A Projective Framework for Scene Segmentation in the Presence of Moving Objects*

David Demirdjian and Radu Horaud
GRAVIR-IMAG, INRIA Rhône-Alpes
655, avenue de l'Europe
38330 Montbonnot St.Martin, France
David.Demirdjian@inrialpes.fr Radu.Horaud@inrialpes.fr
http://www.inrialpes.fr/movi/people/Demirdjian/index_frame.html

Abstract

Given a sequence of pairs of images gathered with an uncalibrated stereo camera pair and given a set of point-to-point correspondences between these image pairs, we describe a method that segments the observed scene into static and moving objects while it rejects badly matched points. Unlike many approaches which were suggested in the past, the method allows for both motion of the camera pair (egomotion) and non rigid scenes (scenes composed of static objects as well as objects undergoing various motions).

First we establish the projective framework enabling us to characterize rigid motion in projective space. Second we use this characterization in conjunction with a robust estimation technique to determine egomotion. Third we describe a method based on data classification which further considers the non-static scene points and groups them into several moving objects. Finally we show some preliminary experiments involving a moving stereo head observing both static and moving objects.

1 Introduction and motivation

Detection of moving objects is one of the major areas in image sequence analysis. It is necessary for telesurveillance, autonomous system navigation and many other applications. Achieving motion analysis is a difficult task especially when moving objects are not rigid and when the observer is moving as well.

Motion-based segmentation can be classified into two major categories: 2D optical flow segmentation and 3D motion segmentation. The first approaches [7] consist of segmenting a scene using techniques that group parts of the image corresponding to similar velocities. Ignoring 3D geometry, these methods can deal with non rigid objects but often give spurious segmentation in the case of non trivial scenes.

On the contrary, 3D approaches use a geometric characterization of the motion in order to segment a scene. For example, in [8] and [5], this characterization is explicit (modelized by a rigid motion), but in many other approaches, the modelization is implicit. Thus, when two consecutive images are involved, motions are associated with an essential or a fundamental matrix which encapsulates motion parameters [9]. These methods generally give limited results because ambiguous parameterizations are involved. Many researchers prefer then to use three or more images. For example, [3] uses a parallax shape-based constraint which enables to retrieve independent motions over three frames. In [6], a fine parametric model for the optical flow enables to robustly compute the dominant motion in images.

Most existing methods consider monocular systems and it seems to us that we are the first to use an uncalibrated sequence of pairs of images for motion segmentation (motion-stereo). Furthermore, the problem with existing methods is that either they assumed that all objects are rigid and do not work with non rigid bodies (such as persons), or they make no such hypothesis in which case, all moving objects are roughly considered as a unique entity. The work we present here bridges these gaps since it addresses the problem of motion segmentation for a moving uncalibrated stereo rig with a framework which deals with non rigid bodies.

*The work reported was sponsored by the ESPRIT-IV project LTR 26247 VIGOR

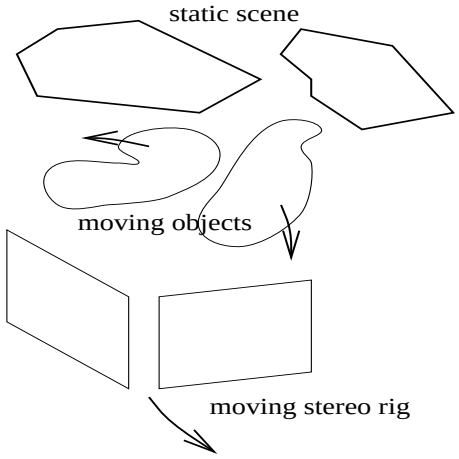

Figure 1: The approach: a moving stereo rig observing both a static scene and moving objects

2 Outline of the approach

In this paper, we will distinguish two categories of image motion: *egomotion* and *motion induced by independent objects*. *Egomotion* can be defined as the motion due to an observer moving around a static and rigid environment. *Motion induced by independent objects* is the image motion created by the relative motion of objects with respect to the static ones.

Contrary to many approaches, we will differently treat these 2 aspects of the motion. *Egomotion* will be estimated using a 3D projective model (section 3 and section 4) and computed as the dominant motion in images (section 5). Points not conforming to the obtained dominant motion will be considered as belonging to independent objects that will be retrieved using an image-based *hierarchical clustering* algorithm (section 6). In section 7 the complete algorithm is applied to a real stereo sequence and discussed in section 8.

3 Projective reconstruction

Given a pair of images gathered with an uncalibrated stereo camera pair, it is well known that the fundamental matrix describing the epipolar geometry can be recovered. From this matrix it is further possible to determine two projection matrices $\mathbf{P}_x$ and $\mathbf{P}'_x$ which verify:

$$\boldsymbol{x} \simeq \mathbf{P}_x \boldsymbol{X} \text{ and } \boldsymbol{x}' \simeq \mathbf{P}'_x \boldsymbol{X} \tag{1}$$

where $\boldsymbol{X}$ denotes the projective coordinates of a 3-D point M in a 3-D projective basis $\mathcal{B}_x$. $\boldsymbol{x}$ and $\boldsymbol{x}'$ denote the projective coordinates of the projection of M onto the left and right images, i.e., the projective coordinates of two image points m and m'. $\simeq$ denotes projective equality (up to an unknown scale factor).

Moreover, it is possible to linearly solve eqs. (1) and determine the 4-vector $\boldsymbol{X}$ which represents the projective coordinates of point M. Notice that in order to solve these equations, numerical values for both $\boldsymbol{x}$ and $\boldsymbol{x}'$ are necessary. These two 3-vectors are given by $\boldsymbol{x}^\top = (\overline{\boldsymbol{x}}^\top \; 1)$ and $\boldsymbol{x}'^\top = (\overline{\boldsymbol{x}}'^\top \; 1)$ where $\overline{\boldsymbol{x}}$ denotes the pixel coordinates of point m.

4 Projective motion

We consider now two different positions of the same stereo camera pair before and after a motion. The camera pair observes the same physical point M. The projective reconstruction before the motion has just been previously described. The projective reconstruction after the motion can be easily obtained from the following equations:

$$\boldsymbol{y} \simeq \mathbf{P}_y \boldsymbol{Y} \text{ and } \boldsymbol{y}' \simeq \mathbf{P}'_y \boldsymbol{Y} \tag{2}$$

where $\mathbf{P}_y$ and $\mathbf{P}'_y$ are the projection matrices associated with the second stereo pair configuration, $\boldsymbol{Y}$ denotes the projective coordinates of M in the projective basis $\mathcal{B}_y$, and $\boldsymbol{y}$ and $\boldsymbol{y}'$ denote the projective coordinates associated with the second image pair.

The relationship between $\boldsymbol{X}$ and $\boldsymbol{Y}$ becomes:

$$\mu \boldsymbol{Y} = \mathbf{H} \boldsymbol{X} \tag{3}$$

where $\mathbf{H}$ is a 4×4 full rank matrix representing a *projective transformation* of the 3-D projective space and which is called a homography. Such a homography is defined up to a scale factor and therefore it has 15 degrees of freedom associated with it. After eliminating the scale factor μ, eq. (3) provides 3 linear constraints in the entries of $\mathbf{H}$ (see [2]).

Next we consider a *rigid 3-D scene* composed of m points M_1 through M_m. Let $\boldsymbol{X}_1$, $\boldsymbol{X}_2$, ..., $\boldsymbol{X}_m$ and $\boldsymbol{Y}_1$, $\boldsymbol{Y}_2$, ..., $\boldsymbol{Y}_m$ be their projective representations in the projective bases $\mathcal{B}_x$ and $\mathcal{B}_y$ respectively. With $m = 5$ such points in general position we obtain 15 linear independent equations which can be solved to determine $\mathbf{H}$.

Therefore, matrix $\mathbf{H}$ can be interpreted as the change of projective basis from $\mathcal{B}_x$ to $\mathcal{B}_y$. A second possible interpretation of this matrix is a projective representation of the motion undergone by the camera pair – *projective motion*:

Definition 1 *Consider an uncalibrated stereo camera pair which observes a 3-D rigid scene while it moves. The projective transformation between two projective reconstructions of the same 3-D scene obtained before and after the motion is called projective motion.*

5 Robust estimation of projective motion

In order to estimate projective motion one may consider eq. (3) for $m \geq 5$ point correspondences. Therefore we obtain $3m$ linear equations which can be solved to determine the entries of $\mathbf{H}$ (remember that there are 3 equations for each match). However, such a linear estimation method has two major drawbacks:

1. the method can deal neither with outliers (mismatched and/or mistracked points) nor with non-rigid scenes (scenes that contain both static and moving objects), and
2. the method minimizes an algebraic distance and hence it gives poor results for badly conditioned data. In particular, for $m = 5$ the method is very sensitive to noise.

To overcome these two drawbacks we introduce a new method based on robust estimation on the one side and on minimizing an Euclidean error on the other side.

5.1 Robust methods in computer vision

Robust regression methods are widely used to solve various vision problems such as estimation of epipolar geometry [11], estimation of the trifocal tensor [10] and so forth. Commonly used robust methods are M-estimators, least-median-squares (LMedS), and random sample consensus (RANSAC) [1].

We wish to apply robust methods in order to compute projective motion in the presence of outliers and/or non static scenes and to eventually distinguish between static and moving objects. Moreover, we would like to deal with situations where only 50% of the points composing the scene belong to static objects. Therefore we must choose a robust method which tolerates up to 50% of outliers. This immediately rules out the M-estimators method which, in our case, will tolerate only up to 7% of outliers.

Therefore we are left with LMedS and RANSAC. At first glance they are very similar. Data subsets are selected by a random sampling process. For each such subset a solution is computed and a criterion must be estimated over the entire data set. The solution yielding the best criterion is finally kept. LMedS minimizes the median of the squares of the errors while RANSAC maximizes the number of inliers. Even if the criteria used by these two methods are quite different, in most practical applications, comparable results are obtained with both methods. The main difference between LMedS and RANSAC resides in the outlier rejection strategy being used. The user must supply RANSAC with a threshold value (which can be computed automatically) while LMedS does not require such a threshold. This feature enables RANSAC (i) to be more efficient in the presence of non homogeneous noise, (ii) to allow for 50% outliers and above, and (iii) to be more efficient because it can quit the random sampling loop as soon as a consistent solution is found.

When applied to the problem of estimating projective motion, the RANSAC method can be summarized as follows:

1. For each sample k, $1 \leq k \leq N$ execute the following loop:
   - 1.1 Randomly select 5 matches among the m matched points between the two stereo image pairs;
   - 1.2 Estimate a homography $\mathbf{H}_k$ from these 5 matches;
   - 1.3 Compute the total number of matches m_k consistent with $\mathbf{H}_k$, that is, matches for which the associated error is under a threshold t_c (see section 5.2).
2. Select the homography $\mathbf{H}_k$ with the largest number of consistent matches m_k and refine the estimation of the homography using these m_k matches.
3. Update the list of inliers and outliers.

The number of samples N must be sufficiently large to guarantee that the probability of selecting a good subset is high enough, say this probability γ must satisfy $\gamma \geq 0.95$.

The theoretical expression of this probability is $\gamma = 1 - (1 - (1 - \varepsilon_{out})^p)^N$ where p is the number of points that are necessary to compute a solution ($p = 5$ in our case) and ε_{out} is the number of outliers that are tolerated ($\varepsilon_{out} = 50\%$ in our case). By substituting all these numerical values in the above formula we obtain $N = 100$ as the minimum number of samples.

Hence for the robust method to be effective, the inner loop of the algorithm must be iterated at least 100 times. Moreover, remember that outliers have two physical meanings: they may well correspond either to mismatches or to moving objects. Therefore we must be able to distinguish between inliers and small motions. To conclude, step 1.2 of the robust method is crucial and it must have the following features:

1. it must be fast because it has to be run many times and
2. it must provide an estimation of $\mathbf{H}$ as accurate as possible.

5.2 A five-point estimator

Let us devise an estimator for $\mathbf{H}$ that minimizes an Euclidean distance. In principle, such an estimator is non-linear because of the non-linear nature of the pinhole camera model. However, as described below, we have been able to devise a method which starts with a linear estimate of $\mathbf{H}$ and which incrementally and linearly updates the Euclidean error. Therefore, this method combines the efficiency of a linear estimator with the accuracy of a non-linear one. In practice it converges in a few iterations (2 to 3) and the solution thus obtained is very close to the solution that would have been obtained with a standard non-linear minimization method.

The method described below can deal with a number of point matches equal or greater than 5. Within the robust method described above it is however desirable to use the minimal set of points – 5 points in our case.

With the notations already introduced in section 4 let $\boldsymbol{X}$ be the vector of 3-D projective coordinates obtained by reconstruction from its projections $\boldsymbol{x}$ and $\boldsymbol{x}'$ onto the first image pair. Matrix $\mathbf{H}$ maps these coordinates onto $\boldsymbol{Y}$ such that $\boldsymbol{Y} = \mu\mathbf{H}\boldsymbol{X}$, and matrices $\boldsymbol{P}_y$ and $\boldsymbol{P}'_y$ reproject these coordinates onto the second image pair. Therefore we have the following estimated image points:

$$\alpha\hat{\boldsymbol{y}} = \mathbf{P}_y\mathbf{H}\boldsymbol{X} \tag{4}$$

$$\alpha'\hat{\boldsymbol{y}}' = \mathbf{P}'_y\mathbf{H}\boldsymbol{X} \tag{5}$$

The 3-vectors $\hat{\boldsymbol{y}}$ and $\hat{\boldsymbol{y}}'$ are defined up to a scale factor, α and α'. By dividing the first and second components of these vectors with their third component we get *estimated image positions* as opposed to $\boldsymbol{y}$ and $\boldsymbol{y}'$ which are *measured image positions*. The Euclidean distance between the measured point position $\boldsymbol{y}$ and the estimated point position $\hat{\boldsymbol{y}}$ is:

$$\varepsilon = d^2(\hat{\boldsymbol{y}}, \boldsymbol{y}) = \left(\frac{\hat{y}^{(1)}}{\hat{y}^{(3)}} - y^{(1)}\right)^2 + \left(\frac{\hat{y}^{(2)}}{\hat{y}^{(3)}} - y^{(2)}\right)^2 \tag{6}$$

with $\hat{\boldsymbol{y}}^\top = (\hat{y}^{(1)}\ \hat{y}^{(2)}\ \hat{y}^{(3)})$ and $\boldsymbol{y}^\top = (y^{(1)}\ y^{(2)}\ 1)$.

Let us write matrix $\mathbf{H}$ as a vector in $I\!R^{16}$:

$$\boldsymbol{h} = (H_{11}\ H_{12} \ldots H_{44})^\top = (h_1 \ldots h_{16})^\top$$

By substituting eq. (4) into eq. (6) and with the notation:

$$w = \frac{1}{\hat{y}^{(3)}} = \frac{1}{(\mathbf{P}_y\mathbf{H}\boldsymbol{X})^{(3)}} \tag{7}$$

we obtain for the Euclidean error:

$$\varepsilon = w^2\left(\sum_{j=1}^{16} a_j h_j\right)^2 + w^2\left(\sum_{j=1}^{16} b_j h_j\right)^2 \tag{8}$$

where the a_j and the b_j coefficients depend on $\boldsymbol{y}$, $\boldsymbol{X}$ and $\mathbf{P}_y$. Since we deal with an image pair the reprojected Euclidean error is $e = \varepsilon + \varepsilon'$

For m point matches we obtain the following criterion:

$$E = \sum_{i=1}^{m} e_i \tag{9}$$

$$= \sum_{i=1}^{m}\left(w_i^2\left(\sum_{j=1}^{16} a_{ij}h_j\right)^2 + w_i^2\left(\sum_{j=1}^{16} b_{ij}h_j\right)^2 + w_i'^2\left(\sum_{j=1}^{16} a'_{ij}h_j\right)^2 + w_i'^2\left(\sum_{j=1}^{16} b'_{ij}h_j\right)^2\right) \tag{10}$$

In order to find the matrix $\mathbf{H}$ or, equivalently, the vector $\boldsymbol{h}$ which minimizes the criterion E of eq. (10) we suggest the following incremental estimation method (notice that, by definition, the parameters w_i and w'_i are dependent of $\mathbf{H}$):

1. *Initialization*: estimate $\mathbf{H}$ using the linear estimator;
2. *Evaluate* the parameters w_i and w'_i using the current solution for $\mathbf{H}$, i.e., eq. (7);
3. *Minimize* the criterion E of eq. (10) using standard weighted linear least-squares to estimate $\mathbf{H}$;
4. *Stop test*: if there is no difference between the value of E obtained at the current iteration and the value of E obtained at the previous iteration, then stop, else return to step 2.

6 Detection of moving objects

The method described above estimates the dominant projective motion associated with the moving stereo sensor. Therefore it divides the observed scene points into (i) a set of inliers corresponding to the rigid scene and (ii) a set of outliers.

The outliers have two interpretations. On one side they may belong to moving scene objects and on the other side they may be "real outliers", i.e., mismatched and/or mistracked points.

In order to further classify the outliers into points belonging to various moving objects and into real outliers we suggest to use data classification techniques. Generally speaking, such a technique groups the available data into several classes based on some metric. The data that we want to classify are the scene points denoted by M. Let M_1, ..., M_n be the outlier points found by the robust method just described. Since the 3-D reconstruction is projective one cannot define a metric in 3-D space. Instead we consider the image projections of these points and therefore each point M is characterized by four such projections: $\boldsymbol{x}$ and $\boldsymbol{x}'$ present in the image pair before the motion and $\boldsymbol{y}$ and $\boldsymbol{y}'$ present in the image pair after the motion. Therefore one possible metric that measures the distance between two points is:

$$\delta(M_1, M_2) = \begin{array}{l} \max\{d(\boldsymbol{x}_1, \boldsymbol{x}_2), d(\boldsymbol{x}'_1, \boldsymbol{x}'_2), \\ d(\boldsymbol{y}_1, \boldsymbol{y}_2), d(\boldsymbol{y}'_1, \boldsymbol{y}'_2)\} \end{array} \quad (11)$$

This metric encapsulates the property that points which belong to the same moving object are closed to each other in *all* four images.

In addition to the point to point metric defined by eq. (11), the classification algorithm needs a cluster to cluster metric. The latter is defined as a single linkage distance:

$$\Delta(\mathcal{C}_1, \mathcal{C}_2) = \min_{M_1 \in \mathcal{C}_1, M_2 \in \mathcal{C}_2} \delta(M_1, M_2) \quad (12)$$

where $\mathcal{C}$ denotes a cluster.

Therefore, the goal is to group within the same set points that are close together and throw out isolated points. Among the many data classification techniques available, the hierarchical clustering algorithm [4] with single linkage is well adapted for our purpose for several reasons. First, it does not need to know in advance the final number of clusters to be found, which means it does not need to know, a priori, either the number of moving objects present in the scene, or the number of real outliers. Second, it uses a simple stop procedure based on the minimum distance allowed between two clusters. Third, the method is fast because the cluster to cluster distances are efficiently updated.

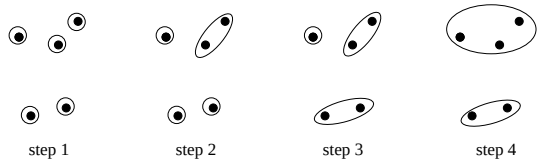

Figure 2: Hierarchical clustering algorithm

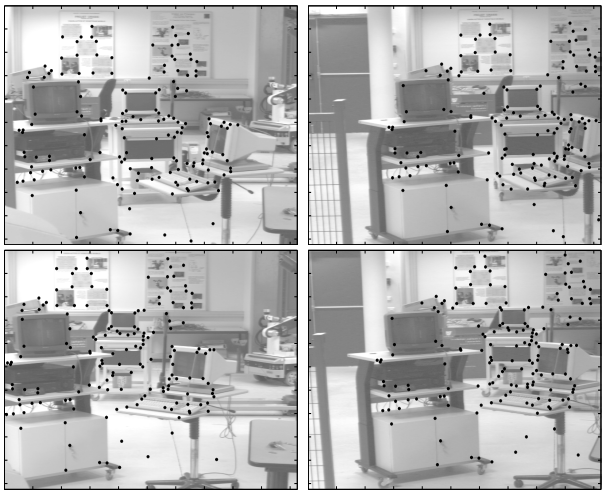

Figure 3: The "screens" sequence

The clustering algorithm based in incremental merging of the data is illustrated on Figure 2. At initialization there are as many clusters as there are points to be grouped. At each iteration of the algorithm the distances between all clusters are evaluated and the two clusters for which this distance is the smallest are merged together. The merging of clusters is thus repeated until the smallest distance is higher than a threshold t_d. It is worth noticing that if a dense matching is performed, a small value t_d can then be confidently chosen.

7 Experiments

This section describes an experiment using real images. A stereo rig has been moved while capturing a stereo sequence of a laboratory scene from which two frames have been extracted (see Figure 3). As it can be guessed from images, moving objects are the

computer screens at the middle and at the right. The motion of the rig is a slight translation to the right.

In the first stereo pair of the sequence, points are extracted and matched while robustly estimating the epipolar geometry between the left and the right camera. These points are then tracked in the following stereo frame and re-matched using a robust estimation of the epipolar geometry. Points that have been successfully matched are tracked in the next frames and the other ones are removed and replaced by additional extracted points. The process then goes on until the end of the sequence and enables then to:

- match points between successive stereo frames and estimate the average detection and matching precision σ, whose value turns to be about 1.0 pixel in this experiment;
- robustly compute the epipolar geometry for each stereo frame and detect a part of mismatched points (the ones that do not respect the robustly estimated epipolar geometry), but some outliers may have not been detected (because they are consistent with the epipolar geometry).

The robust projective motion algorithm has been applied. Figure 4 shows the detected inliers (*i.e.* background points) and outliers (*i.e.* points not conforming to the dominant motion).

Then, the clustering algorithm has been executed with the previous outliers (see Figure 4). We chose the tuning parameter t_d greater than 30 pixels. It effectively detected the two objects as shown on Figure 5 and a set of 9 isolated points which are in fact, as illustrated by Figure 6, mismatched points.

8 Discussion

In this paper, we have described an approach to detect moving objects with an uncalibrated stereo rig. Our approach is divided into two steps: (i) a robust *egomotion* estimation method based on 3D projective constraints, and (ii) objects detection using only image constraints. This framework enables us to robustly deal with many complicated situations (non rigid objects, noise) where many other methods fail.

A problem that could be put forward with this approach is the one of occlusions. We are now trying to solve it by treating whole sequences.

References

[1] M.A. Fischler and R.C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Graphics and Image Processing*, 24(6):381 – 395, June 1981.

[2] R. Horaud and G. Csurka. Self-calibration and euclidean reconstruction using motions of a stereo rig. In *Proceedings of the 6th International Conference on Computer Vision, Bombay, India*, pages 96–103, January 1998.

[3] M. Irani and P. Anandan. A unified approach to moving object detection in 2d and 3d scenes. IEEE *Transactions on Pattern Analysis and Machine Intelligence*, 20(6):577–589, June 1998.

[4] S.C. Johnson. Hierarchical clustering schemes. *Psychometrika*, (32):241–254, 1967.

[5] W. MacLean, A.D. Jepson, and R.C. Frecker. Recovery of egomotion and segmentation of independent object motion using the em algorithm. In E. Hancock, editor, *Proceedings of the fifth British Machine Vision Conference, York, England*, pages 175–184. BMVA Press, 1994.

[6] J.M. Odobez and P. Bouthemy. robust multiresolution estimation of parametric motion models. *Journal of Visual Communication and Image Representation*, 6(4):348–365, December 1995.

[7] S.M. Smith and J.M. Brady. A scene segmenter; visual tracking of moving vehicules. In *Intelligent Autonomous Vehicles*, pages 119–126, 1993.

[8] T.Y. Tian and M. Shah. Recovering 3d motion of multiple objects using adaptative hough transform. IEEE *Transactions on Pattern Analysis and Machine Intelligence*, 19(10):1178–1183, October 1997.

[9] P.H.S. Torr and D.W. Murray. Stochastic motion clustering. In J.O. Eklundh, editor, *Proceedings of the 3rd European Conference on Computer Vision, Stockholm, Sweden*, volume 801 of *Lecture Notes in Computer Science*, pages 328–337, May 1994.

[10] P.H.S. Torr and A. Zisserman. Robust parameterization and computation of the trifocal tensor. In R.B. Fisher and E. Trucco, editors, *Proceedings of the seventh British Machine Vision Conference, Edinburgh, Scotland*, volume 2, pages 655–664. British Machine Vision Association, September 1996.

[11] Z. Zhang, R. Deriche, O. D. Faugeras, and Q-T. Luong. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. *Artificial Intelligence*, 78(1–2):87–119, October 1995.

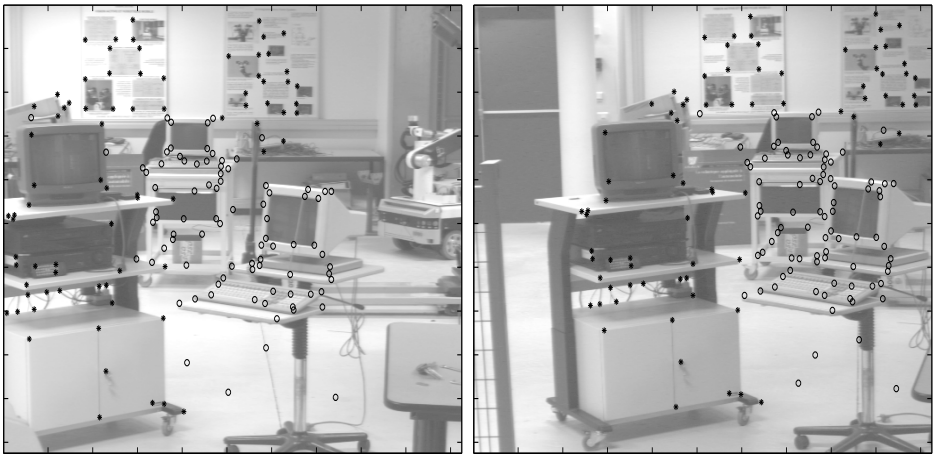

Figure 4: Inliers (dot) and outliers (circle) detected by the robust algorithm

Figure 5: The two detected moving screens

Figure 6: Detected mismatched points