Matching Feature Points for Telerobotics *

Étienne Vincent and Robert Laganière

School of Information Technology and Engineering
University of Ottawa, Ottawa, Canada, K1N 6N5
evincent,laganier@site.uottawa.ca

Abstract

A system that quickly and reliably matches points from images of a same scene, taken by three different cameras is presented. The first step consists in the weak calibration of the camera system. Then feature points are iteratively selected and matched in the first two images, guided by a fundamental matrix. Each correspondence is then validated by computing the position of matches in the third image using a trifocal tensor, and enforcing a constraint on the disparity of neighboring matches.

1 Introduction

In some telerobotic applications, an operator must remotely manipulate machinery while having access to views of the work environment taken by a limited number of fixed cameras. To manipulate his tool more precisely, it would often be useful for the operator to have access to arbitrary viewpoints of the work environment, or to have access to a model of the environment on which measurements could be taken.

If the position, orientation, and internal parameters of the fixed cameras are known, and if point correspondences are established between images taken simultaneously by the cameras, the position of those points in space can be computed by triangulation. These points could form the basis for a model of the environment, or be used in the generation of virtual intermediate viewpoints through interpolation.

A difficulty is that when the environment is constantly evolving, the point correspondences must be constantly recomputed to update the model. We thus seek to implemented a fast system for matching feature points between three views taken from fixed weakly calibrated cameras.

Our system produces a set of matched points between images taken from three cameras. It works by first detecting feature points in the first two views. Then it finds candidate correspondences among the two sets of points by correlating their image neighborhoods. This search for correspondence is guided by the epipolar geometry of the views. Next, the obtained pairs of points are verified to be true correspondences using the third image. For each pair, trinocular geometry is used to determine the expected position of the corresponding points in the third image. The image pattern at that position can then be compared, through correlation, to the other ones to determine the validity of the resulting triplet of points. It is very likely that three points with highly correlated neighborhoods, and agreeing with the cameras' trinocular geometry are all images of the same scene point. However, it is still possible that they are not, so an additional constraint requiring that the disparity of neighboring triplets be similar is imposed.

The goal is to use this system as a starting point for the development of improved feature point detection and comparison techniques that would address the typical problems of fast matching systems. This work is similar to others such as [5], where three images are sparsely matched. Their goal, however, is to calibrate the images. Here, weak calibration is performed offline, and used to increase the speed of matching.

The next section reviews the concepts of trinocular geometry. Then, section 3 describes how the camera system's trinocular geometry is estimated. Next, section 4 discusses how feature points are chosen. And finally, section 5 describes how feature points are matched.

2 Trinocular Geometry

A simple *pinhole model* can be used to represent cameras (see Figure 1). A point in space $\mathbf{X}$ is projected onto the image plane π, to a point $\mathbf{x}$, which is the intersection of the ray joining $\mathbf{X}$ and the camera's focal point $\mathbf{c}$ [3].

When two cameras look at the same scene, the projection $\mathbf{x}$, on one camera plane π, of an unknown point in space $\mathbf{X}$,

*This work was funded in part by the SMART project which was sponsored by IRIS/Precarn

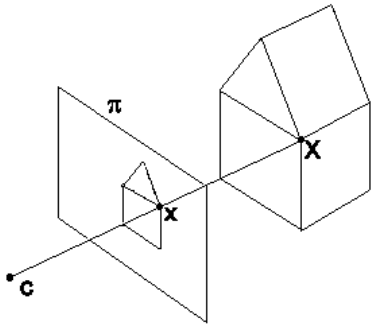

Figure 1. Pinhole model

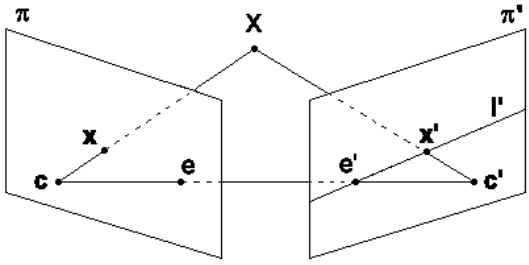

Figure 2. Two-view geometry

can tell us something about where the point will land on the other camera plane (see Figure 2). More precisely, it is known that a matching point in the second image, must be on the *epipolar line*.

Since $\mathbf{X}$ must be somewhere on the ray $\overline{\mathbf{cx}}$, its projection on π' must be on the projection of that ray. This projection, $\mathbf{l}'$, is called the epipolar line of $\mathbf{x}$.

For a pair of cameras, the relation between points in one view, and their epipolar line in the other, is called the *epipolar geometry*. It can be represented by a 3×3 matrix F, of rank 2: its *fundamental matrix*. Points $\mathbf{x}$ in the first image, are related to their epipolar lines $\mathbf{l}'$ in the second image, by:

$$F\mathbf{x} = \mathbf{l}' \tag{1}$$

where the point $\mathbf{x}$ is represented with homogeneous coordinates as $(x, y, 1)^\top$, and the epipolar line is the set of points $\mathbf{x}'$, represented in homogeneous coordinates, such that $\mathbf{l}'^\top \mathbf{x}' = 0$. This matrix can be estimated from known pairs of points between images produced by the two cameras using the fact that:

$$\mathbf{x}'^\top F \mathbf{x} = 0 \tag{2}$$

If three cameras are used, and a match $(\mathbf{x}, \mathbf{x}')$ is already known between the first two images, the position of the matching point $\mathbf{x}''$ in the third image can be determined exactly. Indeed, the projection of $\mathbf{X}$ on π'' should be at the intersection of the epipolar lines $\mathbf{l_1}''$ and $\mathbf{l_2}''$ of $\mathbf{x}$ and $\mathbf{x}'$ respectively.

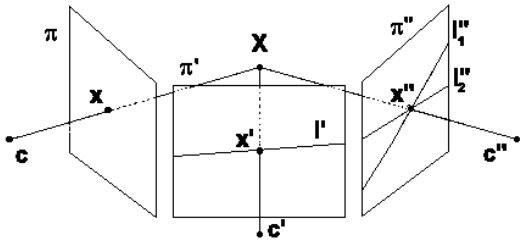

Figure 3. Three-view geometry

This relationship between points in three images is called *trinocular geometry* [6]. In fact, it can be shown that the position of $\mathbf{x}''$ can be determined even if the lines $\mathbf{l_1}''$ and $\mathbf{l_2}''$ are the same and thus have no single point of intersection. Trinocular geometry can be represented by the *trifocal tensor*, a $3 \times 3 \times 3$ tensor T for which:

$$\mathbf{x}''_k = \mathbf{x}_i \mathbf{l}'^{\perp}_j T_{ijk} \tag{3}$$

where $\mathbf{l}'^{\perp}$ is the line going through $\mathbf{x}'$, and perpendicular to $\mathbf{l}'$, the epipolar line of $\mathbf{x}$, and where i, j, k, and l are indices of the vectors and tensor. This formula can be used to compute the position of $\mathbf{x}''$, but our experiments have shown that more stable results can be obtained using:

$$\mathbf{x}''_l = \mathbf{x}'_i \sum_{k=1}^{3} \mathbf{x}_k T_{kjl} - \mathbf{x}'_j \sum_{k=1}^{3} \mathbf{x}_k T_{kil} \tag{4}$$

which defines 9 trilinearities for $i, j \in \{1, 2, 3\}$, 4 of which are linearly independent. $\mathbf{x}''$ can then be estimated by solving the over-constrained system of equations.

A trifocal tensor can be estimated from known corresponding point triplets between images taken from three cameras. Then, when a point correspondence is hypothesized between the first two images, it can be partially verified by using the tensor to *transfer* the match to the third image and check that the obtained point resembles the other two.

For the experiment described in this work, the setup shown in Figure 4 was used. It consists of three fixed cameras for which fundamental matrices and trifocal tensors must be estimated. The next section describes this estimation process.

3 Calibration

To obtain high matching speeds, a fundamental matrix and trifocal tensor must be used to guide the search for matches. When given a point in the first image, the search for its corresponding point in the second image can be restricted to a line using a fundamental matrix relating the first two cameras, and equation (1). Then using a trifocal

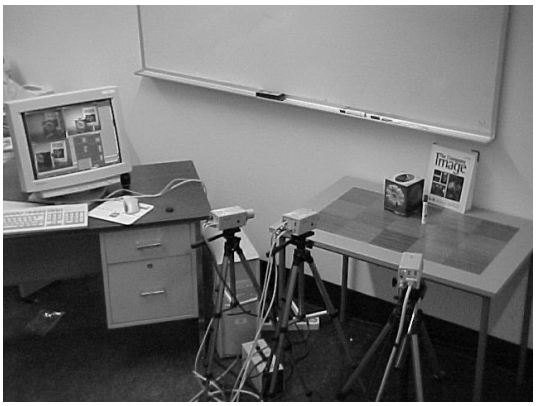

Figure 4. The Experimental Setup

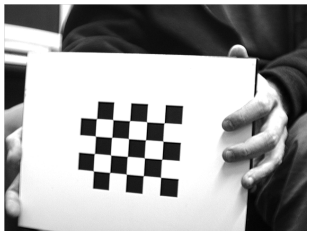

Figure 5. The calibration pattern

tensor with equation (4), the position of the matching point in the third image can be determined. Practically, since only an approximation of the camera system's geometry is available, the search for the match in the second image is limited to a band along the computed epipolar line, and the search for the match in the third image is restricted to a small region around the computed point location.

The estimation process used to determine the needed fundamental matrix and trifocal tensor is called *weak calibration*. To perform this estimation, established correspondences between the three views are used. Such point correspondences are determined automatically, in an offline calibration step that precedes matching. To easily determine such correspondences, the calibration pattern shown in Figure 5 is used.

This pattern can be detected by a function from Intel's Open Source Computer Vision Library [1]. This function will return a list of corner positions on the chessboard pattern on each image. When the lists of corners for three images taken simultaneously are aligned, point correspondences are obtained.

One triplet of views of the pattern is however insufficient for the estimation, as all the resulting detected points are from a single plane in space, which is a degenerate configuration. Thus, at least two shots of the pattern in different

[1] freely available at developer.intel.com

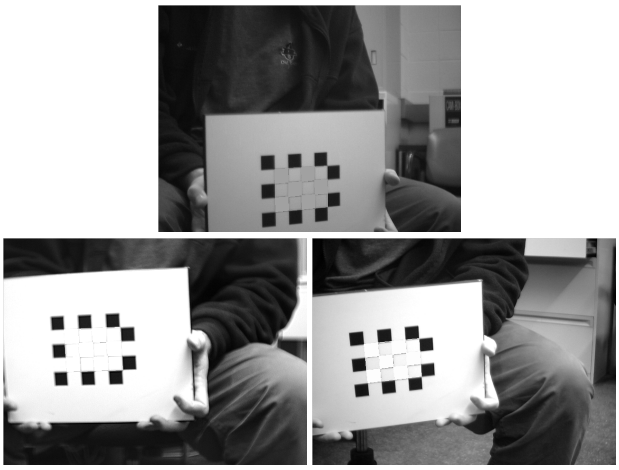

Figure 6. Detected calibration pattern

positions in space are needed. More shots should however be used to obtain better estimates. Five shots normally give good results.

To make the calibration process fast an easy, a user positions the pattern so that it can be seen by all three cameras. When the pattern is detected in all views, the system pauses momentarily to allow the user to reposition the pattern for the next shot. The pattern should be positioned, over the different shots, to cover as much of the scene viewing volume as possible. This is to result in an estimate of the camera system's geometry which is accurate over all possible regions where points should later be matched.

For many reasons, the pattern can sometimes be incorrectly detected. To avoid problems due to the resulting inaccurate calibration, a visual check of the detected pattern corners is performed after each shot is taken. Figure 6 shows the pattern having been detected in all three views, where a grid of white squares has been overlaid to cover the black squares and verify that they were detected accurately.

From the obtained correspondences between the corners of the chessboard pattern, the fundamental matrix and trifocal tensor can be estimated. One approach would be to first estimate the position, orientation and internal parameters of the three cameras, and from this information, to compute the matrix and tensor. However, this approach was found to be unstable, as small changes in the estimated camera parameters cause significant changes in the resulting fundamental matrix and trifocal tensor. It is thus preferable to perform weak calibration directly from the point correspondences.

Many different estimation processes using different parameterizations and minimization methods have been proposed for weak calibration [4, 8]. These can be rather complicated, as the computed matrix and tensor must satisfy some non-linear constraints which can only be enforced

through an iterative minimization. However, it was found that such accuracy in the fundamental matrix and trifocal tensor are not necessarily needed when they are used only to guide matching.

It was found to be sufficient to perform a direct linear estimation of the elements of the fundamental matrix and trifocal tensor using equations (2) and (4). In addition, as explained in [2], it is necessary to normalize the coordinate system, before solving the linear system, to ensure improve the stability.

Another possibility would have been to compute the tensor from the correspondences, and then to extract the fundamental matrix from the tensor. However, it was found that this approach was much less stable, as the error in the tensor translates into large errors in the fundamental matrix. It seems that although the estimated tensor is useful in guiding matching, it does not necessarily accurately describe the camera system's properties. It would be expected to only approximate transfer well over the common viewing area of the cameras, which is all that is needed here.

Once the fundamental matrix and trifocal tensor have been estimated, it was found that it can be useful to visually check their accuracy. This can be done using the chessboard pattern. The pattern can be detected in two views, then, the epipolar lines of corners in the first image can be drawn in the second one, and the computed position of the corners in the third image can be used to compute the expected appearance of the pattern in that image. It can easily be visually checked that the epipolar lines go through the corners in the second image, and that the computed chessboard corresponds to the actual one in the third image. With this system, the computed position of corners in the third image are generally closer than a couple of pixels to their real location, for any position and orientation of the pattern.

4 Feature Point Detection

It would be too costly to attempt matching every point in the three images. In fact, it is even too costly to compare points in the first image with every possible matching point along their epipolar line in the second image. Thus, the matching process is limited to a small number of selected *feature points*. Feature points in the first image are only compared to feature points along their epipolar line in the second image. The selection of feature points in the third image is not needed however, as the search for correspondence there is limited to one point's neighborhood.

Only points for which a corresponding point is likely to be easily distinguishable from its neighbors should be used as feature points. These points should also, as much as possible, represent significant scene features, to result in a good model of the environment. Points of high curvature on image edges are good candidates for this. These point's neighborhoods should have a high information content and often correspond to scene corners.

To select high curvature points, a Harris-like detector was used [1]. This detector can be implemented to run efficiently. It selects points in a relatively stable way, meaning that points corresponding to the same scene feature are often detected in different views of the scene [7]. This feature point detector finds points where the image intensity gradient has a high magnitude in more than one direction using the gradient's autocorrelation matrix:

$$C(x,y) = S * (\bigtriangledown I \cdot \bigtriangledown I^T) = S * \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \quad (5)$$

where S is a smoothing operator. At the point where it is computed, this matrix's greatest eigenvalue corresponds to the image's rate of change in the direction of highest variation, while its least eigenvalue corresponds to the rate of change in the perpendicular direction. If the least eigenvalue has a high magnitude, it means that, at the considered point, the image has a high rate of variation in at least two directions, and thus, that the point is in a high curvature region.

Figure 7 shows some feature points that were detected on images produced by the first two cameras.

5 Matching

As described before, feature points from the first image are compared to the feature points found along their epipolar line in the second image. The comparison is done using variance normalized correlation (VNC), which is designed to produce reliable results over a wide range of viewing conditions. VNC is defined for a candidate match $(\mathbf{x}, \mathbf{x}')$ as:

$$VNC(\mathbf{x},\mathbf{x}') = \frac{\sum_{\mathbf{n},\mathbf{n}'} [I(\mathbf{n}) - \overline{I(\mathbf{x})}][I'(\mathbf{n}') - \overline{I'(\mathbf{x}')}]}{N\sqrt{\sigma_I^2(\mathbf{x})\sigma_{I'}^2(\mathbf{x}')}} \quad (6)$$

where the sum is taken over the points $\mathbf{n}$ and $\mathbf{n}'$ in the neighborhoods of $\mathbf{x}$ and $\mathbf{x}'$ of size N, and where $\overline{I(\mathbf{x})}$ and $\sigma_I^2(\mathbf{x})$ are respectively the mean and the variance of the pixel intensities over a neighborhood of point $\mathbf{x}$.

However, the pairs of points found through correlation along the epipolar lines are not necessarily accurate correspondences. This is why a third image is used. Using a trifocal tensor, the pair is transferred to the third image, and correlation is applied between the neighborhood of the found point, and the neighborhood of the point in the pair from the image closest to the third one. It becomes very unlikely that a mismatched pair of points between the first two images be transferred to a point exhibiting high correlation to them.

After having kept only triplets having high correlation, some mismatches might still be present. Thus, a final con-

straint is applied to eliminate them. The disparity gradient is used, as in [9]. The disparity gradient is a measure of the compatibility of two pairs. For two pairs $(\mathbf{x}, \mathbf{x}')$ and $(\mathbf{y}, \mathbf{y}')$, having disparities $d(\mathbf{x}, \mathbf{x}')$ and $d(\mathbf{y}, \mathbf{y}')$ respectively, the cyclopean separation, $d_{cs}(\mathbf{x}, \mathbf{x}'; \mathbf{y}, \mathbf{y}')$, is the vector joining the midpoints of the line segments $\overline{\mathbf{x}\mathbf{x}'}$ and $\overline{\mathbf{y}\mathbf{y}'}$, and, their disparity gradient is defined as:

$$\Delta d(\mathbf{x}, \mathbf{x}'; \mathbf{y}, \mathbf{y}') = \frac{|d(\mathbf{x}, \mathbf{x}') - d(\mathbf{y}, \mathbf{y}')|}{|d_{cs}(\mathbf{x}, \mathbf{x}'; \mathbf{y}, \mathbf{y}')|} \qquad (7)$$

This compatibility measure is used in a constraint that accepts pairs that share a disparity gradients below some threshold value, with at least 2 of their 3 closest neighbors. This eliminates false matches as long as they are not surrounded by other similar false matches.

Figure 8 shows the result of applying this matching scheme to one frame taken by the three cameras of the experimental setup. In the first image, the lines join the coordinate of feature points there, to their coordinate in the second image, and thus represent the disparity between the first two views. Similarly, the lines in the second image indicate the disparity between that one and the third image. It can be seen that in this frame, there were no mismatches, and that matches were found over the important regions common to the three images, and were feature points can be found.

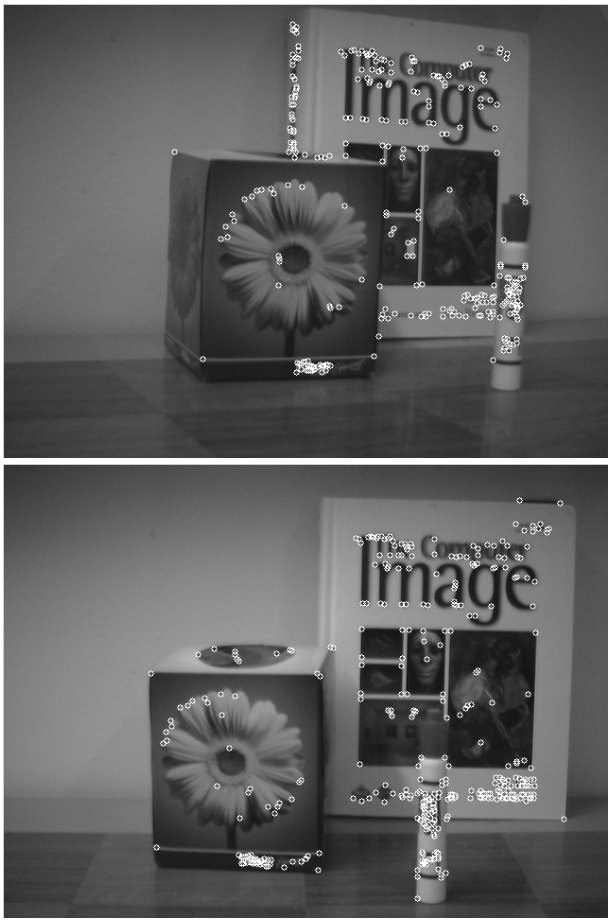

Figure 7. Detected feature points

6 Conclusion

A matching scheme has been presented which is used to find several matches between three images at a frame rate higher than 1 Hz, when running on very common hardware (333 MHz Pentium). This scheme incorporates many well known tools, including a Harris feature point detector, variance normalized correlation, disparity gradients, epipolar and trinocular geometry, and weak calibration from a test pattern.

Many interesting constatation were made. These include the fact that it was sufficient to use a linear method for the tensor estimation. It was also noticed that solving equation (4) gave more stable results for transferring points. Also interesting was that false matches were still found among triplets agreeing with the trinocular geometry and having high correlation, necessitating additional constraints.

Some weak points of the described method are its use of correlation, which is costly, and gives quickly degrading results as the difference between viewpoints increases. Another important weakness is the reliance on Harris feature points. It was found that these feature points, although widely used by many authors, are often poorly distributed in the images, resulting in a poor distribution of matches. This is always the case for scenes that do not contain many well defined corners, a situation that is common, especially when

Figure 8. matched points

relatively low resolution cameras are used. Some solutions to these problems are now being investigated.

References

[1] C. Harris, M. Stephens, A Combined Corner and Edge Detector, *Alvey Vision Conf.*, pp. 147-151, 1988.

[2] R. Hartley, In Defense of the Eight-Point Algorithm, *PAMI*, vol. 19, pp. 580-593, 1997.

[3] R. Hartley, A. Zisserman, Multiple View Geometry, *Cambridge University Press*, 2000.

[4] T. Papadopoulo, O. Faugeras, A New Characterization of the Trifocal Tensor, *ECCV*, pp. 109-123, 1998.

[5] G. Roth, A. Whitehead, Using Projective vision to find Camera Positions in an Image Sequence, *Proc. of Vision Interface*, pp.225-232, 2000.

[6] A. Shashua, Trilinearity in visual recognition by alignment, *ECCV*, pp. 479-484, 1994.

[7] C. Schmid, R. Mohr, C. Bauckhage, Comparing and Evaluating Interest Points, *ICCV*, pp. 230-235, 1998.

[8] P. Torr, A. Zisserman, Robust Computation and Parameterization of Multiple View Relations, *ICCV*, pp. 727-732, 1998.

[9] E. Vincent, R. Laganire, Matching Feature Points in Stereo Pairs: A Comparative Study of Some Matching Strategies, *MG&V*, vol. 10, pp. 237-259, 2001.