, p_{target}) = \left(\prod_{i=1}^{n} \left(\frac{1}{2\pi\sigma^2} e^{-(d_i^2/2\sigma^2)} \right) \right)^{1/n} \quad (3)$$

where n is the number of edge points in the window and d_i is the Euclidean distance between the edge point i in the template and its nearest neighbor in the target.

The maximum-likelihood template matching method is capable of capturing shape similarities at two locations given the only possible image transforms are translations and scaling with small magnitudes. Combined with intensity correlation, the second step in feature matching, the best correspondences are selected in the sense of both shape similar and intensity agreeing.

2.3 Outlier-detection with SVM regression

Although both shape and intensity similarities are considered in our feature matching method, false matches are still unavoidable in practice, which requires a further outlier-detection mechanism.

Given the set of feature points $\mathbf{U}=\{\mathbf{u}_1, \ldots \mathbf{u}_N\}$ in the template and the matches $\mathbf{V}=\{\mathbf{v}_1, \ldots \mathbf{v}_N\}$ in the target, the tracking problem, according to Section 2.1, is formulated as searching for an affine transform $\mathbf{t}$ where $\mathbf{t}(\mathbf{u}_i)$ maximally approximates $\mathbf{v}_i$ by minimizing a certain loss function $Q(\mathbf{v}_i, \mathbf{t}(\mathbf{u}_i))$. Since false matches might exist in feature detection, we expect the optimal regression function may have large gaps at some data, which are called the *outliers*, given the correct correspondences are dominant in the data set. The proposed outlier detection algorithm is inspired from this very observation. Specifically, the loss function $Q(\cdot, \cdot)$ is defined as the linear ε-insensitive function used in the SVM technique [20], which is given by

$$\left\| \mathbf{v}_i - \mathbf{t}(\mathbf{u}_i) \right\|_\varepsilon = \begin{cases} 0 & \text{if } \left\| \mathbf{v}_i - \mathbf{t}(\mathbf{u}_i) \right\| \le \varepsilon \\ \left\| \mathbf{v}_i - \mathbf{t}(\mathbf{u}_i) \right\| - \varepsilon & \text{else.} \end{cases} \quad (4)$$

The ε-insensitive function offers several benefits for outlier detection: (a) It does not prefer eliminating large regression errors; therefore, the gaps between the outliers and the regression function would be salient and easy to detect. (b) With the insensitive margin ε, regression data can be classified into support vectors or non-support vectors, which offers a convenient way to distinguish the outliers from the normal data.

Refer to (1), the affine parameters can be represented as a 6-dimensional vector $\mathbf{w}=(a, b, e, c, d, f)^T$. Denote $\hat{\mathbf{v}} = (x'_1, y'_1, \cdots x'_N, y'_N)^T$ and $\hat{\mathbf{u}} = (u_{x1}, u_{y1} \cdots u_{xN}, u_{yN})^T$, where $u_{xi} = [x_i, y_i, 1, 0, 0, 0]$ and $u_{yi} = [0, 0, 0, x_i, y_i, 1]$, the transform $\mathbf{t}$ can be reorganized as a one-dimensional function $\mathrm{t}(\mathbf{u}) = \mathbf{w}^T \hat{\mathbf{u}}$. Consequently, the optimization problem is formulated as a general SVM regression problem

$$\min_{\mathbf{w}} \sum_{i=1}^{2N} \left\| \hat{\mathbf{v}}_i - \mathbf{w}^T \hat{\mathbf{u}}_i \right\|_\varepsilon \quad (5)$$

where N is the number of feature correspondences.

Applying the standard SVM technique [20], (5) can be rewritten as

$$\min_{\xi, \hat{\xi}} \Phi(\xi, \hat{\xi}) = \sum_{i=1}^{2N} (\xi_i + \hat{\xi}_i) \quad (6)$$

subject to

$$\begin{aligned} \hat{\mathbf{v}}_i - \mathbf{w}^T \hat{\mathbf{u}}_i &\le \varepsilon + \hat{\xi}_i \quad i = 1,2,\cdots,2N \\ -\hat{\mathbf{v}}_i + \mathbf{w}^T \hat{\mathbf{u}}_i &\le \varepsilon + \xi_i \quad i = 1,2,\cdots,2N \\ \xi_i, \hat{\xi}_i &\ge 0 \quad i = 1,2,\cdots,2N \end{aligned} \quad (7)$$

Denote $\mathbf{z} = (\xi, \hat{\xi}, \mathbf{w}^T)^T$, where $\xi = (\xi_1, \xi_2, \cdots \xi_{2N})$ and $\hat{\xi} = (\hat{\xi}_1, \hat{\xi}_2, \cdots \hat{\xi}_{2N})$, the above optimization problem thus becomes a standard Linear Programming (LP) problem.

By solving the derived LP problem, each feature datum is assigned with a pair of auxiliary values ξ_i and $\hat{\xi}_i$. A datum i is said a support vector if $\xi_i + \hat{\xi}_i > 0$ and non-support vector otherwise. Support vectors indicate the given datum far away from the regression function while non-support vectors means the datum accurate enough with the permissible error margin ε. A feature match (x', y') in the target image is said an outlier if the regression datum corresponding to the x' or y' becomes a support vector. Outliers are then removed from the data set (instead of being replaced by other matching candidates as we did in medical image segmentation [21]) and the regression process is repeated with the remaining matching pairs. The iteration continues with a decreased insensitive margin ε [21] until no outliers are detected any more, when the transform parameters are computed with the remaining feature correspondences by minimizing the mean squared errors.

2.4 Summary of the algorithm

The proposed algorithm for rigid object tracking in an image sequence is summarized below.

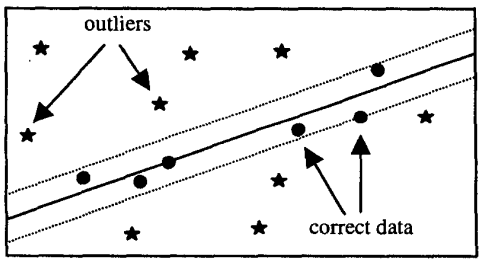
1. Extract features (corners) from the template and find the correspondences in the target image.
2. Perform SVM regression to find outliers according to the current ε-insensitive margin.

3. Remove outliers and repeat the step 2 with a reduced ε. The iteration terminates when ε reaches a predefined small value, such as 1.0.
4. Compute the affine parameter set with the remaining data by minimizing the mean squared error.
5. Apply the transform on the template image and obtain the predicted object location in the target image. The predicted object contour is then adjusted to fit the edge boundaries in the target image by matching with the nearest edge points in the target edge image. The resulting contour then serves as the new template to process the image in the next frame.

3. Discussions

3.1 Performance analysis

The role of the SVM regression process in the proposed method is to extract the data far away from the underlined fitting function, which are identified as support vectors. We claim that, supposing the outliers are sparsely distributed in the feature space, i.e., they themselves would not construct a "very likely" fitting function; the proposed algorithm is guaranteed to find the underlined function even when the outliers, which are assumed to distribute uniformly, exceeds 50% of total data.



(a)

Figure 1. Illustration of outlier detection. The underlined function is represented by the solid line and the ε-insensitive margin resulted from the SVM regression is delineated by the dash lines. The data outside the margin are support vectors.

A qualitative performance analysis of outlier detection in 1-dimensional case is given in Figure 1, where the correct data are drawn in dots and outliers are in stars. Given the outliers are distributed uniformly, i.e., they are distributed almost equally on both side of the underlined fitting function, the ε-insensitive function successfully catches all outliers while keeps the correct data untouched. The results shown in the figure is by no means artificial, since the minimizing of the ε-insensitive function has a property of trying to even up (or differ by one) the number of data outside the insensitive margins provided that the outliers are sparsely distributed in the feature space. Proving of this property could be straightforwardly done by shifting the insensitive margin up and down in the figure and observing the changes of the lost function. The extension to higher dimensional cases is similar.

3.2 Computational complexity estimation

Denote the number of the extracted features in the template image as M and the number of corners in the target image as N, where generally $M<N<<$number of image pixels, the computational cost for feature matching is $O(MN)$ operations of window-based edge/intensity correlations. The SVM regression process consists of a small number (usually <10) of linear programming operations, which does not cost a lot in general. Consequently, the total expense for object tracking in two images could be roughly computed as

$$Cost = 2\left(C_{edge\_det} + C_{corner\_det}\right) + MN \times C_{window\_corr} + nC_{lp} \qquad (8)$$

Since most major computations in (8) (Except the item nC_{lp}) can be implemented in parallel, the requirement of real-time processing is easy to meet.

4. Experiments

We have used the proposed method for rigid object tracking in real scenes. Figure 2 shows the performance of outlier detection with the proposed SVM regression method. The cup contour in the template image was manually extracted with 32 feature points detected. Figure 2(b-1) shows the target image with the detected feature correspondences overlaid, from which we see the existence of incorrect matches. By applying the proposed SVM regression method, all outliers were removed and only 19 correct matches were survived (Figure c-1). The corresponding object contours obtained by applying the transforms derived are given in Figure b-2 and c-2, respectively. Although the contour difference shown is not quite salient, it is, however, crucial for the last step of the itemized algorithm in Section 2.4, in which the result in Figure b-2 would probably lose the track of the cup handle.

Figure 3 displays the same experiment as that in Figure 2, except that an additional 50 pairs of synthesized correspondences were added for the purpose of testing the robustness of the proposed method. The synthesized feature correspondences (marked as "o") were generated randomly with a standard deviation of 20 pixels to the correct data. Although the percentage of outliers is quite large (≈70%) in this experiment, the proposed method was still able to, as shown in Figure 3 (c), stamp all incorrect data out while keep most correct ones saved.

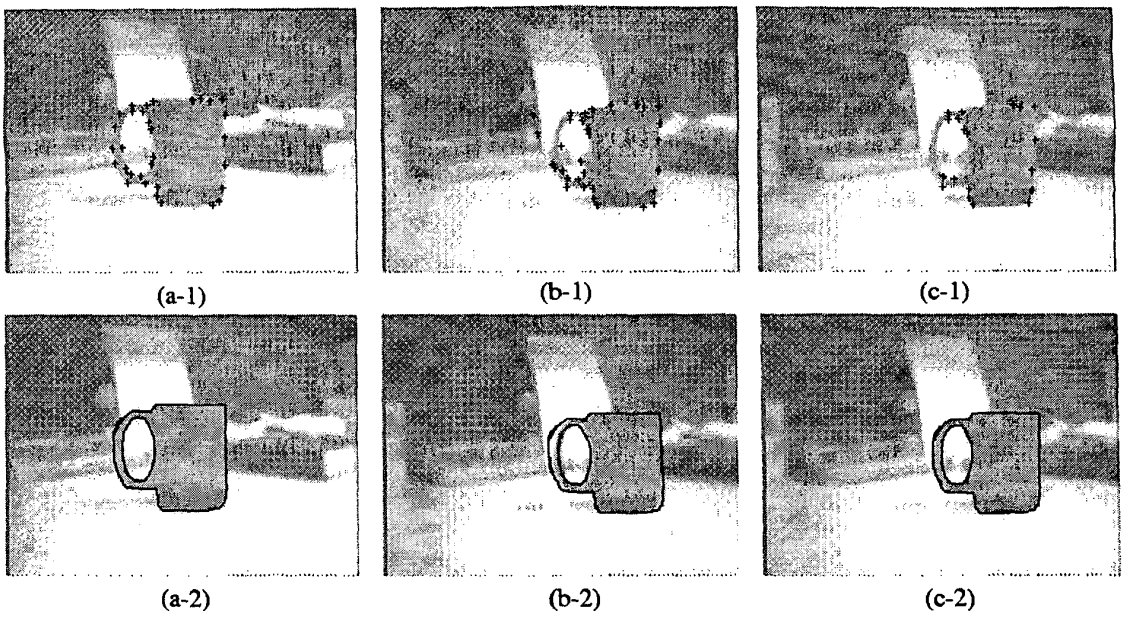


Figure 2. (a-1) Template image with extracted features; (b-1, c-1) Target image with detected feature correspondences before and after the SVM regression applied; (a-2) Template image with object contour overlaid; (b-2, c-2) Target image with contours computed from the feature matches in (b-1) and (c-1), respectively.

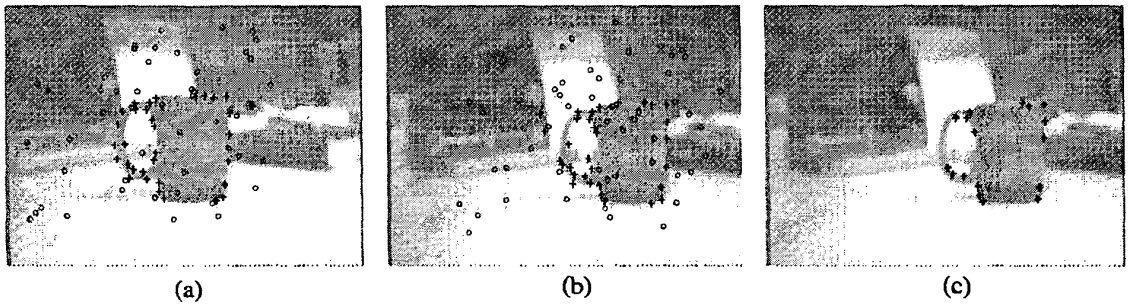


Figure 3. Detection of outliers with 50 randomly generated feature correspondences imposed.

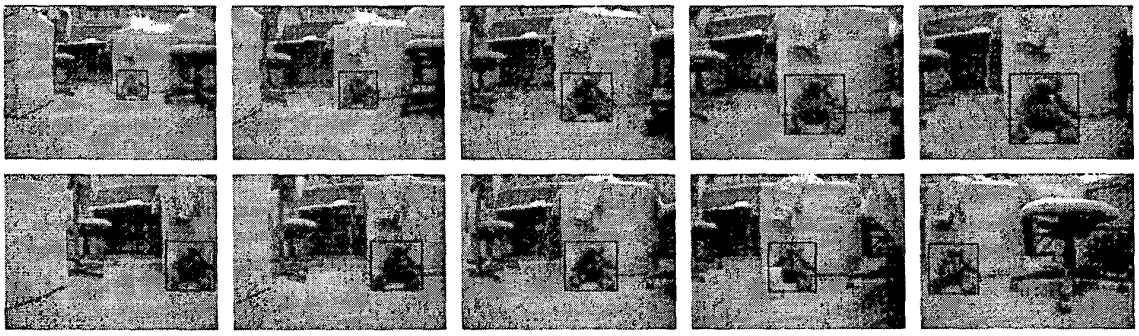


Figure 4. Tracking of the bear images in a frame sequence. Pictures in the first and second rows were taken when the robot was "going forward" and "turning right", respectively

The last experiment conducted was to track a toy bear in images when the robot, on which a monochrome camera was mounted, was moving. The image-sampling rate was about 2 frames/second with a resolution of 320×240 in pixels. Figure 4 displays 10 selected frames with the tracked portions marked in rectangles. The contour of the toy bear in the top-left image was manually extracted and the features were searched within the sketched rectangle, which is defined as the "tightest" rectangle embracing the bear contour. Starting from the template frame, the bear contour in the next frame was determined with the proposed tracking method. Then, the new contour and the corresponding rectangular area in that frame is serve as the new template to process its next frame, and so on. Although the bear image undergoes great changes in size and position in the sequence, the target location was still tracked very well even when occlusion existed.

5. Conclusion

A novel feature matching-based object-tracking algorithm is presented in this paper. The tracking problem is modeled as searching for the affine transform of object images in two frames with the assumption of weak perspective camera model. Feature correspondences are detected with a two-stepped correlation method, in which both shape and intensity information is exploited for matching. False feature matches are detected and removed with a new iterative SVM regression technique, where outliers are easily identified if the corresponding data become support vectors in the regression. We have shown that, in both theoretical analysis and experimental test, this method is able to detect uniformly distributed outliers even if they make up over 50% of the data. The proposed method has been used to track a real object under cluttering background and very encouraging results were achieved even when occlusions existed in some frames.

Acknowledgement

This project is partly supported by the NSF Grant with award number IIS-00-85980. The authors would like to acknowledge the anonymous reviewers for their invaluable comment and suggestion.

References

[1] C. Harris, "Tracking with rigid models", *Active Vision*, MIT press, pp.59-74, 1992.

[2] L.H. Matthies, T. Kanade and R. Szeliski, "Kalman filter-based algorithms for estimating depth from image sequences", *International Journal of Computer Vision, Vol. 3*, pp. 209-236, 1989

[3] M. Kass, A. Witkin and D. Terzopoulos, "Snakes: active contour models", *Proceedings of International Conference on Computer Vision*. pp. 259-268, 1987.

[4] L. Chun, Y. Shiu. "An unbiased active contour algorithm for object tracking", *Pattern Recognition Letters*, pp.491-8, 1998

[5] A. Rosenfeld, R. Hummel and S. Zucker, "Scene labeling by relaxation operations", *IEEE Transactions on Systems, Man, & Cybernetics. Vol. 6, no. 4*, pp. 420-33, 1976

[6] S. Zucker, Y. Leclerc and J. Mohammed, "continuous relaxation and local maxima selection: conditions for equivalence", *IEEE Transactions on Pattern Analysis & Machine Intelligence, vol. 3*, pp. 117-27, 1981

[7] S. Li, "Inexact matching of 3D surfaces", *technical Report VSSP-TR-3/90*, University of Surrey, Guildford, UK.

[8] Z. Zhang, R. Deriche and O. Faugeras, "A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry", *Artificial Intelligence, vol. 78*, pp. 87-119, 1995

[9] R.D. Cook and S. Weisberg, "Characterizations of an empirical influence function for detecting influential cases in regression", *Technometrics, Vol 22*, pp. 337-44, 1980

[10] P. Rousseeuw and A. Leroy, *robust Regression and Outlier Detection*, John Wiley and Sons, 1987

[11] L. Shapiro and M. Brady, "Rejecting outliers and estimating errors in an orthogonal regression framework", *Technical Reports OUEL 1974/93*, University of Oxford, 1993

[12] P.H.S. Torr and D.W. Murray, "Outlier Detection and Motion Segmentation", *Spie–the International Society for Optical Engineering, Vol 2069*. pp. 432-43, 1993

[13] P. Huber, *Robust Statistics*, John Wiley and Sons, 1981

[14] M.A. Fischler, R.C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography", *Communications of the ACM, vol.24, no.6*, pp.381-95. 1981

[15] S. Olsen, 'Epipolar line estimation", *Proceedings of Second European conference on Computer Vision*, Santa Margherita Ligure, Italy, pp 307-11, 1992

[16] P. Torr, A.W. Fitzgibbon and A. Zisserman. "Maintaining multiple motion model hypotheses over many views to recover matching and structure", *Sixth International Conference on Computer Vision*, pp.485-91, 1998

[17] C. Harris and M. Stephens, "A combined corner and edge detector", *Proceedings Alvey Conference*. pp. 189-92, 1988

[18] J. Canny, "A computational approach to edge detection", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 8, pp. 679-98, 1986

[19] C.F. Olson, "Maximum-Likelihood Template Matching", *Proceedings of the International Conference on Computer Vision and Pattern Recognition 2000, vol. 2*, pp. 52-57, 2000.

[20] V.N. Vapnik, *The Nature of Statistical Learning Theory*, New York: Springer-Verlag, 2000

[21] S. Wang, W. Zhu & Z.-P. Liang, "Shape deformation: SVM regression and application to medical image segmentation", *IEEE International Conference on Computer Vision*, Vol. 2, pp. 209-216, Vancouver, Canada, July 9-12, 2001