University of Edinburgh

Department of Computer Science

Extensions to a Change Detection in MRI Brain Scan Data Program

4th Year Project Report

Peter I Oliver

31st May 1995

Abstract

This dissertation describes research into the problem of automated change detection within MRI brain scan imagery. The recent quality improvements of such data, coupled with faster machinery and algorithms has led to the development of a realisable system. Problems overcome include the synthesis of multi-modal (or multi-channel) data and point-wise image region matching strategies. Matching involves computing directed line segments that point from local regions in the source image to similar local regions in the target image. These mappings, if correct, will then encode both the global mis-registration of the image, and the local structural changes. After constraints, aimed at improving the initial matching, have been imposed, a least squares estimate is made of the global transformation. This global transformation is then removed from the match mappings, allowing analysis of the residual vectors which indicate local structural changes between images. Pixel level quantisation often distorts data and hinders image matching. In solution, a multi-variate sub-pixel matching algorithm is developed which aims to describe local brain structure changes as quantified pixel movement vectors. Considerations are given to the benefits post-operative match map improvement and analysis techniques. Results are shown throughout demonstrating the performance of the techniques described.

Acknowledgements

Warmest thanks go to Bob Fisher whose most patient and expert explanations and guidance made the project what it is.

Thanks to Jonny Tams for moral support and a rather nifty WYSIWYG equation editor which exports to LaTeX(so I didn't have to learn the math environment properly).

Thanks also to Dr Crystl Donnely of the University of Edinburgh Maths and Statistic departments on her help in locating the Geiss *et. al.* [5] paper. Thanks to Dr. J. E. Rimmington of the Dept. of Medical Radiology/Magnetic Resonance Imaging Unit for her co-operation with this project and supplying the image data. Thanks to Elizabeth Guest for her valuable initial guidance.

And finally, deepest thanks to my wife Wendy for all her support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In 1990 Woodward [10] delivered his M.Sc. dissertation describing a method for detecting brain structure changes over time by analysing MRI (Magnetic Resonance Imagery) data. The data available at the time was poor, and much of Woodward's work was hindered by unavailability of good test data. Nevertheless, he developed a system the showed promise.

The availability of better data sets and faster machinery has prompted a re-apraisal of Woodward's work. This is the topic of this dissertation.

Woodward worked with two-dimensional MR imagery of 128 × 128 by eight bits per pixel. Currently available imagery consists of 256 × 256 by sixteen bits per pixel, multi-channel serially sectioned data. A full description of the data and file formats can be found in the files pointed to by the Appendix. New image matching techniques which utilise the full duality of the bi-variate data are developed.

This dissertation presents the development of Woodward's initial findings into a change system capable of run-times measured in minutes and not hours (as Woodward's was).

The basic strategy is this: Two images are first taken and matched on a pointwise basis. Image pixel values need not be scalar—vector quantities such as those from colour imagery, or multi-modal CT imagery are shown to work better. Matching involves computing directed line segments that point from local regions in the source image to similar local regions in the target image. These mappings, if correct, will then encode both the global mis-registration of the image, and the local structural changes. After constraints, aimed at improving the initial matching, have been imposed, a least squares estimate is made of the global transformation. This global transformation is then removed from the match mappings, allowing analysis of the residual vectors which indicate local structural changes between images.

This dissertation is presented in four main parts. A thorough investigation of image region similarity measures is conducted, followed by their application to image matching. Next, measures aimed at improving the initial matching are considered, concluding with considerations for change analysis. For a very brief introduction to the capabilities of the system developed here, the reader is referred to Plates 1 and 2.

Chapter 2

Image Matching

The initial point to point image matching requires that, for each pixel in the initial image, a corresponding location is found in the later image. Under the assumption that the structure, magnification and rotation of any part of the image changes little from the initial to subsequent images, a match for any pixel can be found by maximising the amount of similarity between the pixel and it's neighbours and a similarly defined region in the second image.

Clearly then, essential to the success of image region matching is the choice of similarity metric (or image match evaluation function). This chapter presents a number of evaluation functions, assesses their suitability for use with MRI brain scan data and presents empirical justification for the conclusions reached.

2.1 Image Region Similarity Metrics

The MRI brain scan data consists of two data sets (pre-processed and smoothed to remove the skull and other artifacts) for each scan. The first gives a measure of proton density (PD), and the second is derived data known as T2. The T2 images highlight tissue structures of interest to physicians and neurophysiologists for easier human interpretation.

A number of single channel image match evaluation functions have been considered, such as:

- square root of sum of squares of differences of pixel values over the matched region. This is the Euclidean metric.
- sum of absolute differences of pixel values over the matched region. This is the Manhatten metric.
- largest single absolute difference of pixel values over the matched region. This method is especially sensitive to spot noise, and so it may be necessary to first pre-process the images with a noise reducing filter.
- statistical cross-correlation of image regions [10].

Each of these evaluation functions are described in detail below, first in the context of single channel matching, and then generalised to multi-channel matching.

2.2 Single Channel Image Matching

A single channel matching problem matches images whose pixels are scalar values. Any correlation in the local structure of the two images should be found by the matching process.

Woodward [10] considered such a single channel matching problem in the pre-cursor to this project. He used a local normalised correlation image match function (Equation (2.4)), but did not present any alternatives. This is surprising since correlation is computationally more complex than many of the alternatives, and his matching algorithm averaged eight hours to arrive at an initial global image match for a relatively small pair of images ([10], p. 70).

To a large extent, Woodward's empirical evaluation was marred by poor test data. Between the before and after images there had been such marked brain degradation that little correspondence between the two images existed—even to the human observer. Faced with this problem, Woodward had no choice but to create his own 'after images' by transforming the initial data by some known parameters. While this provides a good testbed for evaluation, it is difficult to emulate the types of distortion to be found in 'real' data.

The following sections present in detail a number of single channel image match evaluation functions, and assess their suitability to region matching in MRI brain scan data.

2.2.1 Single Channel Euclidean Metric

Let $\{x_i\}$ be the values of pixels from a neighbourhood, and let $\{y_i\}$ be the values of pixels from a corresponding neighbourhood from the second image. Then if samples are taken over a neighbourhood of size N, define the Euclidean similarity metric as:

$$e = \sqrt{\sum_{i=1}^{N}(x_i - y_i)^2} \tag{2.1}$$

The best region match is given by choosing a neighbourhood $\{y_i\}$ that minimises e for a given $\{x_i\}$.

2.2.2 Single Channel Manhatten Metric

As above, let $\{x_i\}$ and $\{y_i\}$ define neighbourhood samples of size N from two images, then the Manhatten similarity metric is given by:

$$l_1 = \sum_{i=1}^{N} \mid x_i - y_i \mid \tag{2.2}$$

The best region match is given by choosing a neighbourhood $\{y_i\}$ that minimises l_1 for a given $\{x_i\}$.

2.2.3 Single Channel Largest Absolute Difference Metric

Again, let $\{x_i\}$ and $\{y_i\}$ define neighbourhood samples of size N from two images, then the largest absolute difference metric is given by:

$$l_\infty = \max_i(\mid x_i - y_i \mid) \tag{2.3}$$

The best region match is given by choosing a neighbourhood $\{y_i\}$ that minimises l_∞ for a given $\{x_i\}$.

This method suffers from spot noise. For this reason it may be especially important that the images are first preprocessed with a noise reducing filter.

2.2.4 Single Channel Normalised Cross-Correlation Metric

Woodward [10] used this metric as the basis for his region match evaluation function. It has a history of effective use [1, 6, 9, 7].

Define the mean of the N x samples (from the first image region) as:

$$\overline{x} = \frac{1}{N}\sum_{i=1}^{N} x_i$$

and the variance as:

$$\sigma_x^2 = \frac{1}{N}\sum_{i=1}^{N}(x_i - \overline{x})^2$$

and similarly for $\overline{y}$ and σ_y^2 (from the second image region). Then the normalised correlation function is given by[1]:

$$\rho = \frac{1}{N}\sum_{i=1}^{N}\frac{(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sigma_x^2\sigma_y^2}} \tag{2.4}$$

The best region match is given by choosing a neighbourhood $\{y_i\}$ that maximises ρ for a given

[1] A neater form for the divisor might be: $\sigma_x\sigma_y$, but during evaluation a direct translation would perform two `sqrt()` operations. The equations given throughout have been carefully chosen to reflect the actual implementation.

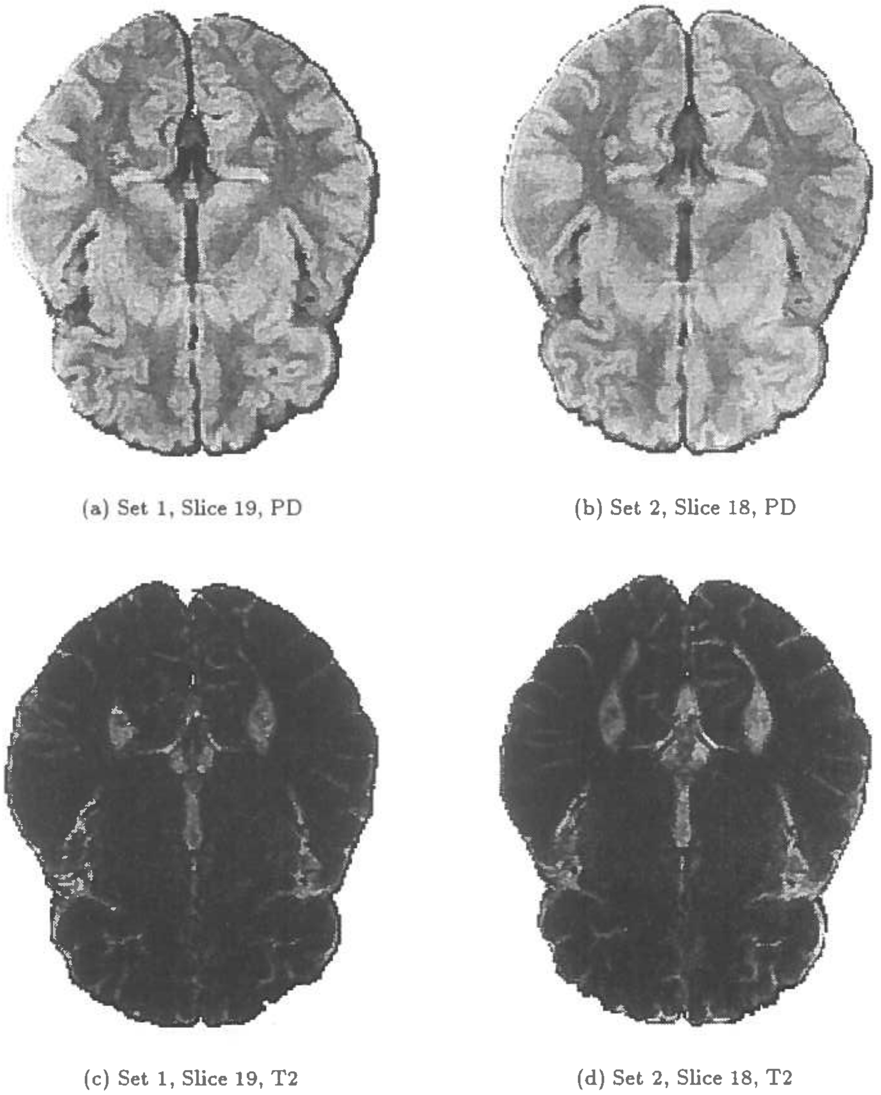
$\{x_i\}$. Correlation returns a value for ρ in the range $[-1,+1]$. $\rho = +1$ means the regions are perfectly correlated, with a value of zero denoting no correlation. The higher the value of ρ then the stronger the correlation. Strong correlation shows that for any pixel in the first region the associated pixel in the second region has a similar value with respect to the local standard deviation of each region. Negative correlation values show that the regions are anti-correlated, i.e. if any pixel in the first region has a high value then the associated pixel in the second region has a low value and vice versa.

Cross-correlation thus provides a similarity measure based on the structure of the neighbourhood pixel values and is robust to differences in contrast and absolute measurement level between the matched regions. The standard deviation of a neighbourhood provides a measure of contrast over that neighbourhood, and since the correlation evaluation (Equation (2.4)) normalises by dividing through by the local standard deviation of each region the effects of contrast differences are largely compensated for. Fisher and Oliver [3] suggest an adaptation to Equation (2.4) when applied to image region matching. Here we argue that, given the consistent nature of modern image capture devices, artificial discrepancies in contrast will be small (assuming the illumination distribution changes little), and so we would wish to penalise two regions whose contrast differed greatly. This is based on the assumption that contrast differences are due to image structure rather than artifacts introduced by the image acquisition process. The cross-correlation is modified to give:

$$\rho = \frac{1}{N} \sum_{i=1}^{N} \frac{(x_i - \overline{x})(y_i - \overline{y})}{\max(\sigma_x^2, \sigma_y^2)} \tag{2.5}$$

Now, if the two regions $\{x_i\}$ and $\{y_i\}$ are slight variations of the same scene then this change has little effect. If however, $\{x_i\}$ and $\{y_i\}$ have a marked difference in contrast ρ is pulled toward zero, causing decreased correlation.

A number of different image region similarity metrics have been proposed above. These all vary in computational complexity and sophistication. Woodward's region matching algorithm used the correlation metric given in Equation (2.4) ([10], p. 28), but took on average eight



(a) Set 1, Slice 19, PD

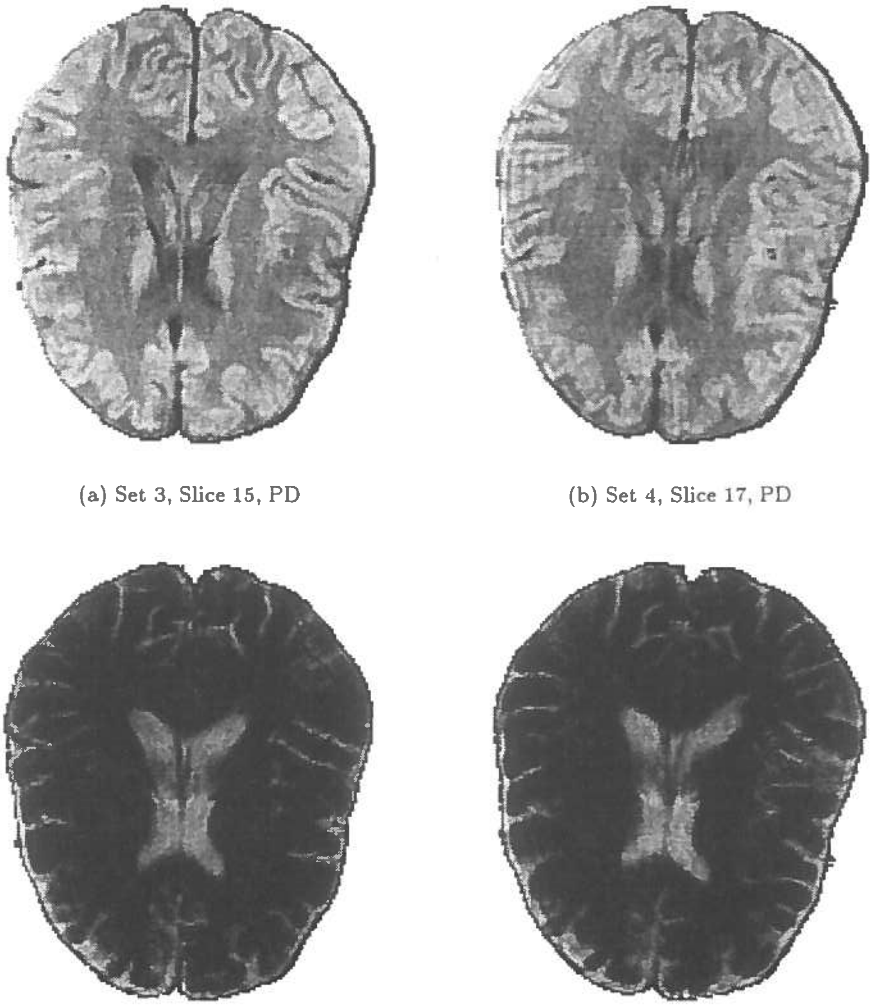
(b) Set 2, Slice 18, PD

(c) Set 1, Slice 19, T2

(d) Set 2, Slice 18, T2

Figure 2.1: Corresponding sections from brain data sets 1 and 2

hours to reach an initial match for each image pair ([10], p. 70). The next section details the implementation, testing and evaluation of the considered metrics, drawing conclusions on the best metric for this application, while later sections detail and evaluate generalisations of the metrics to cover the multi-variate (multi-channel) case, concluding this chapter with remarks upon suitability of the metrics with respect to the project as a whole.



(a) Set 3, Slice 15, PD

(b) Set 4, Slice 17, PD

(c) Set 3, Slice 15, T2

(d) Set 4, Slice 17, T2

Figure 2.2: Corresponding sections from brain data sets 3 and 4

2.3 Testing and Evaluation of Single Channel Matching

All five of the single channel image region matching metrics given in Section 2.2 as Equations (2.1), (2.2), (2.3), (2.4) and (2.5) have been implemented, tested and evaluated. This section presents the tests and experiments carried out to verify and evaluate the matching algorithms suggested by these metrics.

The general search framework when computing an initial match for all pixels is one of brute force search, with searching limited to a region about the expected match pixel. For this initial matching it is assumed that the two images are already approximately registered, and so the expected match pixel will occupy nearly the same position in the second image as it does in the first.

Since MRI data currently available has two channels (the PD and T2 images), evaluation will be carried out for three single channel cases: (1) PD only, (2) T2 only, and (3) for a composite image of the average of the PD and T2 images, i.e. for pixel values PD_i and $T2_i$, define the associated pixel value in the composite image as $C_i = (PD_i + T2_i)/2$, where i indexes pixels, and PD, $T2$ and C represent pixel values from the PD, the T2 and the composite images respectively. It is anticipated that generalised evaluation functions will yield better results than a single channel evaluation function fed with such generalised, or partial data.

Figures 2.1 and 2.2 show the test images. These images are used to verify that the matching algorithms work and have been implemented correctly.

The first set of tests takes an image, and translates it by a known amount for the second image. Three such tests are carried out: (1) on a PD image, (2) on a T2 image and (3) on an image that is the average of the T2-PD pair. Since the exact relation between the two images is easily determined, this is used to provide evidence of correctness.

The second set of tests modifies the correlation match algorithms to to maximise anti-correlation, and the second image is taken to be an exact negative of the first but translated by a known amount as above. Again this is used to provide evidence of correctness.

The third set of tests takes an image, and multiplies each pixel value by the same amount for the second image. This introduces a global contrast difference between the images. This is used to provide testing of region match performance in the face of contrast differences. The structure of the images remains the same; the contrast difference has therefore been isolated.

The fourth set of tests takes an image, and increases each pixel value by a constant amount for the second image. This introduces a global pixel intensity difference. It is used to provide evidence of correctness.

The fifth set of tests takes an image, and rotates it by a known amount for the second image. Since rotation of images is involved and the region match mask is not rotated in sympathy, this introduces disparity between the likeness of the matched regions and serves to provide a measure of the success of the similarity metrics when faced with similar but not identical regions.

The sixth set of tests examines the multiplicative effects of rotation, pixel value scaling and pixel value offset on image region matching.

The final bank of tests in this section uses a number of "real" MRI data sets taken from the patient at different times. This will provide a measure of how well the metrics cope with real not manufactured data.

These tests do not form a complete suite (for example the behaviour under small image magnification changes should be evaluated). They do however provide enough evidence to empirically show correctness of both the theory and implementation of the initial region matching process.

Unless indicated otherwise, the base test images are those shown in Figure 2.1. In some cases, these images have been transformed in known and detailed ways to provide artificial image artifacts used as obstacles during testing.

2.3.1 Interpreting the test results

In the following sections, the test results will be presented in tabular form. Interpret the entries as follows:

- **Mapping**: Gives the base images used for matching:

PD↦PD	An PD image is matched with another PD image
T2↦T2	A T2 image is matched with another T2 image
PDT2/2↦PDT2/2	A composite image made from the average pixel values of a T2 and PD image is matched with another similarly defined composite image
T2,PD↦T2,PD	A pair of images, one a T2, the other a PD are matched with a similar pair of images

- **Metric**: The region similarity metric used:

Contrast	The contrast penalised cross-correlation similarity metric given in Equation (2.5) or (2.11)
Cross	The cross-correlation similarity metric given in Equation (2.4) or (2.10)
Euclidean	The Euclidean metric given in Equation (2.1) or (2.7)
Manhatten	The Manhatten metric given in Equation (2.2) or (2.8)
l_∞	The l_∞ or L_∞ metric given in Equation (2.3) or (2.9)

- **Time**: Approximate timings (with a 486DX PC clone running at 50MHz).

- **Matches**: Number of pixels in the initial image for which a match has been computed.

- **Bad**: Number of bad matches.

- **Criteria**: Criteria for deciding if a match is bad. This figure represents the maximum allowed error (Euclidean distance measured in pixels) between the expected match and the actual match position.

- **Mean Error**: Mean error (in pixels) between the expected match and actual match positions.

- **Error S.Dev**: The statistical standard deviation (in pixels) of the differences between the expected match and actual match positions.

2.3.2 Test 1: Translation

This test shows the searching algorithm implemented is correct. It takes a number of images and translates them by a known constant amount. Since the local structure and pixel values remain unchanged between the original and translated images all the suggested similarity metrics should perform faultlessly (providing of course there are no regions of constant gradient greater in size than the matching mask).

For these tests, matching was performed with a 5×5 neighbourhood centred around each pixel, with search restricted to an 11×11 area about the (non-translated) initial position.

Tests were carried out on the following image sets: PD$\mapsto$PD, T2$\mapsto$T2, PDT2/2$\mapsto$PDT2/2 (Figure 2.1). Without exception each metric provided the same correct region matches for every one of the 17,876 pixels in the initial images. Furthermore, the correlation metrics returned a maximum correlation for every match as +1 (i.e. the strongest possible confidence that the match is good). The only other results of interest are the average timings for each of the strategies, presented below:

Metric	Time	Matches
Contrast	10 m 49 s	17,876
Cross	10 m 57 s	17,876
Euclidean	2 m 37 s	17,876
Manhatten	2 m 22 s	17,876
l_∞	2 m 24 s	17,876

Table 2.1: Test 1: Identical images

This test shows the metrics to be fundamentally correct in measuring the absolute likeness between two identical local regions. The difference in the timings reflect the computational complexity of the metrics.

2.3.3 Test 2: Inverted images

For this test the correlation based matching algorithms were modified to search for the greatest anti-correlation. The second image was an inverse of the first (like a photographic negative) but translated in the vertical direction by a small amount. Matching was performed with a 5 × 5 neighbourhood centred around each pixel, with search restricted to an 11 × 11 area about the (non-translated) initial position.

Without exception, each pixel was matched correctly with a correlation value of −1 for each match. The timings were comparable to those above.

This shows the implementation and theory to be correct in measuring anti-similarity between image regions. It should be noted that this test has little validity in determining the suitability of cross-correlation for image region matching over the other metrics presented since they too can be easily adapted to search for such anti-similarity.

2.3.4 Test 3: Contrast Differences

This test aims to compare the behaviour of the five different region similarity metrics when used with images of the same structure but where each pixel value has been multiplied by a constant. It is reasonable to assume that any image matching algorithm will encounter images taken with different hardware and even at different times. Such images may contain this sort of pixel scaling, both on a global and local scale.

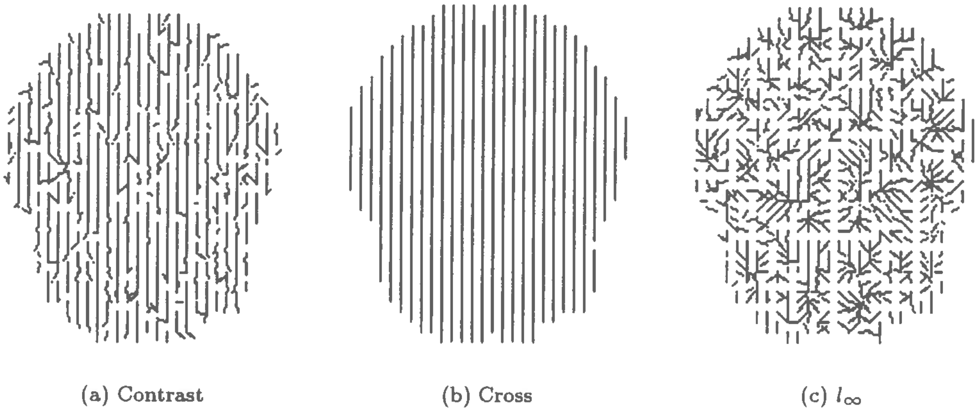
The test takes an initial image, multiplies each pixel value by 1.5 for the second image, and translates it by five pixels in the vertical direction. Matching was performed with a 5 × 5 neighbourhood centred around each pixel, with search restricted to an 11 × 11 area about the (non-translated) initial position.

If the similarity metric exhibited a random nature then chance alone would give around 17,686 bad matches. Typical performance of the initial matching is given below:

Mapping	**Metric**	**Time**	**Matches**	**Bad**	**Criteria**	**Mean Error**	**Error S.Dev**
T2↦T2	Contrast	10 m 45 s	17,834	5,121	0	1.06	2.27
T2↦T2	Cross	10 m 49 s	17,834	0	0	0.0	0.0
T2↦T2	Euclidean	2 m 37 s	17,834	16,802	0	6.34	3.35
T2↦T2	Manhatten	2 m 20 s	17,834	17,323	0	6.48	3.23
T2↦T2	l_∞	2 m 20 s	17,834	13,223	0	5.11	3.89

Table 2.2: Test 3: Pixel value scalar difference

Figure 2.3 shows the vector mapping from pixels in the first image to pixels in the second image for every fifth pixel for the top three results. An ideal map is exhibited by (b) where each pixel maps to a location five pixels below itself.



(a) Contrast

(b) Cross

(c) l_∞

Figure 2.3: Match maps for pixel value scalar difference test, showing (a) poor performance of the contrast penalised correlation metric, (b) perfect performance of the cross-correlation metric, and (c) partial failure of the l_∞ metric.

This shows a failure of all but the normalised cross-correlation metric. Perhaps a scalar of 1.5 is a much larger value than can be expected in 'real' data? This thought prompted a second test (test 3a) similar to above but with a scalar multiplier of 1.2 for the second image, the results of which are tabulated below.

Again chance alone would give around 17,686 bad matches.

Mapping	Metric	Time	Matches	Bad	Criteria	Mean Error	Error S.Dev
T2↦T2	Contrast	10 m 44 s	17,834	2,155	0	0.32	1.21
T2↦T2	Cross	10 m 47 s	17,834	0	0	0.0	0.0
T2↦T2	Euclidean	2 m 35 s	17,834	14,970	0	4.80	1.87
T2↦T2	Manhatten	2 m 20 s	17,834	16,503	0	5.21	3.33
T2↦T2	l_∞	2 m 22 s	17,834	7,324	0	2.43	3.48

Table 2.3: Test 3a: Pixel value scalar difference

The amount of scalar difference has been cut down by about half, and the results for the contrast penalised metric appear to reflect this. The Manhatten and Euclidean metrics show a smaller improvement, and the l_∞ metric shows an order of improvement similar to the contrast penalised metric. Both the Manhatten and Euclidean metrics sum an error over the match region, but the l_∞ metric takes the single largest error and finds a match where this is minimal.

The cross-correlation with normalisation for local contrast appears to be the frontrunner at this point. The two tests documented here show that all but the cross-correlation metric (Equation (2.4)) fail rapidly with increasing global pixel value scaling—suggesting that the others may work well with 'real' data.

Of course any such global pixel value scaling can be effectively removed by preprocessing the images, but it is difficult to devise a reliable scheme for correction of more localised scalar differences.

2.3.5 Test 4: Pixel value offset differences

This test takes an image and increases each pixel value by a constant amount for the second image. This is intended to simulate a global pixel value offset, such as may be present as an artifact of digitiser base level differences.

Typical results are tabulated below, in which the second image has a global pixel offset of +100 with respect to the first image. Matching was performed with a 5 × 5 neighbourhood centred around each pixel, with search restricted to an 11 × 11 area about the initial position. Chance alone would give around 17,657 bad matches.

Mapping	Metric	Time	Matches	Bad	Criteria	Mean Error	Error S.Dev
PD$\mapsto$PD	Contrast	10 m 47 s	17,804	0	0	0.0	0.0
PD$\mapsto$PD	Cross	10 m 52 s	17,804	0	0	0.0	0.0
PD$\mapsto$PD	Euclidean	2 m 38 s	17,804	13,281	0	4.34	3.69
PD$\mapsto$PD	Manhatten	2 m 22 s	17,804	15,398	0	4.91	3.51
PD$\mapsto$PD	l_∞	2 m 23 s	17,804	2,184	0	0.88	2.51

Table 2.4: Test 4: Pixel value offset difference

Both cross-correlation metrics are robust to constant global offset differences in pixel values. The remaining three metrics suffer badly, and are rendered useless by such large differences. The l_∞ result is somewhat better than may be expected when comparing with the Euclidean and Manhatten metric results.

2.3.6 Test 5: Global image rotation

This test takes a PD and a T2 image, and rotates each by five and ten degrees for the secondary images. This test should give an indication of the robustness of each similarity metric to small rotations between images.

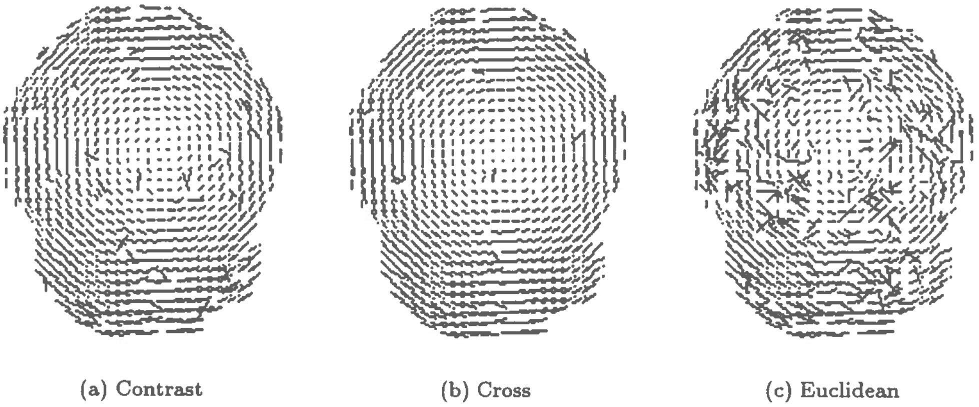
The results presented below were taken for a PD image rotated by five degrees and for a T2 image rotated by ten degrees. The PD test used a 5 × 5 match mask with search restricted to a 17 × 17 area about the initial position. The T2 test used a 5 × 5 match mask with search restricted to a 33 × 33 area about the initial position.

For the PD tests, chance alone would give around 16,062 bad matches. For the T2 tests, chance alone would give around 17,371 bad matches.

Mapping	Metric	Time	Matches	Bad	Criteria	Mean Error	Error S.Dev
PD↦PD	Contrast	25 m 22 s	17,804	1,217	3	0.65	1.44
PD↦PD	Cross	25 m 11 s	17,804	522	3	0.26	1.39
PD↦PD	Euclidean	5 m 56 s	17,804	4,867	3	2.47	3.22
PD↦PD	Manhatten	5 m 23 s	17,804	6,176	3	3.16	3.32
PD↦PD	l_∞	5 m 20 s	17,804	3,433	3	1.85	2.98
T2↦T2	Contrast	97 m 14 s	17,834	2,383	3	2.49	4.75
T2↦T2	Cross	98 m 8 s	17,834	1,955	3	2.13	4.51
T2↦T2	Euclidean	23 m 14 s	17,834	7,983	3	5.35	5.80
T2↦T2	Manhatten	20 m 23 s	17,834	10,550	3	6.19	5.95
T2↦T2	l_∞	20 m 25 s	17,834	10,288	3	5.95	5.82

Table 2.5: Test 5: Global image rotation through five degrees (top) and ten degrees (bottom)

This shows breakdown of the contrast penalised correlation metric when faced with even small rotations of five degrees. To assess bad vectors a residual vector map was calculated for each initial mapping by estimating and removing the global image rotation (Chapter 4). Any match greater than three pixels in length was deemed bad. The match maps for a sub-sample of the matches found for the contrast penalised correlation, cross-correlation, and Euclidean metrics on the PD tests are shown in Figure 2.4. A perfect map would show no vectors deviating from the anticlockwise flow.


(a) Contrast

(b) Cross

(c) Euclidean

Figure 2.4: Match maps for image rotation test, showing (a) good performance of the contrast penalised correlation metric, (b) excellent performance of the cross-correlation metric, and (c) poor performance of the Euclidean metric.

2.3.7 Test 6: Combined global image artifacts

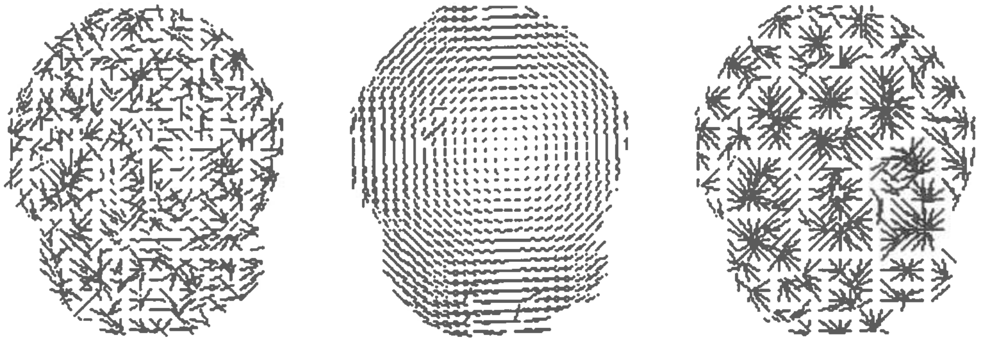
This set of tests explores the multiplicative effect of rotation, pixel value scaling and pixel value offset on region based image matching.

For the T2 test, the initial image has pixel values in the range 0 to 1067, while the second image is a copy of the first, rotated anticlockwise by 5 degrees but with rescaled pixel values in the range 200 to 455. A 5 × 5 match mask was used, with search restricted to a 17 × 17 area about the initial position. Mappings were computed for every second pixel in the horizontal and vertical directions. For the PD test, the initial image has pixel values in the range 0 to 1533, while the second image is a copy of the first, rotated anticlockwise by 10 degrees but with rescaled pixel values in the range 0 to 255. A 5 × 5 match mask was used, with search restricted to a 17 × 17 area about the initial position. Mappings were computed for every foreground pixel.

Chance alone would give around 4,020 bad matches for the T2 tests and around 17,371 for the PD tests.

Mapping	Metric	Time	Matches	Bad	Criteria	Mean Error	Error S.Dev
T2↦T2	Contrast	6 m 27 s	4,456	3,421	3	5.68	2.93
T2↦T2	Cross	6 m 17 s	4,456	95	3	0.18	1.17
T2↦T2	Euclidean	1 m 31 s	4,456	4,081	3	7.56	2.69
T2↦T2	Manhatten	1 m 22 s	4,456	4,074	3	7.64	2.69
T2↦T2	l_∞	1 m 22 s	4,456	4,004	3	6.95	2.69
PD↦PD	Contrast	97 m 14 s	17,834	13,320	3	13.19	5.28
PD↦PD	Cross	98 m 8 s	17,834	1,945	3	2.12	4.51
PD↦PD	Euclidean	23 m 14 s	17,834	17,172	3	12.21	4.85
PD↦PD	Manhatten	20 m 23 s	17,834	17,207	3	12.25	4.58
PD↦PD	l_∞	20 m 25 s	17,834	17,465	3	12.91	4.78

Table 2.6: Test 6: Matching performance with combined global image artifacts



(a) Contrast

(b) Cross

(c) Manhatten

Figure 2.5: Match maps for the multiple global image artifact test, showing (a) breakdown of the contrast penalised correlation metric, (b) excellent performance of the cross-correlation metric, and (c) breakdown of the Manhatten metric.

Figure 2.5 shows a sub-sample of the match maps for three of the metrics in the T2 tests. The star-like patterning of the Manhatten map (and the Euclidean and l_∞ maps also—not shown) has an explanation as follows: The first and second images have the same structure, but with differently scaled and offset pixel intensity distributions. The first image has minimum and maximum intensity values of 0 and 1067 respectively, while the second image has minimum and maximum values of 200 and 455. Say, for the sake of argument, that the first image region has an average pixel intensity of $(0 + 1067)/2 = 533.5$ and the second image region has an average of $(200 + 455)/2 = 327.5$, then any matching strategy that aims to minimise a function of the difference between the two regions will by definition favour areas where this function is small (in this case, where the differences are small). Now where the pixel values of the second image rise above the average, they will be closer on average to the pixel values of the first image. Matches are favoured towards these points, manifesting vector mappings that characteristically point towards high intensity areas of the second image. This behaviour is especially observable in Figure 2.5 (c).

2.3.8 Test 7: Real Data

The preceding tests were concerned with the testing of the proposed image region similarity metrics, and as such featured manufactured artifacts within the secondary image. Such a bank of tests would be incomplete without any evalutation of performance on real data sets.

This test is concerned with the evaluation of the similarity metrics when applied to region based matching of real MRI data.

The results presented are taken from tests carried out on data collected from the same brain at four separate times. These are typical of the data sets currently available, and so the results tabulated are also typical.

The PD test used a 5 × 5 match mask with search restricted to an 11 × 11 area about the initial position. The T2 test used data taken at different times to the PD test and used a 9 × 9 match mask with search restricted to a 25 × 25 area about the initial position. To quantify bad vectors a residual vector map was calculated for each initial mapping by estimating and removing the global image rotation (see Chapter 4). Any match greater than three pixels in length for the PD test and greater than five for the T2 test was deemed bad. The PD images show little change over time and hence there is strong correlation over most of the image (Figure 2.1 (a) and (b)). The T2 images (Figure 2.2 (c) and (d)) were chosen as they embody a large scale change and so provide for evidence of effectiveness towards the limits of expected use.

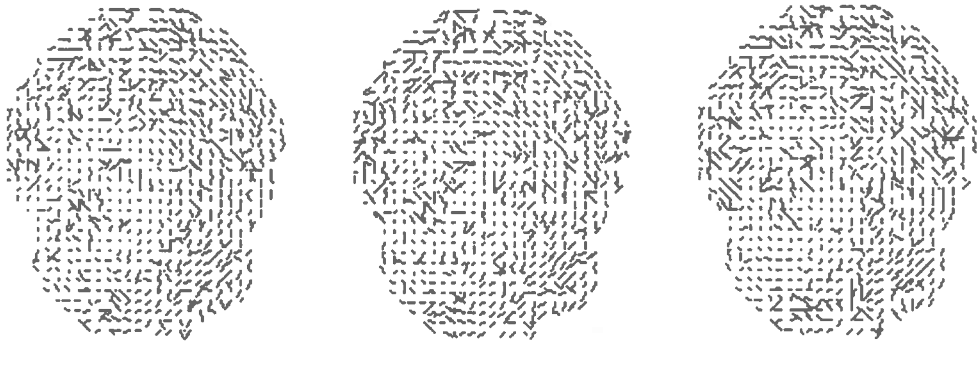
Of course not all matches counted as bad actually are bad since, in this real data, sections of brain have moved relative to one another between images.

Chance alone would give about 13,644 bad matches for the PD tests and about 15,501 for the T2 tests.

Mapping	Metric	Time	Matches	Bad	Criteria	Mean Error	Error S.Dev
PD↦PD	Contrast	10 m 54 s	17,804	4,862	3	2.13	2.06
PD↦PD	Cross	10 m 59 s	17,804	4,701	3	2.05	2.13
PD↦PD	Euclidean	2 m 38 s	17,804	5,205	3	2.14	2.13
PD↦PD	Manhatten	2 m 23 s	17,804	5,376	3	2.28	2.13
PD↦PD	l_∞	2 m 23 s	17,804	5,521	3	2.33	2.09
T2↦T2	Contrast	169 m 58 s	17,729	4,896	5	4.30	4.39
T2↦T2	Cross	169 m 50 s	17,729	4,960	5	4.42	4.57
T2↦T2	Euclidean	41 m 18 s	17,729	5,244	5	4.37	4.75
T2↦T2	Manhatten	37 m 0 s	17,729	5,546	5	4.59	4.77
T2↦T2	l_∞	36 m 19 s	17,729	4,964	5	4.33	4.51

Table 2.7: Test 7: 'Real data'

All tests on real data so far have given image matching somewhat better than that due to chance alone, with the correlation metrics performing slightly better than the other three. Given that the simpler metrics are some five times faster than the correlation metrics consideration must be given to the question of whether the improved results are worth the time trade off in practice. Figure 2.6 shows a sub-sampling of the match vectors for the PD tests. A perfect match map would show the vectors swirling in an anticlockwise direction about a point to the left of the centre of the image.



(a) Contrast

(b) Cross

(c) Euclidean

Figure 2.6: Match maps for the PD real data test, showing similar performance of the region similarity metrics.

From 2.6 it can be seen that (with this data at least, and in the absence of 'known' results) there is little to choose from between the matching techniques.

2.3.9 Conclusions

The fundamental region matching problem is one of designing a good similarity metric. The five metrics presented here exhibit a trade off between quality of matching and speed. The faster metrics suffer badly when presented with similar images that are rotated, or have different pixel intensity distributions. Assuming any rotations encountered are small and therefore can be disregarded for the moment then a standard model [3] for the intensity difference between two images is:

$$B_i = \alpha A_j + \beta$$

Where i and j index corresponding pixels, A and B are the intensity levels of the pixels from the first and second images respectively, α is the gain (or scalar) difference, and β is the offset difference. Differences in illumination and digitiser hardware can be responsible for this kind of difference. The Euclidean, Manhatten and l_∞ metrics assume that the images being matched have the same intensity distributions. The contrast penalised and cross-correlation metrics cope well with this model since they normalise for both offset and scalar differences by (a) subtracting the mean and (b) dividing by the standard deviation. Both correlation metrics exhibit good offset normalisation (Table 2.4). The contrast penalised metric modifies the contrast normalisation term (c.f. Equations (2.4) and (2.5)) to favour those matches between regions of similar overall contrast. With artificial image artifacts, the contrast penalised metric can easily be made to behave almost as poorly as the simpler metrics (Table 2.6), but with real data it is still unclear whether such penalisation is a good thing (Table 2.7). In fact, with the real MRI data considered it may be the case that little advantage is gained from the correlation metrics over the faster ones once further constraints aimed at improving the matches are imposed.

Since the two correlation metrics divide through by the standard deviation of each region the bad effects of a global pixel scalar difference (global contrast difference) is limited, and with the cross-correlation metric these effects are effectively completely removed. However, the contrast penalised correlation metric actively discriminates against contrast differences in an attempt to be more selective at the local level. The failure here may be due to too much discrimination, suggesting a modification of Equations (2.4) and (2.5):

$$\rho = \frac{1}{N} \sum_{i=1}^{N} \frac{(x_i - \overline{x})(y_i - \overline{y})}{(1-\alpha)\sqrt{\sigma_x^2 \sigma_y^2} + \alpha \max(\sigma_x^2, \sigma_y^2)} \tag{2.6}$$

When used for image region matching, $\alpha \in [0, 1]$ is a gain constant, where $\alpha = 0$ gives no discrimination against contrast differences, and $\alpha = 1$ gives maximum discrimination.

2.4 Multi-Channel Image Matching

In the preceding sections five single channel image region similarity metrics have been presented and empirically evaluated. When applied to image region matching the metrics based around the statistical normalised correlation function (Equations (2.4) and (2.5)) are more robust to unwanted global image artifacts (e.g. small rotations between images and dissimilar pixel value intensity distributions), than the simpler difference error metrics (Equations (2.1), (2.2) and (2.3)). In practice, however, global artifacts are few and, in the case of differing pixel intensity distributions, can easily be removed by pre-processing (assuming that the two images are of the same view). This gives rise to almost comparable behaviour between all the metrics, except that the correlation based metrics are some five times slower.

The following sections generalise the image region similarity metrics already given in Section 2.2 to deal with multiple channels such as the T2-PD pairs that the MRI data is organised into, or the three RGB channels of colour stereo imagery. Such similarity metrics work with vectors of pixel values, and this section will show that these vector based metrics give better region matching than their scalar based single channel counterparts.

2.4.1 Multi-Channel Euclidean Metric

Let $\vec{x_i}$ be a vector of pixel samples, one element from each channel of the initial image. Let $\vec{x_i}$ have dimension M where M is the number of channels. Similarly let $\vec{y_i}$ be a vector of matched pixels in the second image. Then if samples are taken over a neighbourhood of size N, define the extended Euclidean similarity metric as:

$$E = \sqrt{\sum_{j=1}^{M} \sum_{i=1}^{N} (\vec{x}_{ij} - \vec{y}_{ij})^2} \tag{2.7}$$

where j indexes the individual elements of the vectors. The best region match is given by choosing a neighbourhood $\{\vec{y_i}\}$ that minimises E for a given $\{\vec{x_i}\}$. This function reduces to the single channel case of Equation (2.1) when $M = 1$.

2.4.2 Multi-Channel Manhatten Metric

As above, let $\{\vec{x_i}\}$ and $\{\vec{y_i}\}$ of dimension M define neighbourhood samples of size N from two multi-channel images, then the extended Manhatten similarity metric is given by:

$$L_1 = \sum_{j=1}^{M} \sum_{i=1}^{N} | \vec{x}_{ij} - \vec{y}_{ij} | \tag{2.8}$$

The best region match is given by choosing a neighbourhood $\{\vec{y_i}\}$ that minimises L_1 for a given $\{\vec{x_i}\}$. This function reduces to the single channel case of Equation (2.2) when $M = 1$.

2.4.3 Multi-Channel Largest Absolute Difference Metric

Again, let $\{\vec{x_i}\}$ and $\{\vec{y_i}\}$ of dimension M define neighbourhood samples of size N from two multi-channel images, then the extended largest absolute difference metric is given by:

$$L_\infty = \sum_{j=1}^{M} \max_i (|\vec{x}_{ij} - \vec{y}_{ij}|) \tag{2.9}$$

The best region match is given by choosing a neighbourhood $\{\vec{y_i}\}$ that minimises L_∞ for a given $\{\vec{x}\}$. This function reduces to the single channel case of Equation (2.3) when $M = 1$.

2.4.4 Multi-Channel Normalised Cross-Correlation Metric

None of the preceding three multi-variate similarity metrics give equal weight to each channel. The channel with the largest pixel value fluctuations will dominate. The cross-correlation metrics normalise within each channel, causing each to have equal weight. For this reason it is anticipated that the correlation metrics will give more correct matches.

Geiss *et. al.* [5] considered a multi-variate cross-correlation function between two multi-variate signals. In a recent paper [3] we apply a modification of this function to image region matching. This modified function is presented below and used as the multi-variate cross-correlation image region similarity metric for multi-channel imagery.

Define the mean vector of the N $\vec{x_i}$ (dimension M) vector samples as:

$$\hat{\vec{x}} = \frac{1}{N}\sum_{i=1}^{N}\vec{x}_i$$

and similarly for $\hat{\vec{y}}$. $\vec{x_i}$ is a vector of pixel intensity values taken from the first (multi-channel) image region, and $\vec{y_i}$ is a vector of pixel intensity values from the second image. i indexes pixels over the respective regions of size N. Now define the vector $\vec{\Lambda_x}$ to hold the local variance as:

$$\vec{\Lambda_x} : \left\{ \vec{\Lambda}_{xj} = \frac{1}{N}\sum_{i=1}^{N}(\vec{x}_{ij} - \hat{\vec{x}}_j)^2, \forall j \in \{1,\ldots,M\} \right\}$$

and similarly for $\vec{\Lambda_y}$, where j indexes the elements of the vectors up to dimension M. The multi-variate normalised cross-correlation function is then given by:

$$\rho = \frac{1}{M}\sum_{j=1}^{M}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{(\vec{x}_{ij} - \hat{\vec{x}}_j)(\vec{y_i}_j - \hat{\vec{y}}_j)}{\sqrt{\vec{\Lambda}_{xj}\vec{\Lambda}_{yj}}}\right) \tag{2.10}$$

The best region match is given by choosing a neighbourhood $\{\vec{y_i}\}$ that maximises ρ for a given

$\{\vec{x_i}\}$. Similarly to the one-dimensional case (Equation (2.4)) ρ lies within the range $[-1, +1]$ and has a similar interpretation.

Note that this function (2.10) normalises for pixel value offset differences by subtracting the mean, and normalises for pixel value gain differences by dividing by the standard deviation. Such normalisation is done for each channel. This function reduces to the single channel case of Equation (2.4) when $M = 1$.

For the same reasons given for the single-channel case (Section 2.2.4), the multi-variate cross-correlation metric can be modified to penalise contrast differences:

$$\rho = \frac{1}{M} \sum_{j=1}^{M} \left(\frac{1}{N} \sum_{i=1}^{N} \frac{(\vec{x}_{ij} - \hat{\vec{x}}_j)(\vec{y}_{ij} - \hat{\vec{y}}_j)}{\max(\vec{\Lambda}_{xj}, \vec{\Lambda}_{yj})} \right) \tag{2.11}$$

reducing to the single channel case of Equation (2.5) when $M = 1$.

During evaluation of the one-dimensional case of these two correlation based similarity metrics, a combination of them was proposed and is given in Equation (2.6). This too can similarly be extended to the multi-variate case as follows:

$$\rho = \frac{1}{M} \sum_{j=1}^{M} \left(\frac{1}{N} \sum_{i=1}^{N} \frac{(\vec{x}_{ij} - \hat{\vec{x}}_j)(\vec{y}_{ij} - \hat{\vec{y}}_j)}{(1-\alpha)\sqrt{\vec{\Lambda}_{xj}\vec{\Lambda}_{yj}} + \alpha \max(\vec{\Lambda}_{xj}, \vec{\Lambda}_{yj})} \right)$$

again with $\alpha \in [0, 1]$ acting as a parameter that specifies the amount of contrast penalisation. The idea for this combined metric came too late in the project to allow for adequate evaluation, and so is merely presented here in the hope that it may be implemented and evaluated at a later date.

2.5 Testing and Evaluation of Multi-Channel Matching

The multi-variate image region matching algorithms presented above have been tested and evaluated. From the preceding discussions, in which the multi-variate versions of the one-dimensional similarity metrics have been introduced, it can be seen that the multi-variate case is a straightforward generalisation. This means that the new metrics inherit the flaws and strengths from the uni-variate cases. It has been seen that the correlation based metrics provide a normalised similarity measure that can compensate for global image intensity distribution differences, and hence provide a more robust region similarity measure than the simpler (and faster) Euclidean, Manhatten and l_∞ metrics. It has also been seen that *in practice* there is little to choose from between the metrics if the quality of data is good, and the image distributions are identical—except that the correlation based metrics perform marginally better but at a cost of five times the run-time.

The testing presented here therefore explores the benefits of extending the metrics to M-dimensions over the simpler case of compressing the M-dimensional data to one-dimension and using the simpler metrics.

For guidance on interpretation of the results tabulated below, refer to Section 2.3.1.

2.5.1 Test 8: Global Image Rotation Revisited

In the uni-variate global rotation case of test 5 (Section 2.3.6), the two correlation based metrics gave many fewer bad matches than the simpler metrics. In this next test, the same images are used but with the multi-variate generalised metrics. Here, each test takes four images (as opposed to two in the previous tests) (i.e. two PD,T2 pairs).

The first of the two rotation tests takes a second image that is rotated through five degrees anti-clockwise from the first, and the second test takes the same images but rotated through ten degrees.

The results from these tests are tabulated below in Tables 2.8 and 2.9. In each table the uni-variate case PDT2/2$\mapsto$PDT2/2 (in which a single averaged image replaces a PD,T2 pair) is tabulated below the bi-variate PD,T2$\mapsto$PD,T2 case.

For the five degrees test, both the PD,T2 and PDT2/2 tests used a 5 × 5 match mask with search restricted to a 17 × 17 area about the initial position in an image rotated through five degrees. Chance alone would give about 15,921 bad matches for the PD,T2 tests and about 16,231 for the PDT2/2 tests.

Mapping	Metric	Time	Matches	Bad	Criteria	Mean	Error
						Error	S.Dev
PD,T2↦PD,T2	Contrast	46 m 8 s	17,647	329	3	0.19	1.11
PD,T2↦PD,T2	Cross	46 m 24 s	17,647	154	3	0.09	0.80
PD,T2↦PD,T2	Euclidean	11 m 52 s	17,647	321	3	0.17	1.04
PD,T2↦PD,T2	Manhatten	10 m 49 s	17,647	428	3	0.21	1.13
PD,T2↦PD,T2	L_∞	9 m 30 s	17,647	405	3	0.24	1.30
PDT2/2↦PDT2/2	Contrast	25 m 31 s	17,991	1,257	3	0.66	2.09
PDT2/2↦PDT2/2	Cross	25 m 42 s	17,991	496	3	0.25	1.32
PDT2/2↦PDT2/2	Euclidean	6 m 5 s	17,991	1,230	3	0.61	1.91
PDT2/2↦PDT2/2	Manhatten	5 m 31 s	17,991	1,758	3	0.88	2.22
PDT2/2↦PDT2/2	l_∞	5 m 32 s	17,991	1,624	3	0.87	2.27

Table 2.8: Test 8: Rotation through five degrees (revisited)

Of importance to this test is the pixel intensity distributions of the four images used. They have all been "normalised" (by linearly compressing the the range of pixel values) so that all pixel values fall within the range $[0, 255]$. (This is also the case for test 5). This means that any benefits the normalised correlation based metrics may have in compensating for global intensity distribution differences have been reduced to a minimum. This is significant since, as such global differences can be removed easily by pre-processing the images, the normalisation property of the correlation based metrics may not be necessary to ensure good overall matches. In other words it may be possible to gain similarly good matches from the simpler metrics if the 'real data' is first 'normalised'. This would yield a matching algorithm almost as good, but some five times faster. This is of prime importance once image match improvement constraints are imposed (Chapter 3) as this will usually call for repeated region matching. This argument is based on the assumption that *local* contrast differences (e.g. an illumination gradient) are negligible. Should such local differences exist, then clearly the normalised cross-correlation metrics have the advantage since they normalise on a local scale.

The second set of tests take the same images, but rotated through ten degrees. The results are tabulated below in Table 2.9. The uni-variate case PDT2/2↦PDT2/2 is tabulated below the bi-variate PD,T2↦PD,T2 case. Both the PD,T2 and PDT2/2 tests used a 5 × 5 match mask with search restricted to a 33 × 33 area about the initial position. Chance alone would give about 17,189 bad matches for the PD,T2 tests and about 17,524 for the PDT2/2 tests.

Mapping	**Metric**	**Time**	**Matches**	**Bad**	**Criteria**	**Mean Error**	**Error S.Dev**
PD,T2↦PD,T2	Contrast	170 m 30 s	17,647	665	3	0.66	2.98
PD,T2↦PD,T2	Cross	170 m 44 s	17,647	561	3	0.54	2.73
PD,T2↦PD,T2	Euclidean	44 m 37 s	17,647	1,941	3	1.52	3.91
PD,T2↦PD,T2	Manhatten	39 m 55 s	17,647	2,680	3	2.09	4.37
PD,T2↦PD,T2	L_∞	34 m 28 s	17,647	2,420	3	2.28	4.43
PDT2/2↦PDT2/2	Contrast	98 m 33 s	17,991	2,747	3	2.83	4.79
PDT2/2↦PDT2/2	Cross	99 m 13 s	17,991	2,169	3	2.38	4.58
PDT2/2↦PDT2/2	Euclidean	23 m 28 s	17,991	11,373	3	6.31	5.85
PDT2/2↦PDT2/2	Manhatten	21 m 26 s	17,991	13,320	3	7.13	5.93
PDT2/2↦PDT2/2	l_∞	21 m 22 s	17,991	11,770	3	6.42	5.78

Table 2.9: Test 8: Rotation through ten degrees (revisited)

These rotation tests have demonstrated the abilities of the image similarity metrics when used on rotated MRI brain images under perfect conditions. The significant conclusions are twofold, firstly *all* the addressed multi-variate similarity metrics perform better under rotation than their uni-variate counterparts (i.e. compared with the results in Table 2.5), and secondly the order of improvement for each of the metrics has been roughly constant, with the uni-variate cases showing some four times more bad matches than the bi-variate cases. In [3] a strong argument is presented that shows the multi-variate cross-correlation metric used here to be correct. This argument carries through to the issue of correctness of the other multi-variate metrics developed here. As each of them shows improvement of the order of four times, then this is supportive evidence that these other metrics have also been correctly generalised. As will be seen in the next test, such a high order of improvement is not to be expected when dealing with less than perfect real data.

2.5.2 Test 9: Real Data Revisited

Tests on manufactured rotation of images (above) revealed substantial improvements in the number of correct matches found when moving from uni-variate matching to bi-variate matching. The cost of this is to double the run-time over the uni-variate case, but to improve the quality of matching some four times (under ideal conditions). Tests on real data show similar but not so drastic improvements.

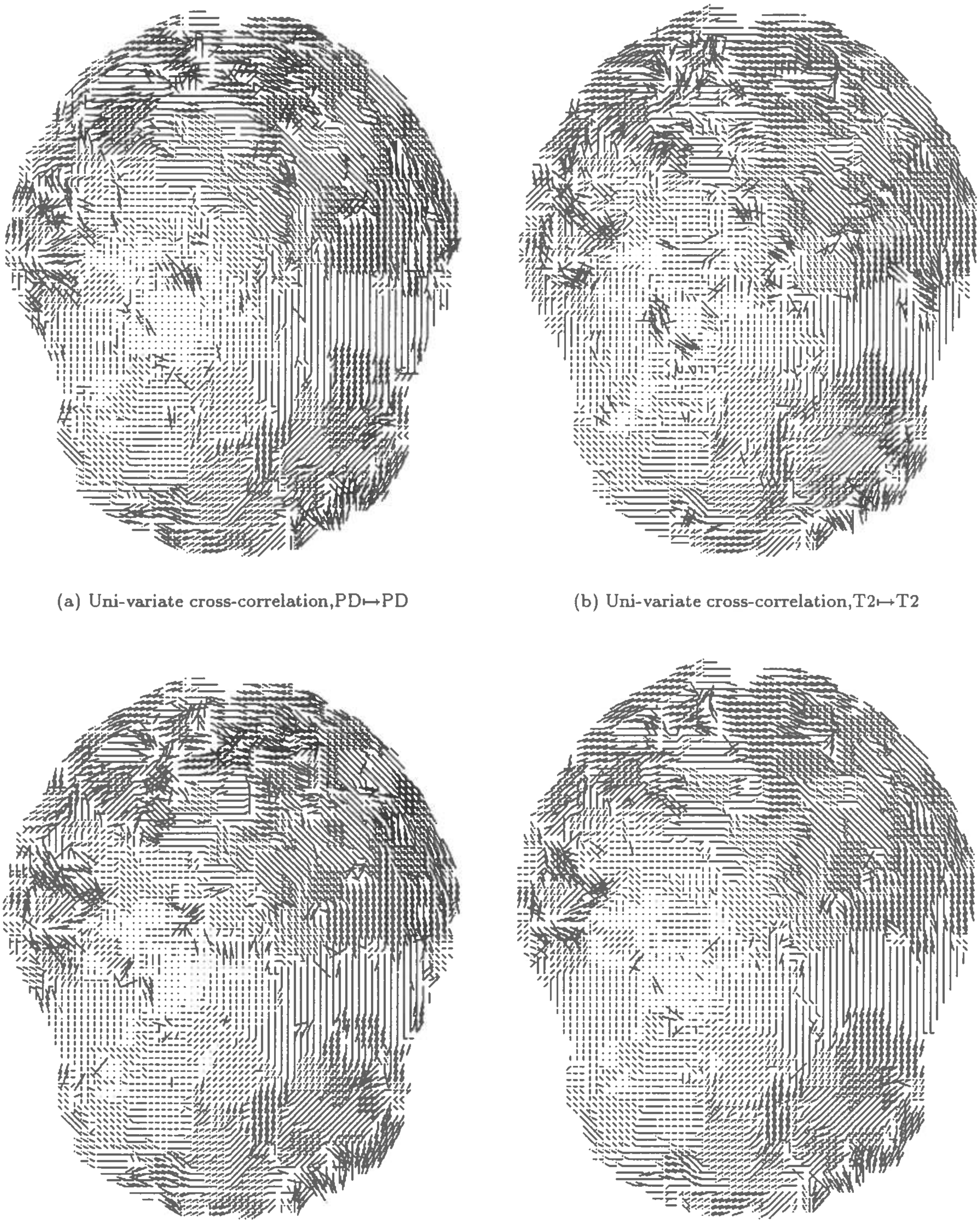
Table 2.10 shows the results for bi-variate matching of the brain images shown in Figure 2.1 and, comparing this to the uni-variate results (shown both in Tables 2.7 and 2.10), a substantial improvement can be seen. For this test a 5 × 5 match mask was used, with search restricted to an 11 × 11 area about the initial position. Chance alone would give about 13,523 bad matches for the PD,T2↦PD,T2 tests and around 13,787 for the PDT2/2↦PDT2/2 tests.

Mapping	Metric	Time	Matches	Bad	Criteria	Mean	Error
						Error	S.Dev
PD,T2↦PD,T2	Contrast	19 m 48 s	17,647	3,220	3	1.58	1.93
PD,T2↦PD,T2	Cross	19 m 48 s	17,647	3,199	3	1.54	1.99
PD,T2↦PD,T2	Euclidean	5 m 24 s	17,647	3,869	3	1.78	2.04
PD,T2↦PD,T2	Manhatten	4 m 54 s	17,647	4,011	3	1.82	2.06
PD,T2↦PD,T2	L_∞	4 m 21 s	17,647	3,889	3	1.79	1.98
PDT2/2↦PDT2/2	Contrast	10 m 39 s	17,991	4,888	3	2.13	2.13
PDT2/2↦PDT2/2	Cross	10 m 42 s	17,991	4,637	3	2.02	2.12
PDT2/2↦PDT2/2	Euclidean	2 m 35 s	17,991	5,254	3	2.24	2.35
PDT2/2↦PDT2/2	Manhatten	2 m 21 s	17,991	5,509	3	2.33	2.15
PDT2/2↦PDT2/2	l_∞	2 m 21 s	17,991	5,551	3	2.32	2.09

Table 2.10: Test 9: Bi-variate matching (top) and uni-variate matching (bottom) of images in Figure 2.1

All metrics show notably improved performance (especially L_∞) over the uni-variate cases.

Figure 2.7 compares match maps for uni-variate and bi-variate cases of the brain images given in Figure 2.1. Matches were computed for every second pixel in the horizontal and vertical directions, using a 9 × 9 match mask, with search restricted to an 11 × 11 area about the initial position. A perfect match would show all vectors swirling anti-clockwise around a point to the left of the centre of the image. Bad matches are evident as cross-hatchings on the maps. Clearly the bi-variate case shows good improvement over the uni-variate cases.



(a) Uni-variate cross-correlation,PD↦PD

(b) Uni-variate cross-correlation,T2↦T2

(c) Uni-variate cross-correlation,
PDT2/2↦PDT2/2

(d) Uni-variate cross-correlation,PD,T2↦PD,T2

Figure 2.7: Comparison of match maps obtained by matching every second pixel of the images given in Figure 2.1. Single channel mapping (a) and (b) show many bad matches, visible as areas where long vectors create cross hatching patterns. Combining two single channels into one 'averaged' channel before matching (c) shows no fewer bad matches. Treating each channel separately (d) and using a bi-variate matching algorithm produces many fewer bad matches.

For the next test (Table 2.11) a 9 × 9 match mask was used, with search restricted to a 25 × 25 area about the initial position. Chance alone would give about 15,429 bad matches for the PD,T2↦PD,T2 tests and around 15,642 for the PDT2/2↦PDT2/2 tests.

Mapping	Metric	Time	Match	es Bad	Criteri	aMean	Error
						Error	S.Dev
PD,T2↦PD,T2	Contrast	307 m 44 s	17,553	3,426	5	3.28	4.04
PD,T2↦PD,T2	Cross	306 m 16 s	17,553	3,438	5	3.43	4.28
PD,T2↦PD,T2	Euclidean	82 m 45 s	17,553	7,745	5	6.15	5.22
PD,T2↦PD,T2	Manhatten	74 m 30 s	17,553	8,105	5	6.37	5.30
PD,T2↦PD,T2	L_∞	64 m 55 s	17,553	4,972	5	4.35	4.52
PDT2/2↦PDT2/2	Contrast	173 m 23 s	17,890	5,603	5	4.83	4.58
PDT2/2↦PDT2/2	Cross	172 m 44 s	17,890	5,068	5	4.51	4.55
PDT2/2↦PDT2/2	Euclidean	42 m 22 s	17,890	11,117	5	7.74	4.80
PDT2/2↦PDT2/2	Manhatten	38 m 6 s	17,890	11,690	5	8.03	4.74
PDT2/2↦PDT2/2	l_∞	37 m 15 s	17,890	9,289	5	6.80	4.78

Table 2.11: Test 9: Bi-variate matching (top) and uni-variate matching (bottom) of images in Figure 2.2

Again, a very strong improvement can be seen by the L_∞ over the uni-variate l_∞ case. Without exception, moving to a multi-variate similarity metric will improve the quality of matches for any of the metrics explored here. With real MRI data, the most correct matches have been found by the cross-correlation metric (**Cross**), but tests have also shown that often one or more of the simpler, faster, metrics may be almost as good with MRI data.

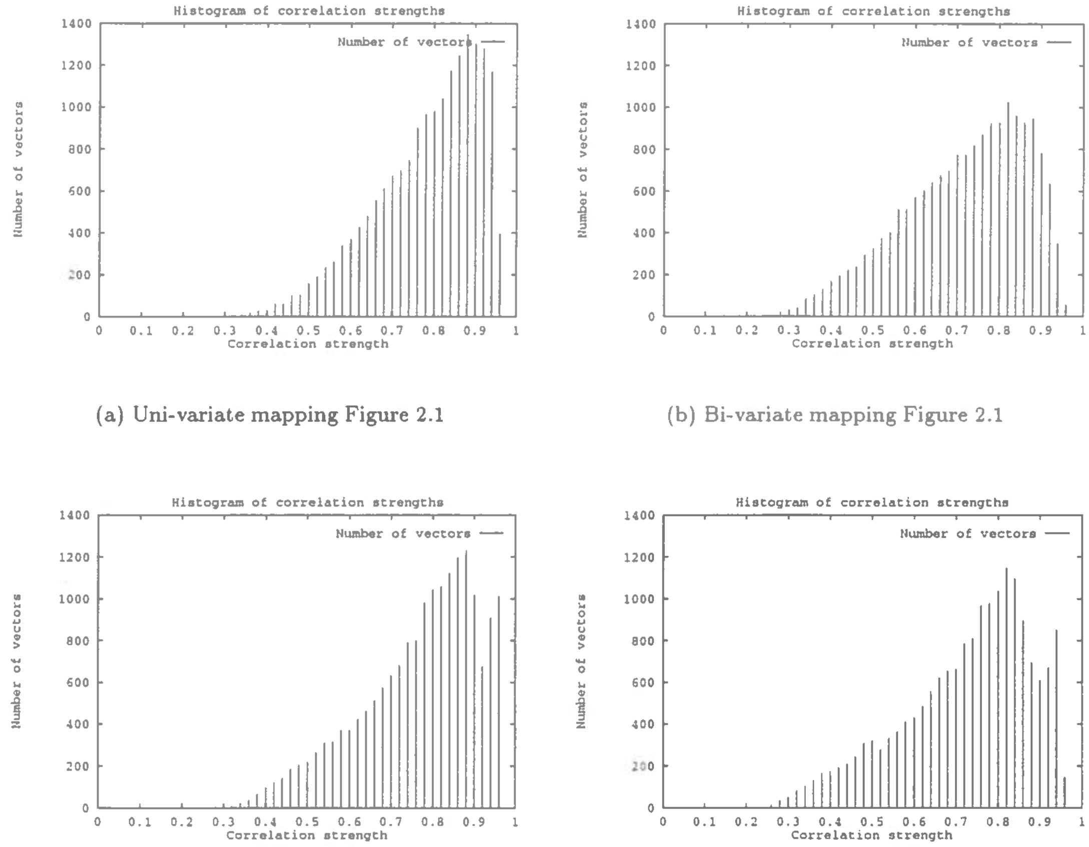
2.5.3 Summary and Conclusions

In this chapter, five uni-variate image region similarity metrics have been presented and evaluated within the context of region matching within MRI brain scan data. Of these metrics, only the normalising cross-correlation metric of Section 2.2.4 (Equation (2.4)) has been shown to be robust in a wide range of circumstances. This metric compensates for contrast differences between images. However, in practice, it is sometimes more prudent to penalise contrast differences—in which case the contrast penalised correlation metric of Section 2.2.4 (Equation (2.5)) performs slightly better. This has suggested a new metric which is given in Section 2.3.9 as Equation (2.6). The correlation based metrics provide more correct matches than the simpler metrics presented here. However, correlation takes some five times longer to perform than any of the other three simpler metrics (Euclidean, Manhatten and l_∞), and initial experiments with real data suggest that one or more of these simpler metrics may perform adequately once further improvement constraints are imposed (Chapter 3).

Woodward ([10], p. 70) took on average eight hours to match two simple images using the cross-correlation similarity metric of Equation (2.4). He used a pair of 128 × 128 images with eight bits per pixel. The current MRI data used for this project is some 256 × 256 with sixteen bits per pixel, but match times for a pair of images average under eleven minutes (e.g. Table 2.1). The reasons for this improvement are numerous. Firstly, more advanced modern hardware may have given a speed improvement of a factor of two. Woodward's code was far from optimal, and the removal of a few small array index checks provides another factor of two improvement ([10], p. 71). Further improvements have been made by calculating mappings for only foreground pixels, and then by keeping these calculations to a minimum. Woodward's code made many unnecessary repeated calculations. These speed improvements have made change detection by image region similarity a practical technique.

Woodward used only uni-variate MRI data. The availability of multi-channel MRI imagery has led to the development of multi-variate image region similarity metrics for this project. Each of the uni-variate metrics has been extended to handle multi-variate image data. Tests with these multi-variate image region similarity metrics have shown a general improvement over their uni-variate counterparts. The reason for this is that the multi-variate metrics use all the available data whereas the uni-variate metrics must use either a single channel from the multi-channel images, or use some single channel that represents a compressed multi-channel image. In either case not all the available information is used. In all cases the multi-variate generalisation of the metrics is a straightforward operation, resulting in region matching that takes a time linearly dependent on the number of channels being matched. In the focus of this project, two channel MRI data matching is improved by the use of bi-variate image region similarity metrics.

Both the uni-variate and multi-variate correlation metrics produce a measure of the likeness of two regions. This measure is within the range $[-1, +1]$ (see Section 2.2.4). The higher the measure, the more similar the two regions. When applied to image region matching, the match with the strongest correlation measure is chosen. This figure can act as a measure of confidence in the match. As improvement in the number of matches has been observed when moving up to a uni-variate case, it is reasonable to expect that the confidence of the matches will generally be greater (i.e. that the metric will have a better 'idea' of what makes a good match). Figure 2.8 shows a comparison of the histograms of correlation values between uni-variate and bi-variate matching. What seems to be happening here is a general migration to lower correlation values in the bi-variate cases. This is probably a side effect of the bi-variate matching function averaging the correlations from two channels. Because of such averaging, the chances of a highly ranked (but possibly bad) match decreases.



(a) Uni-variate mapping Figure 2.1

(b) Bi-variate mapping Figure 2.1

(c) Uni-variate mapping Figure 2.2

(d) Bi-variate mapping Figure 2.2

Figure 2.8: Comparison of histograms of correlation strengths for matched pixels using (a), (c) uni-variate correlation, and (b), (d) bi-variate correlation

Tests with real data show that the normalised cross-correlation metrics give a larger number of correct matches. The simpler metrics often come close to the quality of the correlation metrics, but rarely supersede them. This still leaves the question of which metric is best in this situation. If the number of correct matches is the most important aspect then clearly the cross-correlation metric is the best choice, yet the simpler metrics run some five times faster and often approach the quality of the correlation metrics. Further image region matching constraints may reduce the quality divide between the simpler metrics. In which case one of the faster metrics may suffice to give a reasonable change detection system that runs with reasonable speed.

Multi-modal computer assisted tomography (CAT) data is now becoming commonplace. The physician or neurophysiologist may now have a selection of image data available, from any one patient, obtained from different scanning modalities (e.g. CT, PET, and MRI scans). Pelizzari *et. al.* [8] give a system for the accurate three dimensional matching of such multi-modal data. Since different modalities highlight different structures of interest, such registered data is an ideal input into the multi-variate matching algorithms developed here.

Consideration must be given to the size and shape of match mask used. Larger masks will give more reliable matches, but at the expense of speed and localisation. In practice, a 9 × 9 mask seems a good size, but for the purpose of comparative testing a 5 × 5 mask has been used for the tests presented here. Woodward ([10], p. 30) used a roughly circular match mask, under the argument that this would be more robust to rotational differences between images. Woodward produced no evidence of analysis of the behaviour of differently sized and configured match masks, and little analysis is presented here. This is left as a suggestion for further work.

Finally, attention is drawn to the consistently good performance of the cross-correltation metric (**Cross**). Not only does this return the fewest bad matches, but the mean target placement error and standard deviation are consistently lower than with other metrics. This suggests the normalised cross-correlation metric to be the most stable and robust.

Chapter 3

Improving the initial match map

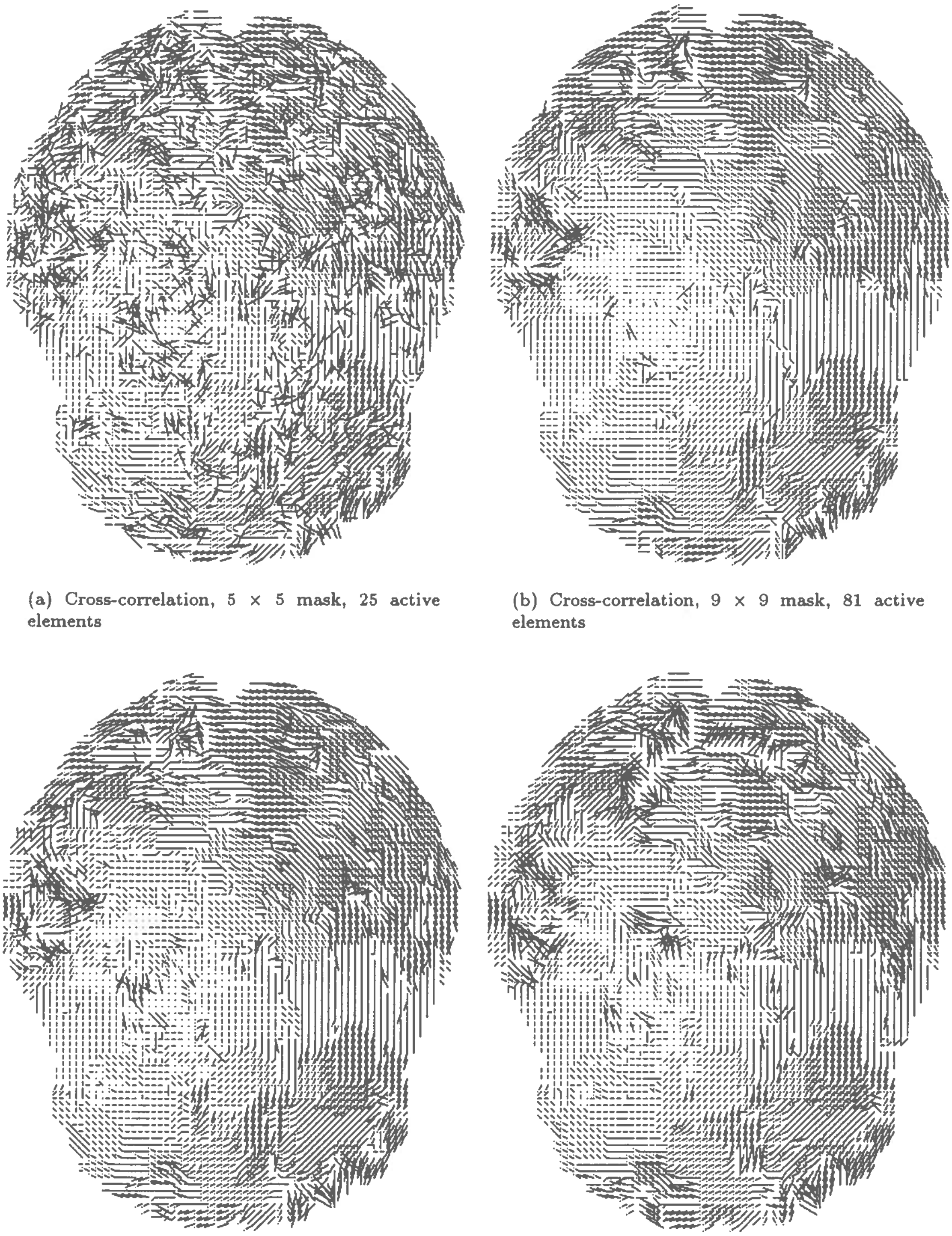
Chapter 2 describes how a brute force search of the second image is used to arrive at an initial image region match map for any pixel in the first image. This map is composed of a number of directed line segments which each specify the probable movement of pixel-sized areas of the brain between the first image and the second. For convenience of notation these directed line segments are known as *pixel movement vectors*. The initial mapping process is often not reliable enough to form a basis on which to analyse and detect brain structure changes over time. For example, many bad matches are generated in practice (Chapter 2, Tables 2.10 and 2.11). Even if these bad matches are detected and removed they will still distort any analysis performed. Woodward ([10], p. 37) used a post-operative smoothing algorithm, based on vector neighbourhoods, to 'swing' around vectors pointing in different directions to their neighbours. This has proved unsatisfactory for two reasons: (1) it assumes that all neighbouring vectors are pointing in the correct direction—iterating in order to work to the centre of whole groups of mis-aligned vectors, and (2) badly aligned vectors will affect correctly aligned vectors. Also, the image structure being mapped to is not consulted in any form during this smoothing. When two similar image are being initially mapped, each vector is obtained by a search aimed at maximising some local image region similarity function. If a region fails to map to the correct target region (and can be identified as such), then choosing the next best similar region may provide the correct mapping—iterating until all bad vectors have been identified and corrected.

The aim of this chapter is then to explore methods of improving initial image match mapping. Such improvement may be of two forms: (1) the match map may have fewer bad matches, and (2) the match map may be arrived at more quickly.

By reducing the surface area of the image region match mask, speed improvements can be made. This is the subject of the first part of this chapter. The remainder of the chapter looks at match map improvement aimed at reducing the number of bad matches. This is done by analysis of neighbourhoods of matches and by mapping to sub-pixel levels.

3.1 Improving region mapping speed

Although speed improvements of the order of 4,400% have been made over Woodward's [10] system, to produce an exhaustive initial match map for a pair of two channel MRI images with a fair degree of reliability can still take between five minutes and three hours. Time taken increases with the complexity of the region similarity metric, match mask size and search area. Choice of similarity metric depends upon the quality of data. Choice of search area depends both on initial image registration and the amount of brain structure migration between images. The choice of match mask size involves a localisation trade-off, with faster mapping a side effect of smaller match masks. The smaller the mask, the more local the structure movement that can be detected. However, the smaller the mask, the more susceptible to image noise and local ambiguities it will be, and hence, more bad matches will be found (i.e. coincidental bad matches).



(a) Cross-correlation, 5 × 5 mask, 25 active elements

(b) Cross-correlation, 9 × 9 mask, 81 active elements

(c) Cross-correlation, 9 × 9 mask, 25 active elements

(d) Euclidean, 9 × 9 mask, 25 active elements

Figure 3.1: Test 10: Bi-variate matching of images in Figure 2.1. (a) Shows a poor match map returned by a 5 × 5 match mask with 25 active elements, using the multi-variate cross-correlation metric. (b) Shows a good match map returned by moving to a 9 × 9 match mask with 81 active elements—at the expense of three times the run-time of (a). (c) Shows a good match map returned by a 9 × 9 match mask with only 25 equidistant active elements (in a chequer-board pattern), using the multi-variate cross-correlation metric—runtime is same as for (a). (d) Shows a good match map returned by a the same match mask as (c), but using a multi-variate Euclidean metric—speeding runtime by a factor of five over (c).

The choice of match mask affects not just the initial mapping, but also the post-operative smoothing of the map (since the original images are reconsulted).

This section shows that by spreading out the distribution of a match mask, so that it becomes a grid, speed improvements can be made with negligible loss of locality and number of good matches.

The following table details the results from three tests on the real data shown in Figure 2.1. Firstly, an initial map is produced using a 5 × 5 mask; secondly a 9 × 9 mask is used; and thirdly a 9 × 9 mask with only 25 equally spaced active elements (i.e. a grid shaped mask, like a chequers board) is used. In all cases, search is restricted to an 11 × 11 area about the initial position, and chance alone would give about 3,380 bad matches. Matches are computed for every second pixel in the horizontal and vertical directions. For interpretation of the table headings, refer to Chapter 2 Section 2.3.1.

Mapping	Metric	Time	Matches	Bad	Criteria	Mean Error	Error S.Dev
PD,T2↦PD,T2	Cross	4 m 47 s	4,411	788	3	1.54	1.99
PD,T2↦PD,T2	Euclidean	1 m 19 s	4,411	965	3	1.77	2.04
PD,T2↦PD,T2	Cross	15 m 42 s	4,411	202	3	0.57	1.26
PD,T2↦PD,T2	Euclidean	4 m 3 s	4,411	389	3	0.91	1.63
PD,T2↦PD,T2	Cross	4 m 57 s	4,411	228	3	0.68	1.38
PD,T2↦PD,T2	Euclidean	1 m 19 s	4,411	425	3	0.99	1.65

Table 3.1: Test 10: Bi-variate matching of images in Figure 2.1. First test (top row) uses a match mask with 25 active elements, covering a 5 × 5 square area. Second test (middle row) uses a match mask with 81 active elements, covering a 9 × 9 square. Third test (bottom row) uses a match mask with 25 equidistant active elements covering a 9 × 9 square. Significant speed improvements can be seen in the third test with little loss of quality.

A significant improvement over run-time, with little cost to the number of correct matches, has been gained by using a match mask in which only alternate elements are active. Table 3.1 shows this improvement. When a 5 × 5 match mask with 25 active elements is used, processing is quick, but the number of bad matches is unacceptably high. When a 9 × 9 match mask with 81 active elements is used a good number of correct matches are returned, but at a cost of higher run-time. However, when a 9 × 9 match mask with only 25 equidistant active elements is used, processing is quick while incurring little loss in the number of correct matches returned.

Figure 3.1 shows some of the initial match maps produced during these tests.

Only a small number of tests have been presented here. While they indicate that advantages are to be gained when using 'thinner' match maps, they do not provide conclusive evidence. Certainly the effects of the shape, size and coverage of match masks needs to be further investigated.

3.2 Improving the Number of Correct Matches

Section 3.1 has shown how the runtime of image region matching can be speeded up with judicious choice of region match mask. In this next section, attention is turned towards increasing the number of correct matches, both by matching to sub-pixel levels and by providing a post-operative vector map smoothing algorithm.

3.2.1 Image Region Matching to Sub-Pixel Levels

Movement of local brain structure between images is often subtle, and rarely by whole pixel quantities. Any of the vector maps shown in Figure 3.1 show the effects of quantisation to whole pixel levels by the 'blocky' appearance of vector groups. Often vectors are only one or two pixels in length, severely restricting the accuracy to which mappings can be built. This section details the development of a sub-pixel matching algorithm which, at the cost of longer run-times, allows sub-pixel matching to an arbitrary level.

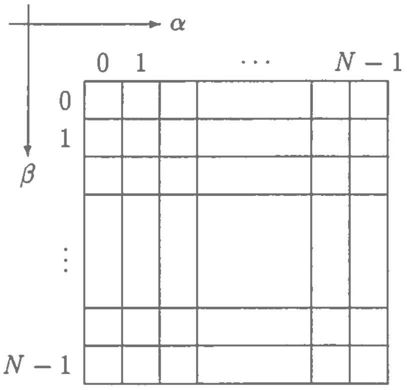
The algorithm has two basic stages. Firstly, the target image is enlarged, interpolating new sub-pixel values, and secondly, match maps are built taking vector start points as whole pixel indices from the first image and mapping to whole pixel indices of the interpolated, enlarged image. Vector end-points are then transformed from the enlarged image coordinate frame into the start image coordinate frame. The result is vector mappings from whole pixels to sub-pixel levels (quantised to the degree of enlargement).

The initial stage of the sub-pixel matching algorithm takes the target image, and enlarges it by a whole pixel scale value. New pixel values are interpolated from the original target image. The scale value used is also the sub-pixel level. For example, if the desired sub-pixel level is 4, then each pixel in the target image is notionally sub-divided into $4 \times 4 = 16$ sub-pixels. This is equivalent to enlarging the target image by 4 in both the horizontal and vertical directions. The choice to enlarge the target image (instead of interpolating sub-pixel values as region matching progresses) provides for a faster matching algorithm since no sub-pixel is interpolated more than once.

Choice of interpolation function will affect the quality of the new target image, and hence the quality of match map returned. Linear interpolation (based on the Gouraud shading model [4] p. 736) is used. The parameters to the interpolation function are the four pixel values a, b, c and d taken from each 2×2 neighbourhood in the original image:

a	b
c	d

and two pixel location indices α and β used to index into the enlarged neighbourhood. Assume that this 2×2 neighbourhood is to be interpolated into an $N \times N$ neighbourhood:

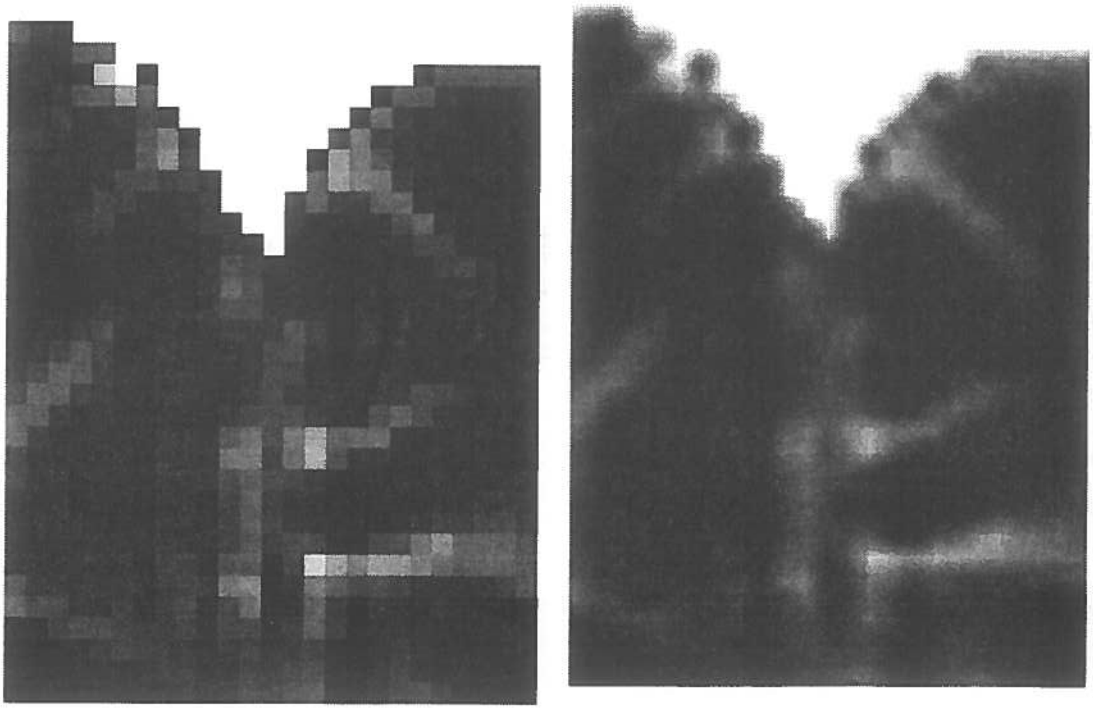


then, letting $n = N - 1$, each new pixel value at location (α, β) is given by:

$$I(\alpha, \beta) = \left(1 - \frac{\beta}{n}\right)\left[\left(1 - \frac{\alpha}{n}\right)a + \frac{\alpha}{n}b\right] + \frac{\beta}{n}\left[\left(1 - \frac{\alpha}{n}\right)c + \frac{\alpha}{n}d\right]$$

At new locations $(0,0)$, $(N-1,0)$, $(0,N-1)$, $(N-1,N-1)$, the new pixel values are exactly a, b, c and d respectively, and any pixel value indexed by $0 \leq \alpha < N$ and $0 \leq \beta < N$ is a linear combination of these four initial values.

Figure 3.2 shows such an interpolated enlargement, compared to a simple pixel-wise enlargement.



(a) Pixel-wise enlargement

(b) Enlargement with linear interpolation

Figure 3.2: Image enlargement with linear interpolation.

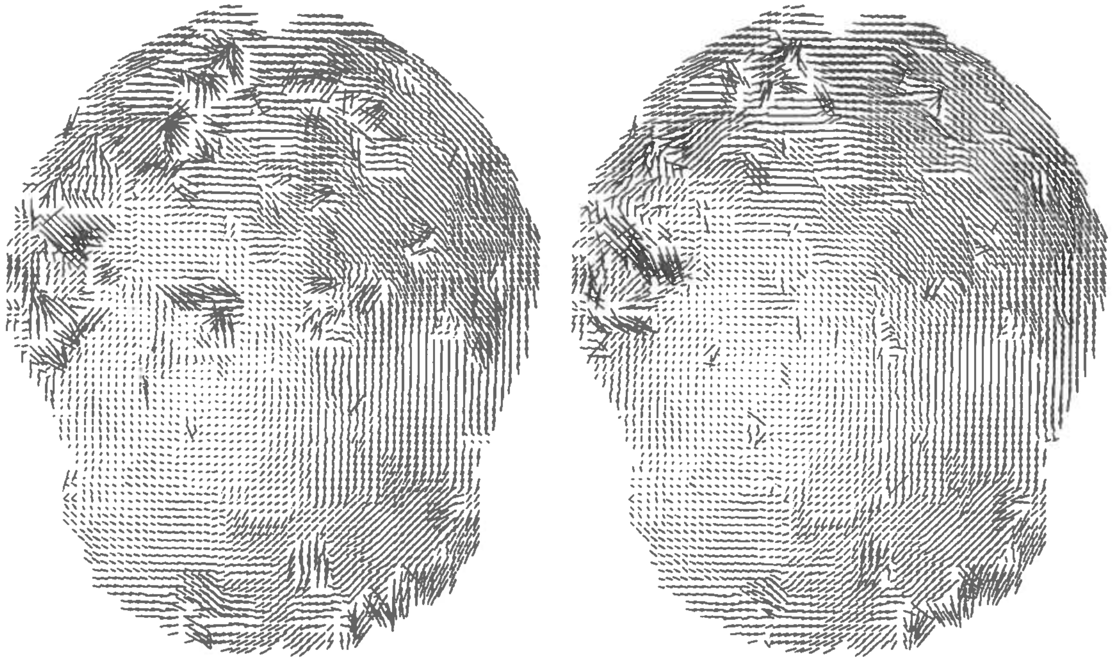
When computing matches to such an enlarged image, an appropriately scaled and sub-sampled target match mask is used. Thus, at every $(N-1)^{th}$ location in either direction, the mask will be aligned such that the pixel values it covers are exactly the same as the normal mask would cover on the original image.

Table 3.2 and Figure 3.3 detail some results from the testing of the sub-pixel matching strategy. Firstly, bi-variate tests on globally rotated images are presented. Here the sub-pixel algorithm is shown to operate better than the whole-pixel algorithm (Table 2.5). Secondly, results from bi-variate tests on real data are presented. Although performance seems statistically no better than the whole pixel algorithm (Table 2.10), Figure 3.3 does shows a marked improvement in the general direction, 'flow' and placement of vectors when compared with Figure 2.7. For the rotation tests a 9 × 9 match mask with 25 active elements was used with search restricted to a 17 × 17 area, sub-pixel level is 4 (i.e. each original pixel is replaced by a 4 × 4 group). Chance alone would give 3,898 bad matches. For the real data tests a 9 × 9 match mask with 81 active elements was used with search restricted to an 11 × 11 area, sub-pixel level is 4. Chance alone would give about 3,380 bad matches. In both cases matches were computed for every second pixel in the horizontal and vertical directions.

Mapping	Metric	Time	Matches	Bad	Criteria	Mean Error	Error S.Dev
PD,T2↦PD,T2	Cross	187 m 25 s	4,321	21	3	0.04	0.37
PD,T2↦PD,T2	Euclidean	43 m 44 s	4,321	35	3	0.07	0.56
PD,T2↦PD,T2	Cross	242 m 21 s	4,411	206	3	0.50	1.27
PD,T2↦PD,T2	Euclidean	55 m 24 s	4,411	404	3	0.87	1.65

Table 3.2: Test 11: Sub-pixel bi-variate matching of image in Figure 2.1 (a) and (b), with second image a copy of first rotated through five degrees (top). Sub-pixel bi-variate matching of real data in Figure 2.1 (bottom). Mappings have been computed for every second pixel in the horizontal and vertical directions.

Figure 3.3 shows two of the match maps obtained during testing. The bi-variate Euclidean match function took almost one hour to arrive at a reasonable match map, while the bi-variate cross-correlation match function took some four hours to arrive at a good match map with far fewer bad matches.



(a) Real data, sub-pixel level 4, bi-variate Euclidean, 9 × 9 mask, 81 active elements

(b) Real data, sub-pixel level 4, bi-variate cross-correlation, 9 × 9 mask, 81 active elements

Figure 3.3: Test 11: Bi-variate matching of images in Figure 2.1 to sub-pixel level. Sub-pixel matching has been achieved by dividing each original target pixel into 16 sub-pixels whose values have been linearly-interpolated. (a) Shows the match map from the bi-variate Euclidean matching function, and (b) shows the match map from the bi-variate cross-correlation matching function.

The whole-pixel level has been shown to be too coarse to permit subtle change detection and analysis in currently available MRI data. At the expense of increased run-times, target images can be interpolated to arbitrary sub-pixel levels, and match vectors computed to these sub-pixels. A sub-pixel level of four (i.e. each whole pixel is replaced by 4 × 4 = 16 interpolated sub-pixels) gives much improved match maps, but in practice sub-pixel level 2 may provide a good time trade-off.

3.2.2 Identifying and Dealing With Bad Matches

The accuracy of correctly pixel movement vectors has been greatly improved by sub-pixel matching (Section 3.2.1), but a large number of false matches are still generated in practice. For example, both the match maps of Figure 3.3 show many such false matches, evident as areas of cross hatching where the longer vectors cross paths. Such bad matches will adversely affect any attempted change detection. If bad matches can be identified, it is possible to either remove them from any further analysis, or (preferably) recompute their mappings with additional constraints imposed from correctly mapped neighbouring vectors. This section details the development of a strategy for identifying possible bad matches, and outlines an extension to a data driven re-mapping strategy for badly placed matches.

Any movement of brain structures over time will alter the total structure of the brain, but only very rarely will the relative local structure change radically. For example, if three colinear local features f_1, f_2 and f_3, are identified in the source image, then, even though colinearity may be disturbed in the target image, it is highly unlikely that the local ordering will change. Figure 3.4 illustrates this point. Detection of possibly bad vectors could be achieved through searching for adjacent vectors which have crossing paths (Figure 3.4 (b)). Unfortunately, such an approach has two downfalls. Firstly, which one of the two vectors are bad (if not both)?, and secondly, well matched adjacent vectors often cross, as illustrated by Figure 3.5. An alternative strategy for

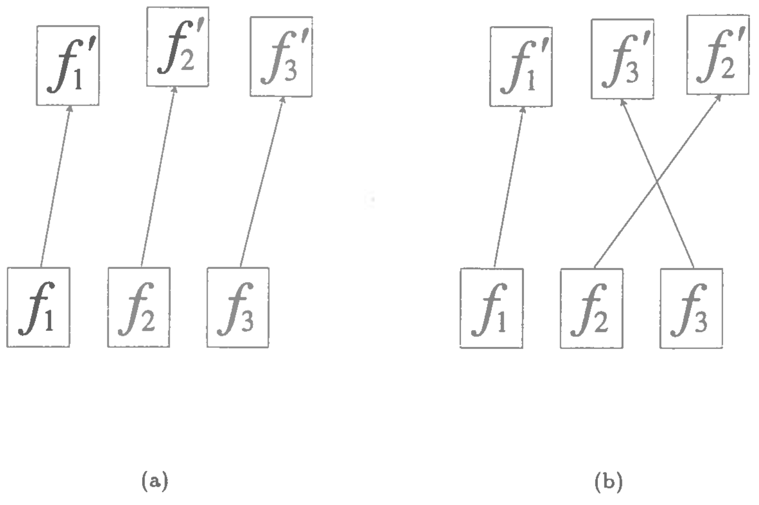


Figure 3.4: Local feature ordering is usually preserved between source and target images (a). Detection of possible bad matches can be made by looking for crossed vector paths (b)

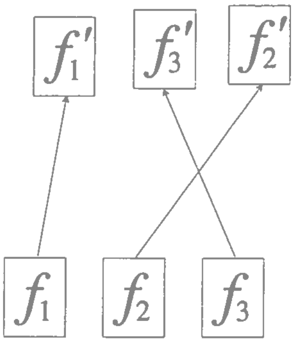


Figure 3.5: Detection of bad matches by crossed vector paths often fails under normal circumstances.

detecting bad matches has been developed from the following three local structure assumptions:

1. For every local structure in the source image, an identifiable variation of the same structure is present in the target image.
2. Changes in the structure of local features between images will be small (i.e. similar local structures between images will in fact be variations of the same structures).
3. The ordering of local features in the horizontal and vertical directions will be preserved between images.

Under assumptions one and two, a bad match is due to coincidental local ambiguities (a case of mistaken identity). If such a bad match is identified, it may be corrected by re-mapping it to the next best match, iterating to reach a final solution. A final solution will always be found under assumption one. Although reasonable, assumption one may be restrictive in practice. Change detection may not only involve determining the movement of brain structures over time, but it may require that new or missing structures are identified. Assumption three allows the development of a bad match detection strategy based on the *convex hull* of the target mappings of neighbouring vectors.

Cormen *et. al* ([2], p. 898) define the convex hull of a set of points as follows:

> The convex hull of a set Q of points is the smallest convex polygon P for which each point in Q is either on the boundary of P or in its interior.

They go on to say:

> Intuitively, we can think of each point in Q as being a nail sticking out from a board. The convex hull is then the shape formed by a tight rubber band that surrounds all the nails.

Figure 3.6 illustrates how the convex hull of the vector target points may be used in the detection of bad matches. Here, the convex hull of the eight-neighbourhood of the vector under analysis is determined. Should the target point of the vector fall outside of this convex hull, then assumption three must be broken and the vector deemed a bad match. Of course this is not a complete

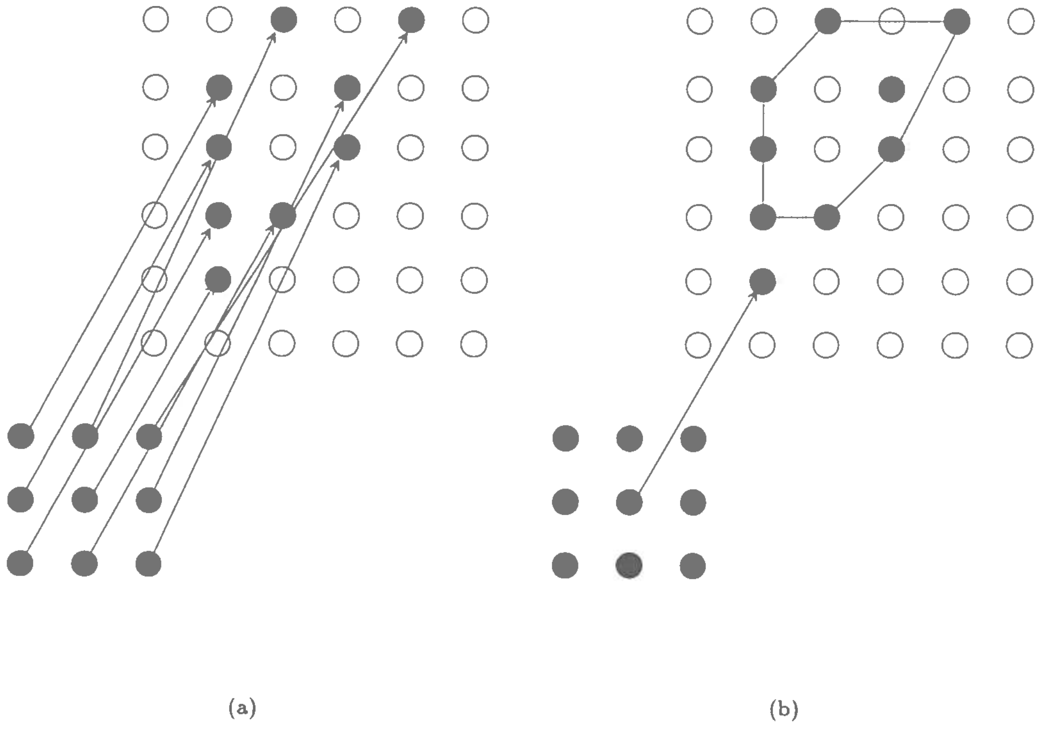


Figure 3.6: Detection of bad matches by violation of a convex hull constraint. Analysing local neighbourhoods of vectors is difficult, with many good match vectors showing crossing paths (a). Comparing the vector target point with the convex hull of the target points of it's neighbours provides an easy solution. A match is bad if it's target point falls outside of the convex hull of the target points of it's immediate neighbours (b).

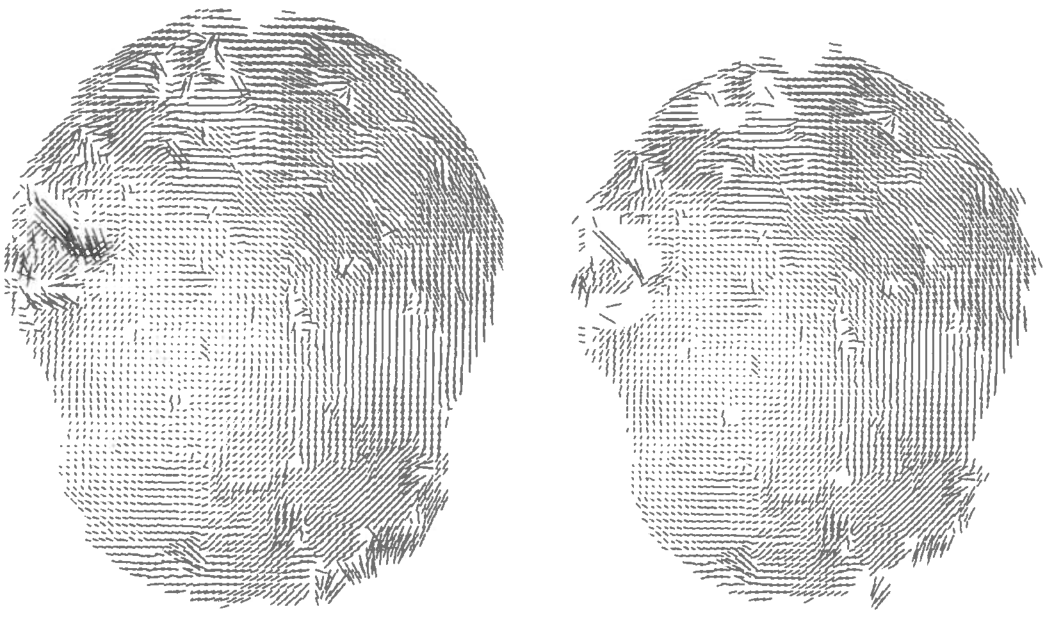
strategy. Bad matches may well exist within the convex hull, and good matches may well exist outside of the convex hull if the local image structure has changed so much as to violate assumption three.

Graham's scan ([2], p. 899) is used to compute the convex hull. Graham's scan first sorts all points by polar angle with respect to the left-lowest point. Each point is then taken in turn (anti-clockwise) and the angle between it and it's two preceding neighbours is considered at each step. If this angle involves a right hand turn, then an internal point has erroneously been included. This point is then removed from consideration, and the algorithm iterates to a solution. In pseudo-code, using a stack of points, the algorithm is:

```
GRAHAMS-SCAN($Q$)
let $p_0$ be the point in $Q$ with the minimum $y$-coordinate,
        or the leftmost such point in case of a tie
let $\{p_1, p_2, \ldots, p_m\}$ be the remaining points in $Q$,
        sorted by polar angle in counterclockwise order around $p_0$
        (if more than one point has the same angle, remove all but
        the one that is farthest from $p_0$)
$top[S] \leftarrow 0$
PUSH($p_0, S$)
PUSH($p_1, S$)
PUSH($p_2, S$)
for $i \leftarrow 3$ to $m$
    do while the angle formed by points NEXT-TO-TOP($S$),
                TOP($S$), and $p_i$ makes a nonleft turn
        do POP($S$)
    PUSH($S, p_i$)
return $S$
```

Once the convex hull is computed, a simple test can be used to determine whether or not any point lies within, or outside of the hull. Since the convex hull is described by an ordered set of boundary points, then if-and-only-if the point under consideration lies on or to the same side of every edge of the boundary, it lies within the convex hull.

Figure 3.7 illustrates the typical effects of this strategy. Here each bad match detected (by application of the convex hull strategy) has been removed. Input to the test is the match map shown in Figure 3.3 (b).



(a) One iteration

(b) Five iterations

Figure 3.7: The convex hull test is used to identify bad matches. Input to the test is the match map shown in Figure 3.3 (b). After just one iteration a small number of bad matches have been removed (a). After five iterations (b), more bad matches have been removed, but many good matches have been eroded from the outer edges.

This algorithm may be less promising than it seems at first sight. In Figure 3.7 (a) the occasional lone bad match has been removed, but many still remain. The reason for this is that badly matched neighbours support each other. Lone bad matches occur fairly infrequently. Where a bad match exists there is most probably a number of surrounding bad matches also. These bad matches can point in any direction and be of any length allowed by the original search window. The average effect then is that, in regions of bad mapping, the convex hull of any particular set of neighbours is quite large, and this therefore causes almost any bad match to be shielded from detection. Even just two badly matched vectors can support each other. This is shown in areas of Figure 3.7 (a). Iterating the convex hull bad vector removal strategy eventually erodes away areas of bad matches, but an unfortunate side effect is that good matches from external edges are also eroded. This is evident in Figure 3.7 (b).

As an alternative to simply removing such detected 'bad' matches, they could be re-mapped to the region of greatest similarity that does lie within the convex hull. Initial tests of this strategy have shown promising results. Typically, after iteration, most areas of bad matches are 'picked' away.

3.3 Summary and Conclusions

By spreading out the distribution of a region match mask, it has been shown that considerable run-time improvements can be made at little cost to the number of bad matches.

Considerable improvement of image region matching has been demonstrated by mapping to sub-pixel levels. In the tests presented, each target pixel has been divided into sixteen sub-pixels, whose values have been linearly interpolated from the whole-pixel level. Such improvement is at the expense of runtime, but initial experiments have shown that sub-dividing each pixel into just four sub-pixels provides a reasonable time trade-off.

A heuristic method for detecting and re-mapping bad matches has been introduced. This method is based upon the assumption that the relative ordering of local image features does not change between images. While promising in theory, this strategy fails to account for mutually supportive neighbouring bad matches. The solution may be to re-map each identified bad match so that it lies within the convex hull of the target points of neighbouring vectors—iterating until close enough to the final solution.

It seems clear at this point that a more effective strategy needs to be developed to overcome bad matches. Using real data, the number of bad matches can be as low as 0.49% (Table 3.2), but even this small number are still significant enough to affect analysis. However, removal or constraining bad matches may not be the total solution. Bad matches are usually due to two main reasons: (1) local image structure has changed radically between images, and (2) local image structure is too faint to facilitate good matching (resulting in bad matches as side effect of image noise). Both these cases can be seen in the images of Plates 1 and 2. If a false match is due to reason one, then this is precisely one of the changes being looking for (the other being movement), and the fact that the match vectors are randomly strewn around such an area can act as an identifying feature. If the bad match is due to reason two, however, then the bad match is due solely to image noise, and is therefore of less significance than other bad matches. Different strategies for dealing with these two types of bad match need to be developed. Woodward [10] used the local standard deviation as a measure local feature importance. If this figure is high, then something interesting is going on in that region, and so any bad matches here are probably due to reason one. If on the other hand, the local standard deviation is low, then the bad match is probably due to reason two.

Chapter 4

Analysis of Image Region Match Map

The preceding chapters have introduced new techniques and improvements for local image region matching applications. These techniques have been presented under a framework problem for detecting local brain structure changes by analysing MR imagery taken at different times. This chapter deals with the removal of global image mis-registration, and presents some considerations for analysis of the residual local match maps.

4.1 Removal of Global Image Transformation

Woodward [10] described a method for the calculation of global image transformation based on a least squares fit. This method has been implemented and used to estimate and remove such transformations from the image match maps presented in earlier chapters. Since the majority of matches arise from global image mis-registration, and not from local structure changes, then these form the bulk of the data that transformation estimation will work with. Removal of such transformation therefore leaves the vector map with (mainly) only localised features evident.

The global transformation is estimated as follows ([10], p.49):

If an image I_1 undergoes a translation (t_x, t_y), rotation θ, defined to be +ve anticlockwise, and magnification λ to become $\overset{I_2}{I - 2}$ then points $P_i, (x_{1i}, y_{1i})$ in I_1 will be expected to map to the points $\hat{E}_i, (x_{ei}, y_{ei})$ in I_2 according to the following equation:

$$\begin{pmatrix} x_{ei} \\ y_{ei} \end{pmatrix} = \lambda \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x_{1i} \\ y_{1i} \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix} \tag{4.1}$$

The centres of rotation and magnification can be incorporated into the translation, so that the rotation and magnification are treated as being centred on $(0, 0)$.

The transformation parameters between I_1 and I_2 are unknown, but the output from the regularisation is a list of points $M_i, (x_{2i}, y_{2i})$ in I_2, which match the points $P_i, (x_{1i}, y_{1i})$ in I_1. If more than two matching pairs are known the system is overdetermined for the unknown parameters t_x, t_y, θ and λ. These parameters will not usually have a solution because the matching points are not exact and will be perturbed by noise and changes. The values of the parameters are then determined by minimising the sum of the squared distances between the actual match points M_i and the expected match points $\hat{E}_i$. Defining:

$$m_x = \lambda \cos\theta$$
$$m_y = \lambda \sin\theta$$

Equation (4.1) can be written as

$$x_{ei} = m_x x_{1i} - m_y y_{1i} + t_x$$
$$y_{ei} = m_x y_{1i} - m_y x_{1i} + t_y$$

and in vector notation:

$$\hat{E} = Pt$$

where:

$$\hat{E} = \begin{pmatrix} x_{e1} \\ y_{e1} \\ \vdots \\ x_{en} \\ y_{en} \end{pmatrix}, t = \begin{pmatrix} m_x \\ m_y \\ t_x \\ t_y \end{pmatrix}, P = \begin{pmatrix} x_{11} & -y_{11} & 1 & 0 \\ y_{11} & x_{11} & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ x_{1n} & -y_{1n} & 1 & 0 \\ y_{1n} & x_{1n} & 0 & 1 \end{pmatrix}$$

x_{ei} and y_{ei} are linear functions or m_x, m_y, t_x and t_y. The solution to this linear least squares problem is:

$$t = (P^T P)^{-1} P^T M$$

In practice, it was found that at least three iterations of an estimate-remove transformation cycle was required to remove global transformations to within half a pixel.

4.2 Residual Vectors

Once global image transformations have been removed from the vector mapping, any correctly mapped vectors that do not map to their own start points do in fact point in the directions of detected change. The length of the residual vector gives the dimension of movement for that pixel-sized brain image structure. Figure 4.1 shows the residual vector map after global image transformation has been estimated and removed from the vector map given in Figure 3.7 (~~b~~a).

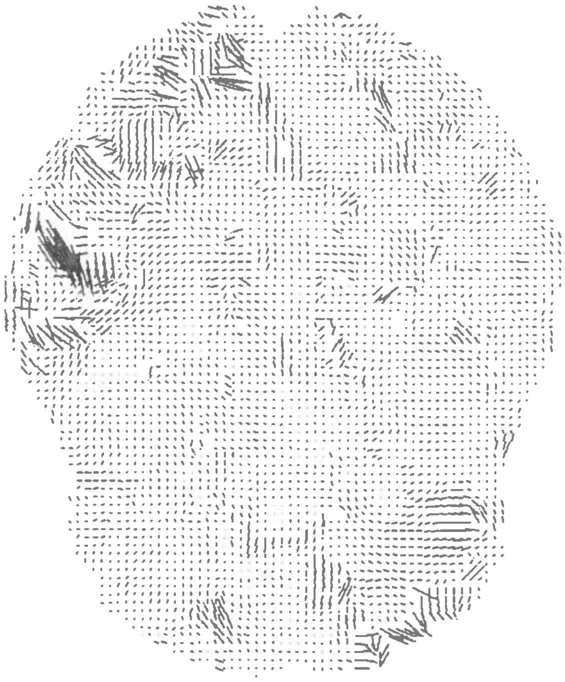
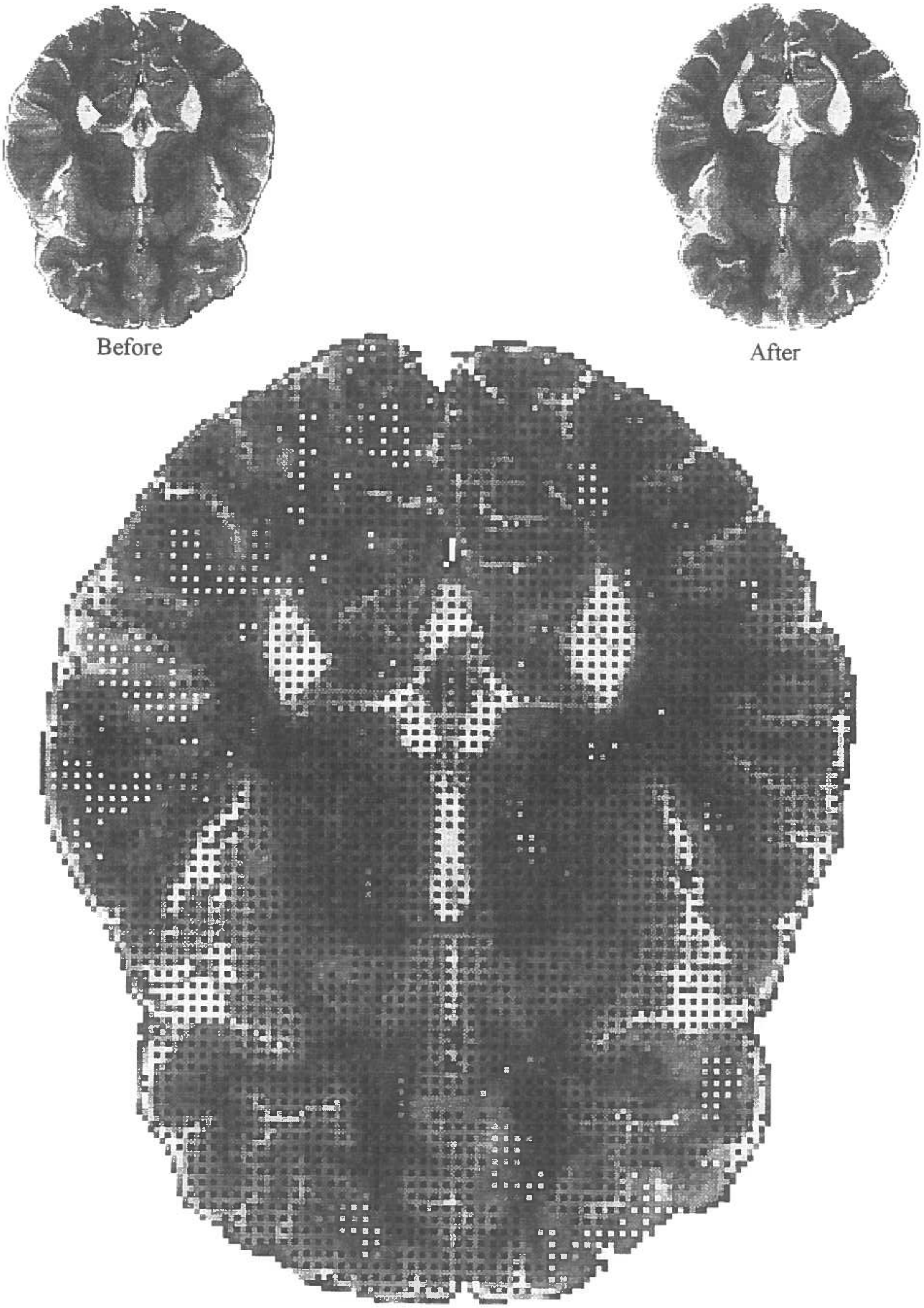


Figure 4.1: The residual vector map, showing likely local structure changes between images, computed as follows: A bi-variate cross-correlation local image region matching function is used on the images shown in Figure 2.1 to compute an initial image match map (Figure 3.3 (b)). This initial map is computed to sub-pixel level at a scale of four. Lone bad vectors are identified and removed (Figure 3.7 (a)). Finally global image transformation is estimated and removed (above). Plate 1 shows this image in full colour.

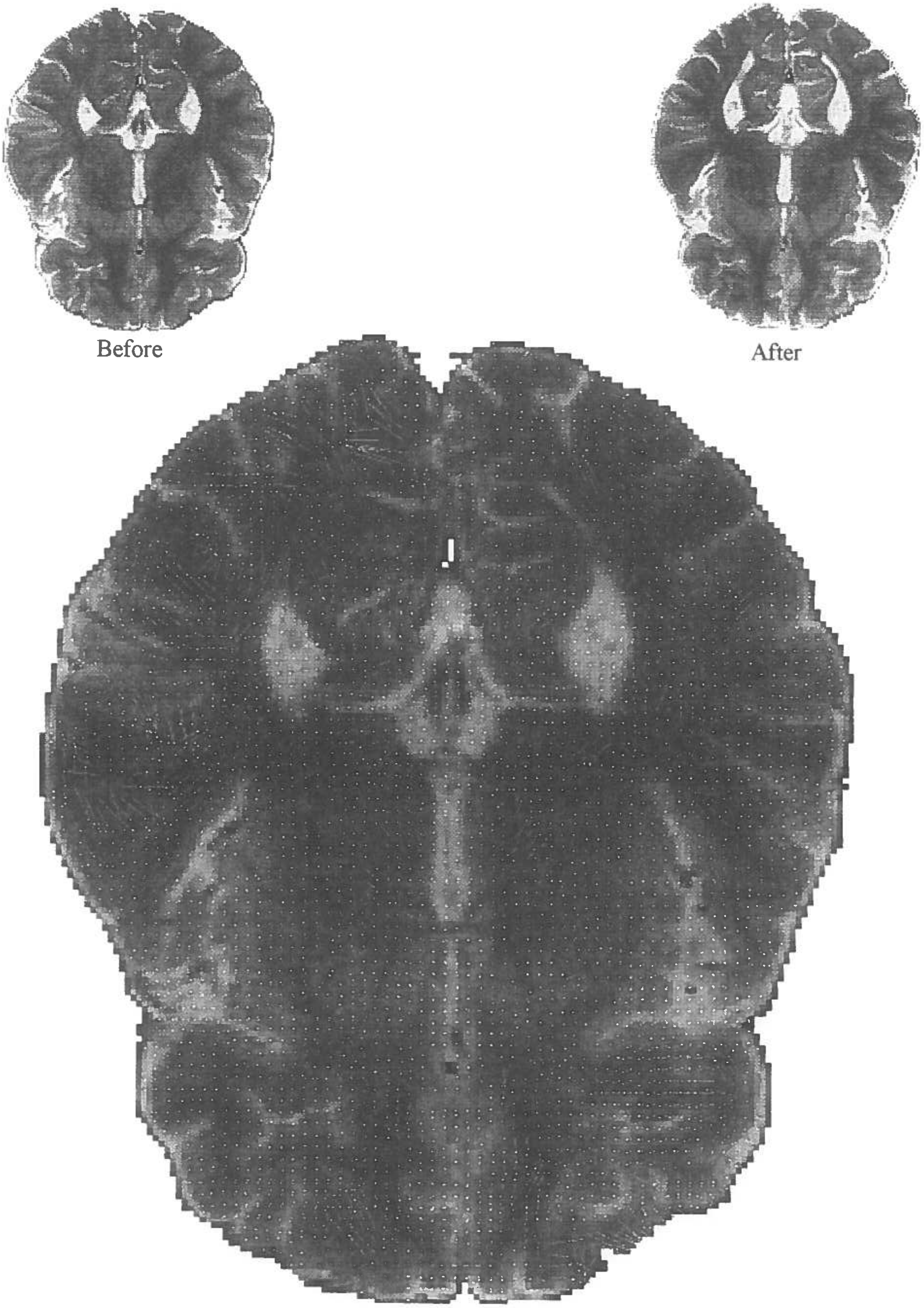
Plate 1 shows the final residual vector map in full colour. From this, areas of change between the source and target can easily be identified. Plate 2 shows the length of each residual vector, as a colour-coded overlay. Darker blue areas show residual vectors of short length, and hence areas of little change. Brighter red/orange areas showing increasingly longer vector lengths, and hence larger scale change. The brightly lit areas about the top and top left of this image show change detection of a fairly self evident nature—easily detected by the human observer. The smaller brightly lit area about the 'bulge' at the bottom right of the image has shown up a more subtle change (the local area has grown in size). This is less easily detectable by the human observer.



Before

After

Plate 2: Visual analysis is aided by replacing the vectors shown in Plate 1 with a colour-coded overlay. Blue shows little to no local structure change, whereas increasingly bright red shows increasingly large changes. Overlay colours are calculated directly from vector lengths, with short vectors contributing to blue areas, and long vectors contributing to red areas.



Before

After

Plate 1: Mappings are first computed for every second pixel in the horizontal and vertical directions. Linear interpolation of image data allows mapping to sub-pixel levels. A least squares estimate has been used to remove effects of global image transformations. Vectors indicating local structure changes are shown in red. Vector start points are shown in grey, while vector target points are shown in white.

University of Edinburgh

Department of Computer Science

Extensions to a Change Detection in MRI Brain Scan Data Program

4th Year Project Report

Peter I Oliver

31st May 1995

Addendum

Figure 3.5 on page 52 is incorrect. It should read as below:

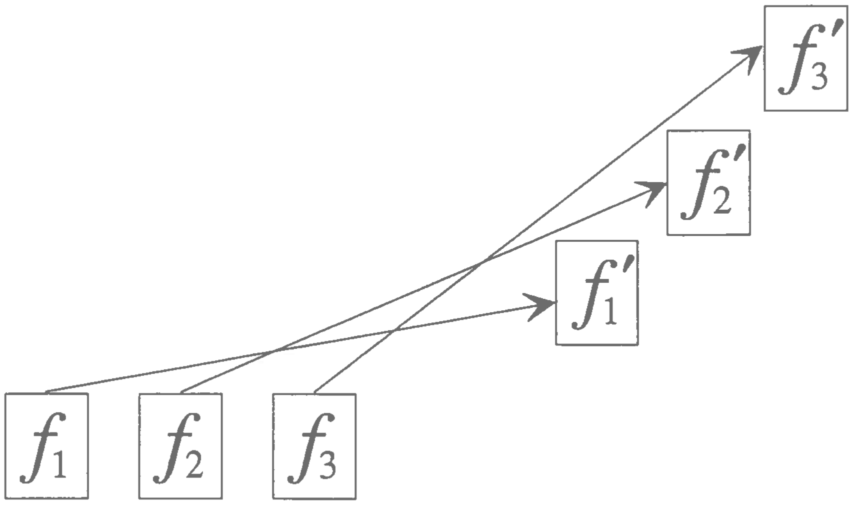


Figure 3.5: Detection of bad matches by crossed vector paths often fails under normal circumstances.

Chapter 5

Conclusions and Further Work

Woodward concluded his dissertation [10] with the remark:

> If this technique is to be investigated further the main areas of effort should be on reducing the speed of operation of the technique and on eliminating the groups of bad matches from the initial mapping.

In this work, I have presented a number of techniques generally applicable to image region matching and analysis.

With the availability of higher powered computers, plentiful memory, and better quality image data, image matching by pixelwise techniques are once again being researched. The main work done for this dissertation include the development of a new multi-variate image region matching strategy based on the normalised cross-correlation function. This function is especially suited to local image region matching as it compensates (on a local level) for differences of pixel value intensity distribution between multi-modal data. A number of other well known image region similarity metrics have been evaluated, and the main conclusion drawn here is that a simple multi-variate extension of the standard Euclidean metric performs almost as well as cross-correlation on real data, but with some five times greater speed.

Also presented is the extension of such multi-variate matching to sub-pixel matching to arbitrary levels. An image is subdivided into sub-pixels whose values are linearly interpolated from the original image. Such sub-pixel matching provides for initial region matching to a much greater accuracy than at the whole pixel level.

Finally, a method for identifying possibly-bad matches, based on the convex hull of the target points of neighbouring matches, is developed. This strategy does not work as well as hoped when faced with large groups of bad mappings. However, argument is given that such large groups do not necessarily need to be re-mapped or excluded from analysis since they most likely signify large scale changes that are so severe as to completely disrupt the local structure in that area. The exception here is where such bad matches are due to a failing of the local similarity metric in the face of noisy data (usually in areas of low gradient intensities). There is a need for further work to be done in identifying such bad matches, and their founding reason. A suggestion is made that the local standard deviation of pixel values can point to the cause of bad matches in any one area. If the standard deviation is low, the bad matches are probably due to noise and low pixel value activity in that area. If, on the other hand, the standard deviation is high, then, in the light of this, such a failing to match correctly is most probably due to there being radical differences between the structures in that part of the images. So much so that the similarity metric does it's job in reporting no cohesive similarity.

The available MRI data is organised into serial two-dimensional sections. Stacking such sections will facilitate the development of a three-dimensional change detection system. This project has made some advances aimed at paving the way to such a development. The image matching strategy developed generalises well to three-dimensional data. Computational complexity (time) still being the only practical problem. It is not necessarily a requirement that a match is made for every pixel in the source image. Further work may develop a matching strategy that aims to match only pertinent areas (if any are identifiable).

Little assessment of the effects of differently shaped image match masks has been presented here. It is a suggestion for further work that such assessment is made.

In final conclusion, the work presented here has advanced the power and flexibility of local image region matching techniques. Application has been made to the detection of brain structure over time without explicit reference to higher order structures other than local image similarity. Simple pixel area matching has been improved by moving to the sub-pixel domain. Linear interpolation has been chosen for ease of analysis and implementation. It is suggested that further matching strategy improvements can be made with more advanced interpolation techniques.

Bibliography

[1] D.H. Ballard and C.M. Brown. *Computer Vision.* Prentice-Hall, New Jersey, 1982.

[2] T.H. Cormen, C.E. Leiserson, and R.L. Rivest. *Introduction to Algorithms.* The MIT electrical engineering and computer science series. McGraw-Hill, first edition, 1989.

[3] R.B. Fisher and P.I. Oliver. Multi-variate cross-correlation and image matching. Submitted for publication to British Machine Vision Conference 95, 1995.

[4] J.D. Foley, A. van Dam, S.K.Feiner, and J.F. Hughes. *Computer Graphics: Principles and Practice.* Addison-Wesley Systems Programming Series. Addison-Wesley Publishing Company, Inc., second edition, 1990.

[5] S. Geiss, J. Einax, and K. Danzer. A two-stage cross correlation approach to template matching. *Analytica Chemica Acta*, 242:5–9, 1991.

[6] A. Goshtasby, S.H. Gage, and J.F. Bartholic. A two-stage cross correlation approach to template matching. *IEEE Trans PAMI*, 6:374–378, 1984.

[7] X. Li, C. Shanmugamani, T. Wu, and R. Madhavan. Correlation measures for corner detection. *Conf. on Computer Vision and Pattern Recognition*, pages 643–646, 1986.

[8] C.A. Pelizzari, G.T.Y Chen, D.R. Spelbring, R.R. Weichselbaum, and C-T. Chen. Accurate three-dimensional registration of CT, PET and/or MR images of the brain. *Journal of Computer Assisted Tomography*, 13(1):20–26, January/February 1989.

[9] H.S. Ranganath and S.G. Shiva. Correlation of adjacent pixels for multiple image registration. *IEEE Trans Comp.*, 34:674–677, 1985.

[10] J.O.N. Woodward. Change detection in MRI brain scan data. MSc Dissertation, Dept. of Artificial Intelligence, Univ. of Edinburgh, 1990.

Appendix A

On-line information

This project has been approached very much in the spirit of research and evaluation of the the developed techniques. A large number of 'C' programs have been developed (and a large number discarded). As of presentation of this dissertation, all relevant details appertaining to the developed system of programs, data structures, etc. (in fact everthing needed to continue this work further) can be found in the subdirectories of my DAI and CS accounts:

DAI: `/petero/PROJECT/` (See the file **`README`**).

DCS: `/pio/PROJECT/` (See the file **`README`**).