Effective Corner Matching

P. Smith*, D. Sinclair†, R. Cipolla* and K. Wood†
*Department of Engineering, University of Cambridge, Cambridge, UK
†Olivetti and Oracle Research Laboratory, Cambridge, UK
pas1001@eng.cam.ac.uk

Abstract

This paper tackles the problem of obtaining a good initial set of corner matches between two images without resorting to any constraints from motion or structure models. Several different matching metrics, both traditional and statistical, are evaluated and the effect of matching using sub-pixel information is studied. It is found that, in most cases, the commonly-used cross-correlation does not perform as well as some other measures, such as the χ^2 test or the sum of squared differences, and that it is essential to use sub-pixel accuracy if mismatches are to be avoided.

Further, a new technique, the *Median Flow Filter*, is introduced. This detects outliers by assuming that the image motion is locally similar. Any matches which are in gross disagreement with the local "median flow" are discarded. Experiments show this technique to be particularly effective, typically lowering the percentage of outliers from around 35% to less than 5%, permitting direct model fitting rather than random sampling techniques for any further analysis.

1 Introduction

Feature matching is a key component in many computer vision applications, for example stereo vision, motion tracking, and identification. Feature matches may be sufficient alone, but are also an ideal platform for "bootstrapping" denser and more complex analysis of images. Of all possible features, "corners" are the most widely used; their two-dimensional structure providing the most information about image motion.

Feature matching is commonly referred to as the *correspondence problem*. The problem is how to automatically match corresponding features from two images, while at the same time *not* assigning matches incorrectly. The common approach for corners, and the one followed in this paper, is to take a small region of pixels (referred to as a correlation window) from around the detected corner and compare this with a similar region from around each of the candidate corners in the other image. Each comparison yields a score, a *measure of similarity*. The match is assigned to the corner with the highest matching score. The most popular measure of similarity is the cross-correlation (see, for example, [3]).

Matching algorithms typically assume that the correlation windows from each image are related by a simple translation, an assumption which is valid in a number of typical applications. In the cases where this is not a valid assumption there are several approaches, the most common of which is to estimate the relationship between the two images and locally warp the correlation windows (see, for example, [6]). The standard correlation methods can then still be used. Other approaches, such as the local invariants method of Schmid and Mohr [9] are also successful. This paper only considers the simple translating case, but the measures of similarity could also be applied to warped correlation windows. The filter introduced in Section 5 can be applied to matches from any source.

The matching process is an ill-posed problem. There will inevitably be some corners which cannot be matched, and there are also likely to be several candidate matches for some corners which are very similar. All of these problems lead to *outliers* in the set of matches. A limited amount can be achieved by setting a threshold on the similarity measure (matches with a high score are usually correct), but it is virtually impossible to eliminate outliers completely.

Most matching algorithms include *constraints* to complement the similarity measure. These may take the form of constraints on which corners to select as candidate matches: a maximum disparity, or corners which agree with some known relationship between the images (such as the epipolar geometry). Constraints such as uniqueness or continuity may also be applied after candidate matches have been found. A good example of the application of constraints may be found in [11].

The application of constraints is not, however, always a good solution. In many cases there is no prior knowledge of where the candidate matches may lie, which removes one possible type of constraint. Post-processing candidate matches can be effective, but is usually time-consuming and relies on finding plausible solutions in the first place.

Despite the best attempts of corner matching algorithms, there will always be some outliers in the set of matches. As a result, algorithms have been developed over the last few years which can cope with a large number of outliers (usually no more than 50%) in the set of matches [10]. However, such algorithms are usually time-consuming and their speed and effectiveness can be greatly improved if a relatively clean set of matches can be found.

This paper presents a system for generating a very clean set of matches using minimal constraints by finding an effective measure of similarity and applying a simple filter to the results. Section 2 introduces the measures of similarity considered in this paper, Section 3 investigates the use of sub-pixel correlation windows, and the different measures of similarity are evaluated in Section 4. Finally, Section 5 introduces the median flow filter.

2 Measures of similarity

Matches are found by evaluating the similarity between image regions and selecting as a match that pair of regions with the closest similarity. There are many possible measures of similarity, of which a few of the most popular are considered here. Some are taken from the field of signal processing (and are now also common in computer vision) [5] and others from the realm of statistics [2, 7].

In the following definitions, individual pixels from the two images I and J are denoted by i and j. Summations are a dual summation over each of the n corresponding image pixels within the correlation window unless otherwise stated. The correlation scores C_1 and C_2 are to be maximised (closest to 1) for the best match; all others must be minimised.

- The *standard cross-correlation* is defined as

$$C_1 = \frac{\sum ij}{\sqrt{\sum i^2 \sum j^2}} \tag{1}$$

- As an alternative, a *zero mean cross-correlation* may be used:

$$C_2 = \frac{\sum \left(i - \bar{i}\right)\left(j - \bar{j}\right)}{\sqrt{\left(\sum i^2 - n\bar{i}^2\right)\left(\sum j^2 - n\bar{j}^2\right)}} \tag{2}$$

- Another common metric is the *sum of squared differences*:

$$C_3 = \sum \left(i - j\right)^2 \tag{3}$$

- A variant of (3), the χ^2 *test*, is well known in the field of statistics as a measure of the similarity between two distributions:

$$C_4 = \sum \frac{\left(i - j\right)^2}{\left(i + j\right)/2} \tag{4}$$

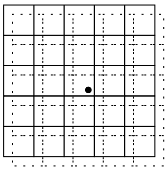


Figure 1: *A comparison between integer and sub-pixel approaches.* The solid line shows the actual image pixels in the region of the corner feature (marked by a dot). We could take these pixel values as the values for our measure of similarity, but ideally we would like a set of pixels centred on the corner, as shown by the dotted lines.

- The *Kolmogorov-Smirnov distance* also comes from the field of statistics. It is defined as the maximum vertical distance between two cumulative distributions. Since images are two-dimensional they must, in this case, be "unravelled" into a vector; we choose to do this column-by-column. In this case, i_k and j_k are the kth elements of this vector:

$$C_5 = \max_{t \le n} \left| \sum_{k=0}^{t} (i_k - j_k) \right| \tag{5}$$

- The *Jeffrey divergence* is an empirical measure of the similarity of two distributions, based on their relative entropy:

$$C_6 = \sum \left(i \log \frac{i}{(i+j)/2} + j \log \frac{j}{(i+j)/2} \right) \tag{6}$$

- Finally, we consider a variant of the *Earth Mover distance* introduced by Rubner *et al.*, which they applied to texture-based image retrieval [8]. This is based on the cost of "moving" the elements of one distribution around to make the other:

$$C_7 = n \left| \bar{i} - \bar{j} \right| + \sum_{\text{moves}} md \tag{7}$$

  The first term accounts for the difference in overall intensities between the two image patches. Once this is accounted for, we can "redistribute" the areas of intensity in one patch in order to make it identical to the other. This thus makes some allowance for differences in the spatial distribution of intensity in the patch, as well as the pixel-by-pixel differences. Rubner *et al.* only use the second term, and normalise by the smaller of the two distributions to compensate for different total masses.

  Our implementation uses a simple algorithm to move intensity from one pixel in the patch to another (ideally the best solution would be found by some minimisation method, such as the Simplex algorithm). Each move contributes an additional cost to the metric: the product of m, the amount of intensity moved and d, the (ground) distance moved. The moving process is continued until the two patches are identical (apart from rounding errors).

3 Sub-pixel correlation windows

While corner features are typically found to sub-pixel accuracy, this information is not always used in the matching stage. It is easier to round the corner location to the nearest pixel and use this image pixel value, and the neighbouring pixel values, in calculating the similarity measure.

Figure 2: *A selection of the test images.* Real images from video and digital stills cameras were used, with a mixture of stereo pairs and motion scenes (containing independently moving objects). Approximately 120 corners were used from each image.

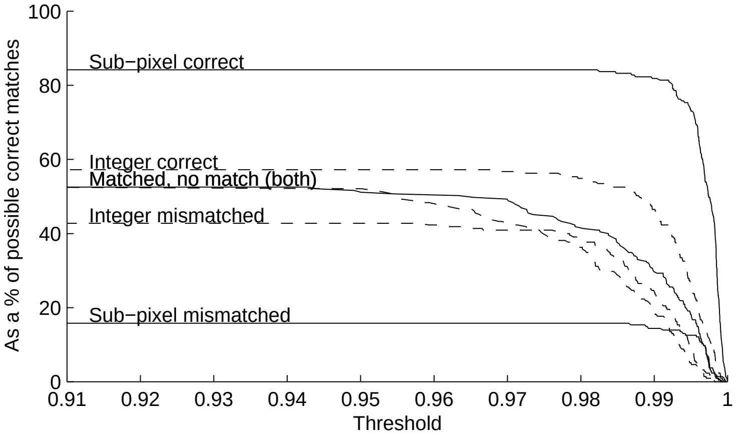


Figure 3: *The effect of sub-pixel correlation windows on numbers of matches*, showing the number of corner matches of each type (correctly matched, mismatched, and matched when there was no match) as the confidence threshold is increased. Combined results from all images using the cross-correlation similarity measure (1) with integer (dashed) and sub-pixel (solid) pixel values. The sub-pixel correlation window finds many more correct matches.

We compare this approach with the more "correct" approach of using the corner's sub-pixel information and calculating intensity values for the correlation window by bilinear interpolation of the image pixels across a grid centred on the corner feature (see Figure 1). This calculation increases the calculation time, but if implemented efficiently does not increase the time by a substantial amount.

3.1 Results

Using the standard cross-correlation measure of (1), the performance of the corner matcher was evaluated over a series of different images. Test images were of real scenes, some also containing independent motion, and were captured with video cameras or digital stills cameras. The image pairs conformed to the necessary criteria for correlation matching, i.e. similar lighting conditions and little deformation of corners between frames. Methods of dealing with larger image deformations were discussed in the Introduction. Around 120 corners were found in each image using the Harris corner detector [4]. For matching, a 5×5 correlation window was used and a disparity constraint of half of the image width was also imposed (all correct matches were within this limit).

The combined results for all images are presented here—each corner's image patch can be considered as an independent pattern, and the image to which it matches is another constraint. If we first consider the case where no confidence threshold is used, a match is found for every corner, including those which have no possible match. This is one source of

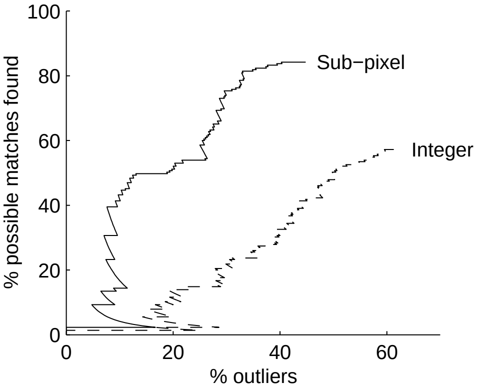


Figure 4: *The effect of sub-pixel correlation windows on matching percentage.* For each possible threshold value the the percentage of possible matches found is plotted against the percentage of outliers. The traces are for the combined results of the the cross-correlation measure of similarity (1) on each of the test images, with and without using sub-pixel values. Sub-pixel correlation windows show a clear improvement over integer windows.

outliers, and an inevitable one. However, there is another source of outliers: mismatching points which could, in fact, be matched correctly. Using the integer pixel locations, 43% of possible correct matches were matched incorrectly, leading to a total outlier fraction of 63%. Using sub-pixel pixel locations for the matching measure reduced the number of mismatches to 16%, and only 45% outliers.

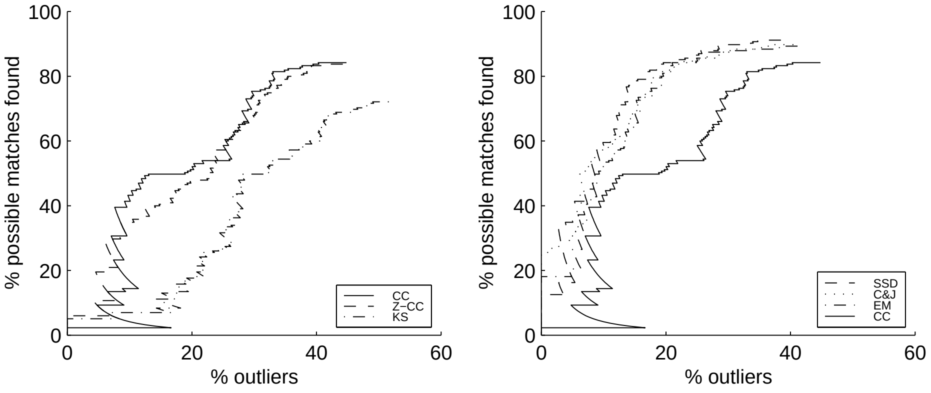
It is essential to employ a threshold on the match confidence to reduce the outlier percentage. Points matched where there was no possible match, and mismatched points, should have a lower similarity score than correct matches. By varying the threshold, a value should be found which includes a large number of correct matches while discarding the majority of the outliers.

Figure 3 shows the effect on the match set as the threshold is varied for each of the two approaches: integer pixel locations and sub-pixel locations. The plot clearly shows the increase in the number of correct matches and the reduction in the number of mismatches between the two approaches. As the threshold is increased, it can be seen that the outliers for which there is no correct match begin to reduce in number more quickly than the correct matches, as required. It can also be noted that the mismatches under the integer pixel method also begin to reduce—they are obviously less confident matches. The most important observation is that, using the sub-pixel locations, the number of correct matches does not begin to tail off until most of the outliers have been removed. In the integer case, the correct matches tail off much earlier, making it more difficult to set a threshold that keeps a significant number of inliers while rejecting outliers.

This can be seen more clearly in Figure 4, which plots a profile showing the percentage of possible matches found against the percentage of outliers. Distance along the curve is parameterised by matching threshold, each point on the curve representing the properties of the match set obtained with a particular threshold. Clearly, lower thresholds provide a greater number of correct matches, and also a greater number of outliers.

As the threshold is increased, the integer pixel method can be seen to reduce both the percentage of outliers and the percentage of matches found at the same time. Thus, for example, if only 20% outliers are required, we must be satisfied with around 15% of the possible matches. The sub-pixel method, on the other hand, reduces first one and then the other. Using this method, we can have 50% of possible matches at our 20% outlier level.

These results show that the sub-pixel calculations should be considered an essential component of any matching algorithm, despite the extra computational overhead. With the



(a) *No better than cross-correlation (CC)*: Zero mean cross correlation (Z-CC) and Kolmogorov-Smirnov distance (KS).

(b) *Better than cross-correlation (CC)*: Sum of squared distances (SSD), the χ^2 test and Jeffrey divergence (identical) (C&J), and the Earth Mover distance (EM).

Figure 5: *Performance of different measures of similarity compared to the cross-correlation.* All use sub-pixel correlation windows.

sub-pixel grid the registration between the two correlation windows is much closer, which increases the confidence in good matches and reduces the chance that another similar corner area will be matched in its place. Equally impressive results were obtained using each of the other measures of similarity evaluated in this paper.

4 Matching corner features

Although the cross-correlation is the similarity measure of choice for many computer vision applications, with so many other possible measures available it has to be able to justify its position. We compared the performance of the cross-correlation against a selection of other well-known measures, introduced in Section 2.

4.1 Results

Each similarity measure was tested on test images from Section 3 using a 5×5 correlation window and a half-image disparity constraint. Corners were again found using the Harris corner detector [4] and, given the results of Section 3, sub-pixel correlation windows have been used throughout.

Figure 5 shows the profiles of each of the different similarity measures as the threshold is varied. The matchers fall neatly into two distinct classes: those which are as good as, or worse than the cross-correlation, and those which are better. The measures of similarity which perform no better than the cross correlation are shown in Figure 5(a). The performance of the zero mean cross-correlation (2) is very similar to that of the cross-correlation, which indicates that the complexity of a measure is not necessarily a measure of its quality. The Kolmogorov-Smirnov distance (5) performs the worst of the measures tested. Given that it took a little “fudge” to make it work for two-dimensional distributions in the first place, this indicates that this particular statistical test is not best suited to images.

Figure 5(b) shows that there are a number of methods, basically indistinguishable, which perform better than the cross-correlation. The sum of squared differences (3), χ^2

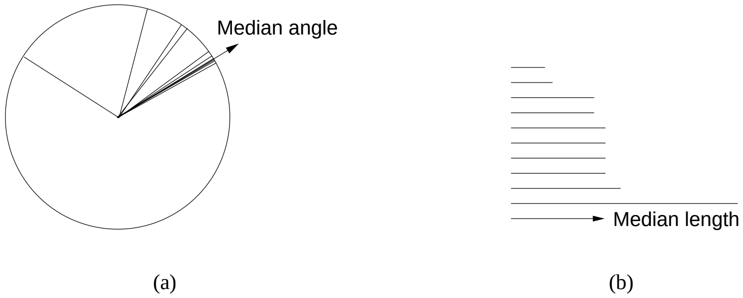


Figure 6: *Calculating the "median flow" in an area of the image* by taking the mean of the most tightly bunched n vectors from the k nearest vectors, using first their direction and then their length (here, $n = 3$).

test (4), Jeffrey divergence (6) and the Earth Mover distance (7) all find more of the possible matches, for any given outlier percentage, than the cross-correlation. The profile can also be seen to degrade more gracefully as the threshold varies, which makes the tricky job of selecting a threshold somewhat easier. We nominate the sum of squared differences as the best of this group, although both the χ^2 test and Jeffrey divergence (the results for which were identical) find the most correct matches for very small outlier levels.

These results show that the cross-correlation is *not* the best measure of similarity in cases where the small-deformation assumption of feature matching is valid. There are a range of techniques available which perform better, some of which also have a lower calculation cost.

4.2 A word of caution

In performing tests on images with larger degrees of distortion e.g. wide baseline stereo, rotation effects, or imperfect cameras, it was found that the similarity measures presented here as the best were a little *too* discriminating. A large number of corners were matched with very low confidence, and of those a large number were mismatched. In these circumstances, the cross-correlation performed better. However, there are a wide number of applications, particularly the recent interest in feature-based mosaicing [1, 12], where the image deformations are small (or the patches may be warped locally to compensate), and the process would benefit greatly by using one of the better matchers presented here.

5 Refining matches: the Median Flow Filter

We have seen that a good number of inliers can be found, with a substantial reduction in the number of outliers, by using sub-pixel matching with a suitable measure of similarity and a sensible threshold. However, some applications require even better performance and even robust algorithms perform better with smaller outlier levels.

Having reached the limits of using a confidence threshold, we here introduce a post-processing stage to the corner matching process which is both fast and does not require any specific knowledge of the scene or motion.

5.1 Algorithm

The median flow filter is founded on the assumption that image motion shows only a small variation across local regions; outliers are usually grossly in error compared to the local image motion. The algorithm considers matches as motion (or "flow") vectors, whatever the source of the image motion.

Each vector in turn is compared with its neighbours. If it points in a similar direction or is a similar, small, length when compared to the "median flow" in that area, then it is classified as an inlier, otherwise it is discarded as an outlier. Vectors are first tested on angle:

Angle For each match, the k nearest motion vectors are found and sorted by direction. We define the "median angle" of these vectors as the mean of the tightest group of n (i.e. those n which are closest together in terms of angle—see Figure 6(a)). We find that values of $k = 10$ and $n = 3$ work well. If the direction of the match vector is within a threshold t_1 of the median angle (we use $t_1 = 5\,^{\circ}$), it is classified as an inlier.

The angle of short motion vectors, particularly in the presence of noise, cannot be determined with any great certainty. Hence, for vectors below a certain length threshold l, we permit a second test based on the length of the vector.

Length The "median length" is taken to be the mean of the most tightly bunched group of n vectors from the k neighbours (see Figure 6(b)). If the motion vector for the match in question is within a threshold t_2 of the median length, it marked as an inlier. We use values of $l = 12$ pixels and $t_2 = 3$ pixels.

Obviously, the level of the thresholds t_1, t_2 and l may be set to be as strict or as relaxed as required, and the "median" parameters k and n may be set as required by the density of matches available.

5.2 Results

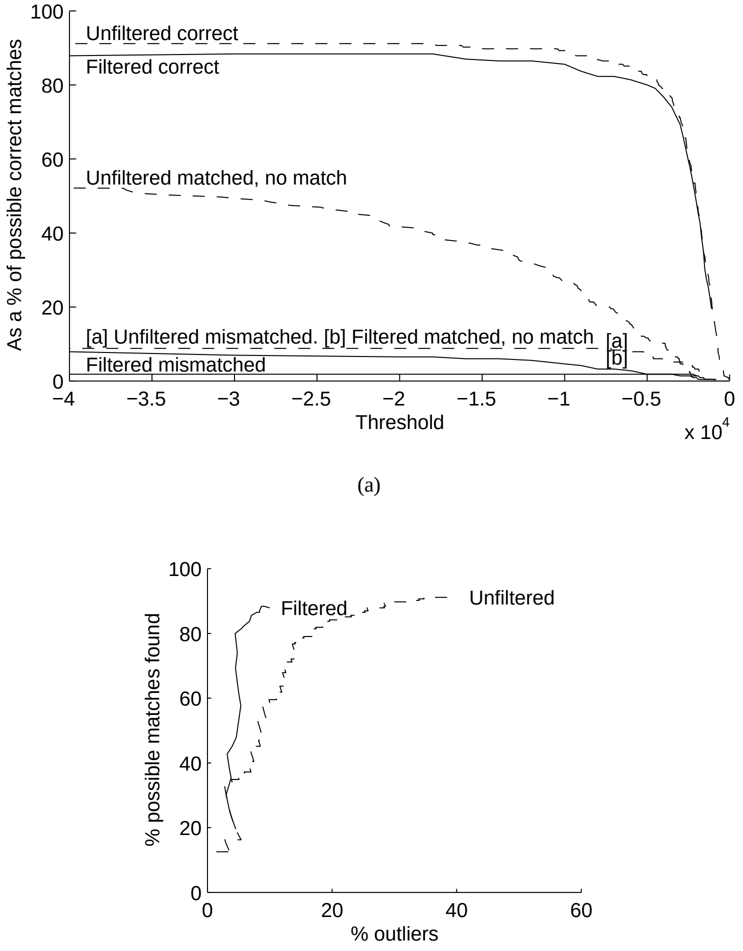
The median flow filter was tested on our series of images, taking the matches from the sum of squared differences matcher (3) for various threshold levels (and thus various numbers of inliers and outliers). Figure 7(a) categorises every match for each value of the confidence threshold. It can be seen that the median flow filter reduces both the number of mismatches and the number of matches with no correct match to very low levels, with only a slight reduction in the number of correct matches. Figure 7(b) profiles the performance of the sum of squared differences matcher with and without the median flow filter, where the clear improvement can also be seen. The filtered match set never has more than 10% outliers.

It can be seen that the median flow filter cleans up the matches very successfully indeed, for any level of outliers. It is also faster than alternative iterative or random sampling techniques, and does not require a model of the expected image motion (or motions). Figure 8 shows the results for one of our test images, displaying matches found with and without the median flow filter.

6 Summary and Conclusion

Despite the correspondence problem being ill-posed, and outliers in matched sets accepted as inevitable, by applying the right technology it is possible to gain significant improvements over current techniques. The use of sub-pixel correlation windows is essential if mismatches are to be avoided and the measure of similarity used should be considered carefully. For many applications the cross correlation is outperformed by the sum of squared differences, the χ^2 test, the Jeffrey divergence, and the Earth Mover distance.

The median flow filter introduced here can form a very potent weapon in the armoury of outlier detection and removal. It is simple and efficient, and does not rely on any application-specific constraints. The thresholds may be derived from the expected variance in corner location and the motion application, making the notion of "similar" to the local flow as strict or relaxed as the application requires.



(a)

(b)

Figure 7: *The effect of median flow filtering.* Performance of the sum of squares matching method (3) with and without post-processing by the median flow filter (with: solid, without: dashed). The median flow filter greatly reduces the number of outliers with only a small reduction in the number of correct matches found.

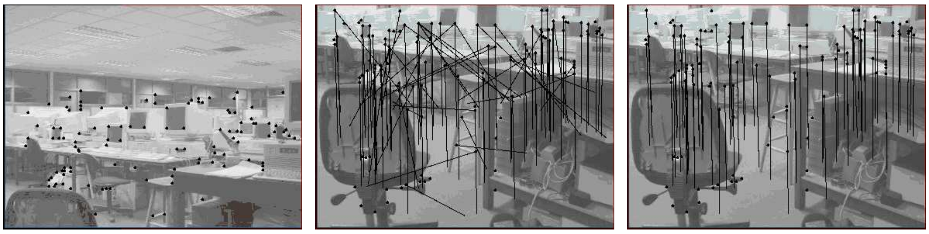


Figure 8: *Example test results.* The camera rotates about a horizontal axis between the two frames, resulting in a vertical image motion. Left: Image 1 with 110 corners overlaid. Centre: Image 2 with matches from the sum of squared differences matcher (3) with no threshold (this finds 97% of possible matches, with 35% outliers). Right: After filtering we are left with the same number of inliers, but only 4% outliers.

Experiments have demonstrated that, with a combination of one of the better measures mentioned and the median flow filter, a large number of the possible matches can be correctly detected with only a very small number of outliers.

7 Acknowledgements

This research was funded by an EPSRC studentship, with a CASE award from the Olivetti and Oracle Research Laboratory, Cambridge, UK. Thanks go to T. J. Cham for the use of some of his test images.

References

[1] T. J. Cham and R. Cipolla. A statistical framework for long-range feature matching in uncalibrated image mosaicing. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 442–447, Santa Barbara, CA, 1998.

[2] W. J. Conover. *Practical Non Parametric Statistics*. Wiley series in probability and mathematics statistics. John Wiley & Sons., second edition, 1980.

[3] O. Faugeras, B. Hotz, H. Mathieu, T. Viéville, Z. Zhang, P. Fua, E. Théron, L. Moll, G. Berry, J. Vuillemin, P. Bertin, and C. Proy. Real time correlation-based stereo: algorithm, implementations and applications. Research Report 2013, INRIA Sophia-Antipolis, France, August 1993.

[4] C. G. Harris and M. Stephens. A combined corner and edge detector. In *Proc. 4th Alvey Vision Conference*, pages 147–151, 1988.

[5] V. S. Nalwa. *A Guided Tour of Computer Vision*. Addison-Wesley, 1993.

[6] P. Pritchett and A. Zisserman. Wide baseline stereo matching. In *Proc. 6th International Conference on Computer Vision*, pages 754–760, Bombay, India, January 1998.

[7] J. Puzicha, T. Hofmann, and J. M. Buhmann. Non-parametric similarity measures for unsupervised texture segmentation and image retrieval. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 267–272, San Juan, PR, June 1997.

[8] Y. Rubner, C. Tomasi, and L. J. Guibas. A metric for distributions with applications to image databases. In *Proc. 6th International Conference on Computer Vision*, pages 59–66, Bombay, India, January 1998.

[9] C. Schmid and R. Mohr. Local greyvalue invariants for image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(5):530–535, May 1997.

[10] P. H. S. Torr and D. W. Murray. The development and comparison of robust methods for estimating the fundamental matrix. *International Journal of Computer Vision*, 24(3):271–300, 1997.

[11] Z. Zhang, R. Deriche, O. Faugeras, and Q.-T. Luong. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. *Artificial Intelligence Journal*, 78:87–119, October 1995. Also Research Report No.2273, INRIA Sophia-Antipolis.

[12] I. Zoghlami, O. Faugeras, and R. Deriche. Using geometric corners to build a 2D mosaic from a set of images. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 420–425, San Juan, PR, June 1997.