Registration of Images with Dissimilar Contrast using a Hybrid Method Employing Correlation and Mutual Information

Karolyn A. Abram
Department of Computer Science, Dartmouth College, Hanover, NH 03755
Dartmouth TR2000-369
June 2, 2000

ABSTRACT

We present a multi-scale hybrid approach for aligning multi-modal images employing correlation and mutual information. We attempt to limit the number of mutual information calculations required in finding the alignment of a test and reference image independent of their contrast. The method is not feature-based and does not require the input images to be pre-processed (i.e., landmarks). We present results from a test involving the registration of T1 and T2 test images.

1 INTRODUCTION

1.1 Overview

The problem of fitting one image into another is commonly referred to as "registration." Finding the best possible translation and rotation necessary to align two images is one approach to solving this problem. Registration is a crucial component of many remote sensing and medical image interpretation applications. Image alignment techniques aid in volumetric estimations of complicated structures and allow radiologists to accurately identify changes between sequential images. For example, radiologists require image alignment capabilities when they must compare images of a patient over time to detect changes. In the case of mammograms, tissue growth can potentially displace and/or distort regions of cells within an image. Other factors such as patient motion or the inability of technicians to place patients in the same anatomical position during imaging also show the need for registration.

Numerous image registration techniques exist for correcting the alignment problems mentioned above; such as, Feature-based/Landmark, Local Error, Correlation, and Mutual Information. Feature-based approaches are limiting because they depend on landmarks and pre-placed reference points within images. Images without the reference points cannot be aligned. Both Local Error and Correlation find possible alignments by minimizing the overall difference between corresponding pixel intensities within the images. These error minimization approaches have shown positive results; however, they fail to align images of dissimilar contrast. Unfortunately, the need to align images differing in contrast is common since the imaging technique, compression, device calibration, or image quality all affect contrast. For example, when comparing T1 and T2 class images, error techniques may fail since these image types vary in their relative contrast to white matter. [1,2] have shown that multi-modal images can be aligned successfully by maximizing the mutual information between the images. Unfortunately, determining shifts and rotations through a pure mutual information approach is complicated and computationally intensive. In an attempt to exploit the qualities of two techniques, we consider a hybrid method relying on correlation and mutual information. We have two main goals: to create a more contrast independent method than correlation alone, and to achieve an algorithm whose running time is lower than that of an exhaustive Mutual Information approach.

1.2 Outline

Contents

2 MULTI-SCALE REGISTRATION METHOD

In this section, we provide the overall framework for our multi-scale approach to register two images. A multi-scale approach means that multiple iterations will be made over the two images that are being aligned. For each iteration, sub-blocks of the test and reference images will be aligned. The rotation and translation parameters for each block will be found using our hybrid method. The mathematical concepts and individual alignment algorithms that are mentioned below will be defined in subsequent sections.

The overall goal of image registration is to compute a molded version of the test image that is the best-aligned match to the reference image. Our notion of the "best-alignment" is in terms of finding the overall minimal distance between each pixel of the two images.

By terming this approach "multi-scale," we mean that numerous passes are made over the two images, which involve sub-blocks of the images. The term "rigid registration" applies to a single pass approach with only two blocks that are the full size of the test and reference images.

At each step, the optimal rotation and translation is found for the blocks in the test image that are being aligned with blocks in the reference image. The block sizes propagate from coarse to fine scales at each of the iterations by a factor of $1/2$.

The parameters determined for each block of the test image are the translation by (x_0, y_0) and the rotation θ_0 to best align it with the reference block. Our hybrid alignment method, which will be described in Section 3, is used to find these parameters between corresponding blocks. The rotation and translation parameters are then used to determine a smooth molding map for the entire test image. The test image is molded using this map to form the new test image for the next iteration. In the next iteration, the images are subdivided again into blocks that are $1/2$ the size of the previous blocks and the process is repeated. The total number of blocks used at each stage more than doubles because we overlap the block regions. The algorithm continues until a predetermined minimum block size is reached. Our code uses template files to obtain the block size and the location of its center. We do not strictly adhere to halving the block size each time so that we can add some overlapping blocks. Figure 1 on the next page shows the breakdown of each block for our template files. The crosses indicate the location of the center of a block.

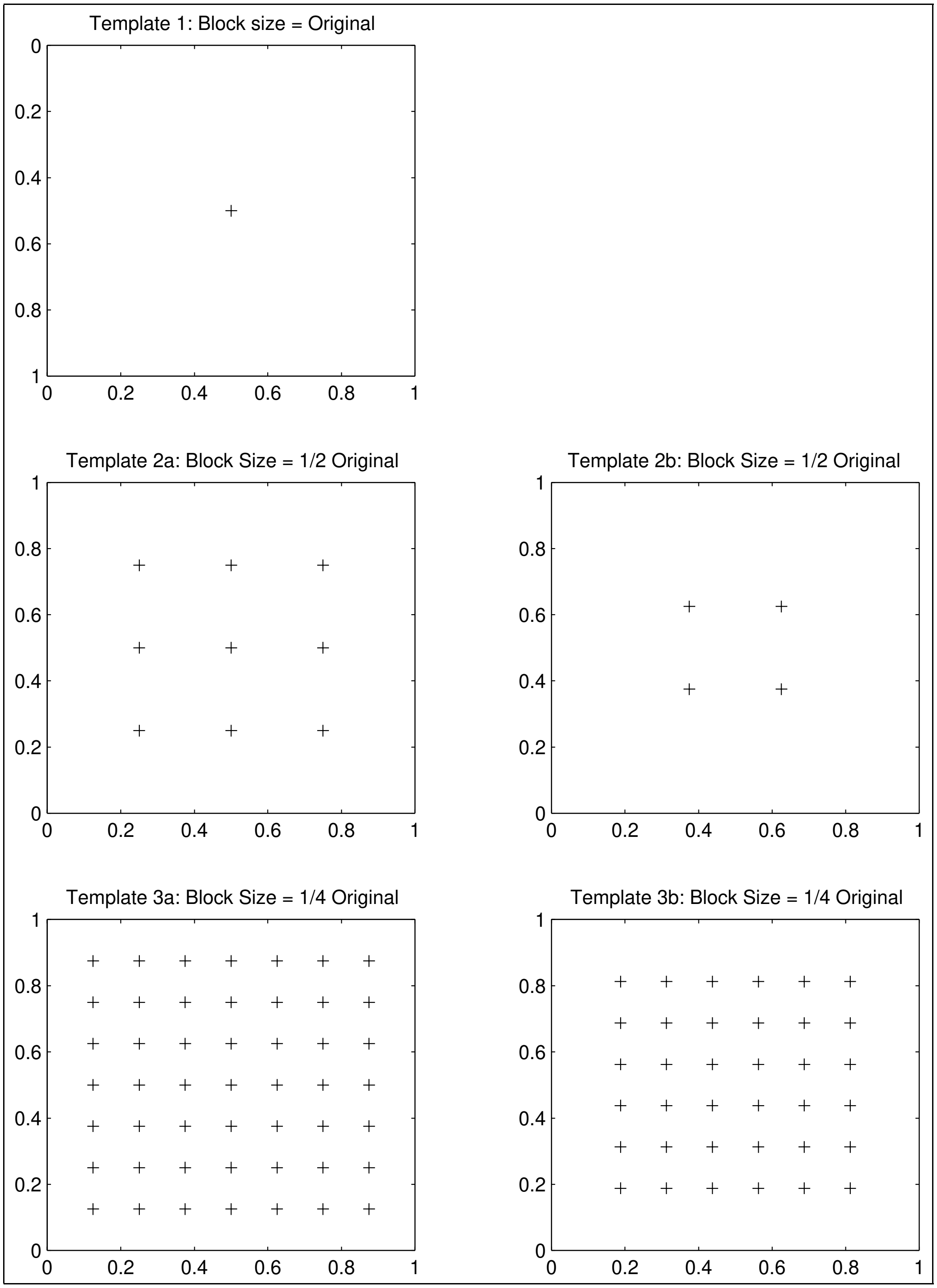


Figure 1. Template files for each stage in the multi-scale alignment

During the very first iteration, a rigid registration is performed to match the boundaries of the test and reference images. To perform the rigid alignment, we use the cross correlation to find the desired rotation and translation. We can use the cross correlation at this stage because it effectively over-emphasizes the edges of the two images during the rigid alignment. This step can be skipped if the input image boundaries already match.

The multi-scale alignment method is explained by following pseudocode:

(Note: For clarity, square images are assumed in the pseudo-code).

```
// Test[i] and Ref[i] refer to the blocks of the
// test and reference images
// NumBlocks and Steps defined in template file
// Initialize Block size to full image
Block size = Ref size

For n = 1 to Steps {
   For i = 1 to NumBlocks {
     If (n == 1)
         // Find block Rotation & Translation Correlation
         MoldMap[i]= Correlation(Test[i], Ref[i])
     Else
         // Find block Rotation & Translation via Hybrid
         MoldMap[i]= Hybrid(Test[i], Ref[i])
   }

   // Mold the test image by the computed MoldMap
   Test = Mold(Test, MoldMap)

   // Adjust Block Size
   Block size /= 2
}
```

The number of operations in this algorithm is dependent on the running time of the hybrid method, which will be mentioned when its implementation is explained. Mold is an $O(S)$ time operation, where S is the full size of the $N \times M$ image matrix. The correlation is an $O(S \log S)$ operation using the FFT [3]. The running time of the hybrid method is bounded from below by that of correlation. The running time of the multi-scale algorithm is $O(Steps \cdot NumBlocks \cdot H(S))$, where $H(S)$ is the upper bound of the running time for our hybrid method.

3 HYBRID METHOD TO DETERMINE ROTATION AND TRANSLATION

Our hybrid method combines techniques to devise a registration approach requiring less computation than exhaustive mutual information and which is more contrast independent than correlation. The next few sections will detail the way in which we incorporate both correlation and mutual information techniques to perform image registration.

3.1 Hybrid Algorithm

Our algorithm combines pieces of the rotation and translation determination methods that we discuss in detail in section 6. The steps in determining a rotation and translation for aligning a block of the test image with the reference image are as follows:

Note: NumRotations, NumPeaks, and range are user-defined variables.

3 Use the **cross correlation** to find rotation angle candidates via the method detailed in Section 6.1.1. In addition to the max peak's rotation, store the rotation angles, θ_n, of the other top NumRotations-1.

4 For each possible rotation, calculate the **phase only correlation** for the two images as detailed in section 6.1.2. In addition to the max peak's coordinates, store the (x_n, y_n) coordinates designated by the other top NumPeaks -1.

5 Calculate the **mutual information** for the NumPeaks translations of NumRotations rotations and a range around the translation peaks.

6 The coordinates of the max mutual information give the rotation/translation answer for aligning the test image with the reference image.

Figure 2 on the next page summarizes the hybrid alignment procedure.

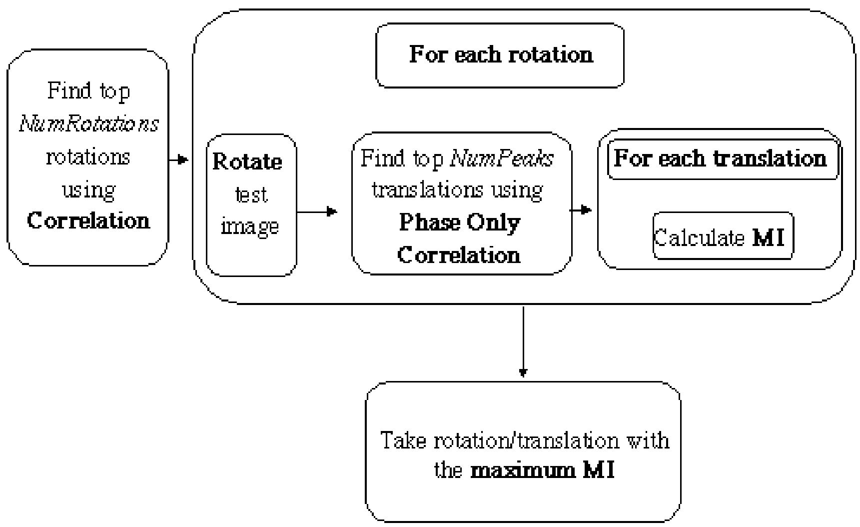


Figure 2. Hybrid method using correlation and mutual information

3.2 Implementation Overview

The implementation of the hybrid registration approach is primarily in Matlab including some core routines in C. A test and a reference image are taken as inputs to the program. The code currently is written to handle grayscale images of size 256 x 256 where the pixel values range between 1-255 (zero values are excluded to avoid divide by zero errors).

3.2.1 Block sizes

For each of the iterations, the block size and position is determined from a template file. The template files allow us to easily modify or add block locations for performing the multiscale registration. Currently, our block size varies from the full 256 pixels down to a minimum test window size of 32 pixels. Our use of the full or windowed test image depends on the iteration. When the correlation is used to find the rotation in the initial rigid alignment pass, equally sized reference and test images are used.

A gaussian filter is passed over the test image prior to calculating the phase only correlation to find the peaks for translation. This filter is used to emphasize the central elements of the test image and avoid allowing edges to drive the correlation.

For the MI calculations, the test blocks are set at some fraction of the reference block. We used test blocks that are 1/2 the size of the reference blocks. The smaller test block determines the amount each pixel can be translated in the test image to match the reference image. The square test block can move ± length/4 in the x and y directions to match the reference block (assuming square images). The diagram below explains the allowed translations:

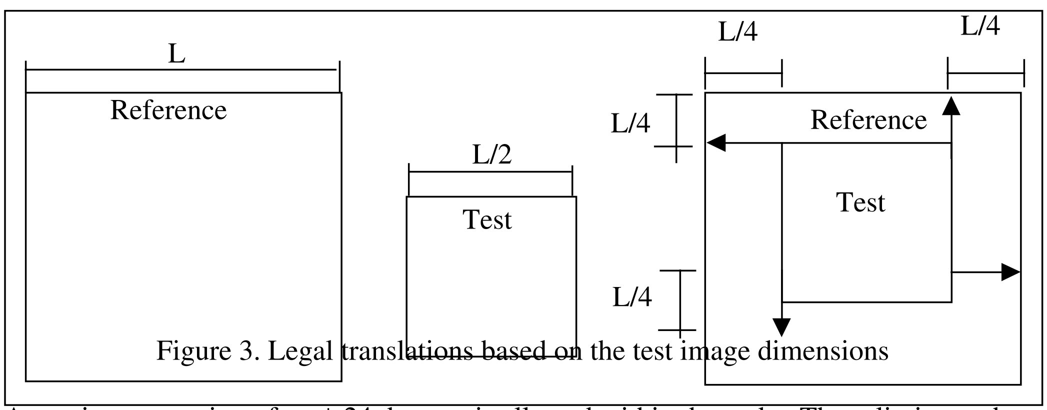


Figure 3. Legal translations based on the test image dimensions

A maximum rotation of up ± 24 degrees is allowed within the code. These limits on the translations and rotations are reasonable since the first stage of alignment (using block sizes of 256 x 256) rigidly align the borders of the images and we do not expect to need alignments outside these boundaries.

3.2.2 Flow Control

The algorithm is implemented identically to how it is described in section 3.2.1. The images are stored in a matrix in Matlab and manipulated throughout various Matlab scripts. The use of the DFT to calculate the correlation offers enhanced performance as opposed to an iterative summation. The flow control is primarily dictated by a large set of nested for loops. The outermost loop goes through each of the possible NumRotations

rotations. The inner three for loops control going through each of the translation peaks found from the phase only correlation. For each of these translations, we calculate the mutual information. In addition to the translation identified by the peak, we calculate the MI for a range of other translations centered about the peak coordinates. Prior to calculating the MI, the code checks that the translation is within the allowed L/4 and that we have not already calculated this MI due to a nearby peak. The block of pseudocode on the next page briefly details this process.

```
// Find NumRotations using cross correlation method in
Section 4.2.1
GetRotations(Test, Ref, θ)

// Go through each rotation and find translations
For i = 1 to NumRotations   {
   // Rotate the test image by θ[i]
   Test = Rotate(Test, θ[i])

   // Calculate Phase Only Corr. of Test and Ref images
  Pcorr = POC(Test, Ref)

   // Find absolute peaks from the Phase Only Corr.
   Peaks = GetPeaks( ABS(Pcorr) )

   // Get (dx,dy) translations from the top NumPeaks
   GetTrans(Peaks, dx, dy, NumPeaks)

   // Find MI for a range around each translation peak
   For j = 1 to NumPeaks {
     For cols = (dx[j] - range) to (dx[j] + range) {
       For rows = (dy[j] -range) to (dy[j] + range) {
          // Translate Test by (cols, rows)
          TestShift = Translate(Test, cols, rows)

          // Calculate MI for TestShift and Reference
          MI_ans[i,cols,rows] = MI(TestShift, Reference)
       }
     }
   }
}

// Answer for optimal rot and x,y shift is the max MI
Max(MI_ans, rot, dx, dy)
```

Obtaining the candidate rotations and the POC calculation both require $O(S\lg S)$ operations. As mentioned before, a rotation and translation of an image can be achieved in $O(S)$ time, where S is the full size of the $N \times M$ image matrix. The MI calculation is also an $O(S)$ time operation. Taking into account these running times, the overall requirement of the hybrid algorithm is $O(NumRotations \cdot S\lg S + NumPeaks \cdot Range^2 \cdot S)$. NumRotations, NumPeaks, and Range are all user-defined constants that are normally ≤ 12. So, we can say the running time of the hybrid method is $O(S\lg S)$. The running time of an exhaustive MI approach is $O(S^2)$.

3.2.3 Calculating the MI of the Test and Reference Images

For calculating the mutual information, some special care is taken to determine the mutual information of the smaller test window and the larger reference image. A window the same size as the test window is chosen from the reference image dependent on the shift for which we are calculating the MI. As shown in the diagram below, to simulate the effect of translating the test image, the shaded window of the reference image is used to calculate the MI with the test window for this (dx_0, dy_0) translation.

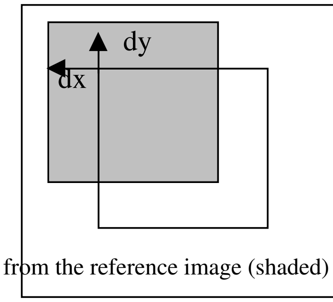


Figure 4. Obtaining a window from the reference image (shaded) that is the size of the test window

4 RESULTS

In this section we illustrate and analyze a full run of our multi-scale approach using the hybrid method for alignment. A separate diagram is shown for each of the template files (listed in Figure 1). The reference image is a T1 image and the test image is identical to the reference image except that it is a T2 image. To test our multi-scale hybrid method, we randomly warp the T2 test image as shown on the following page.

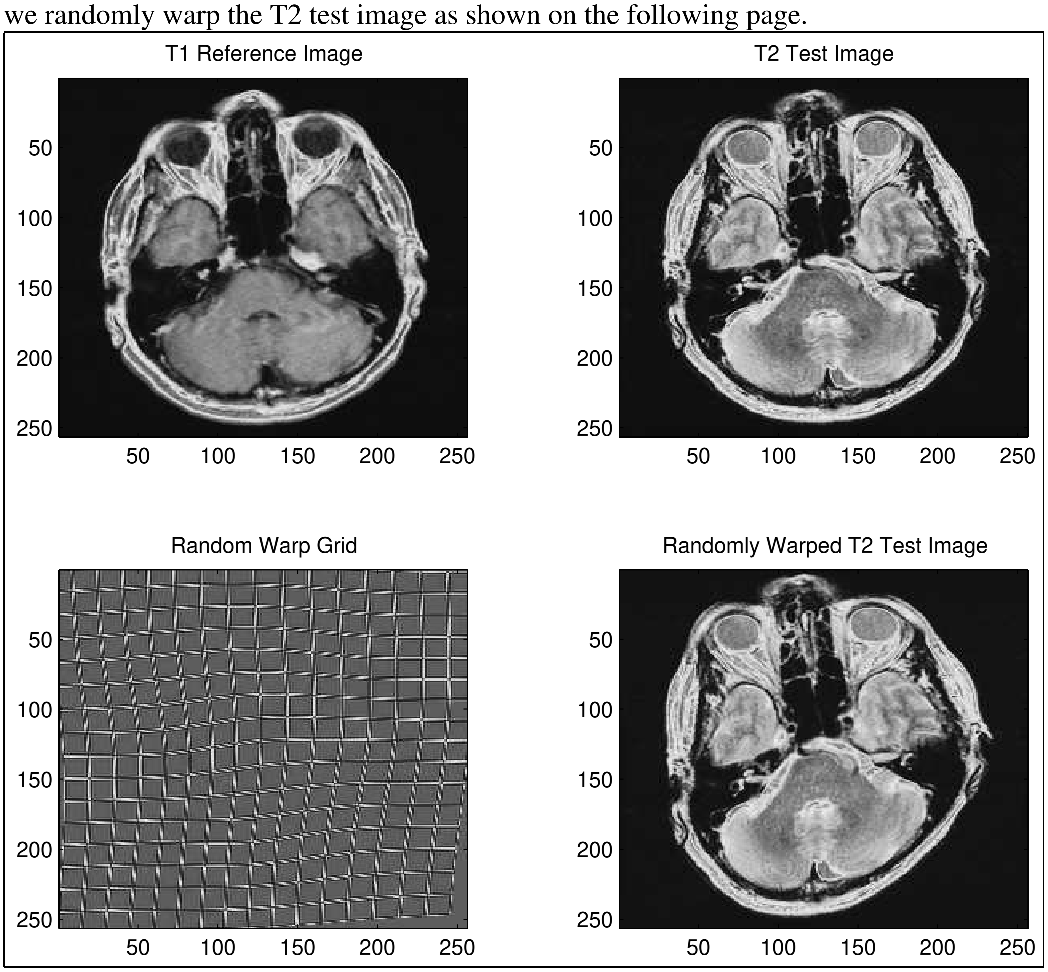
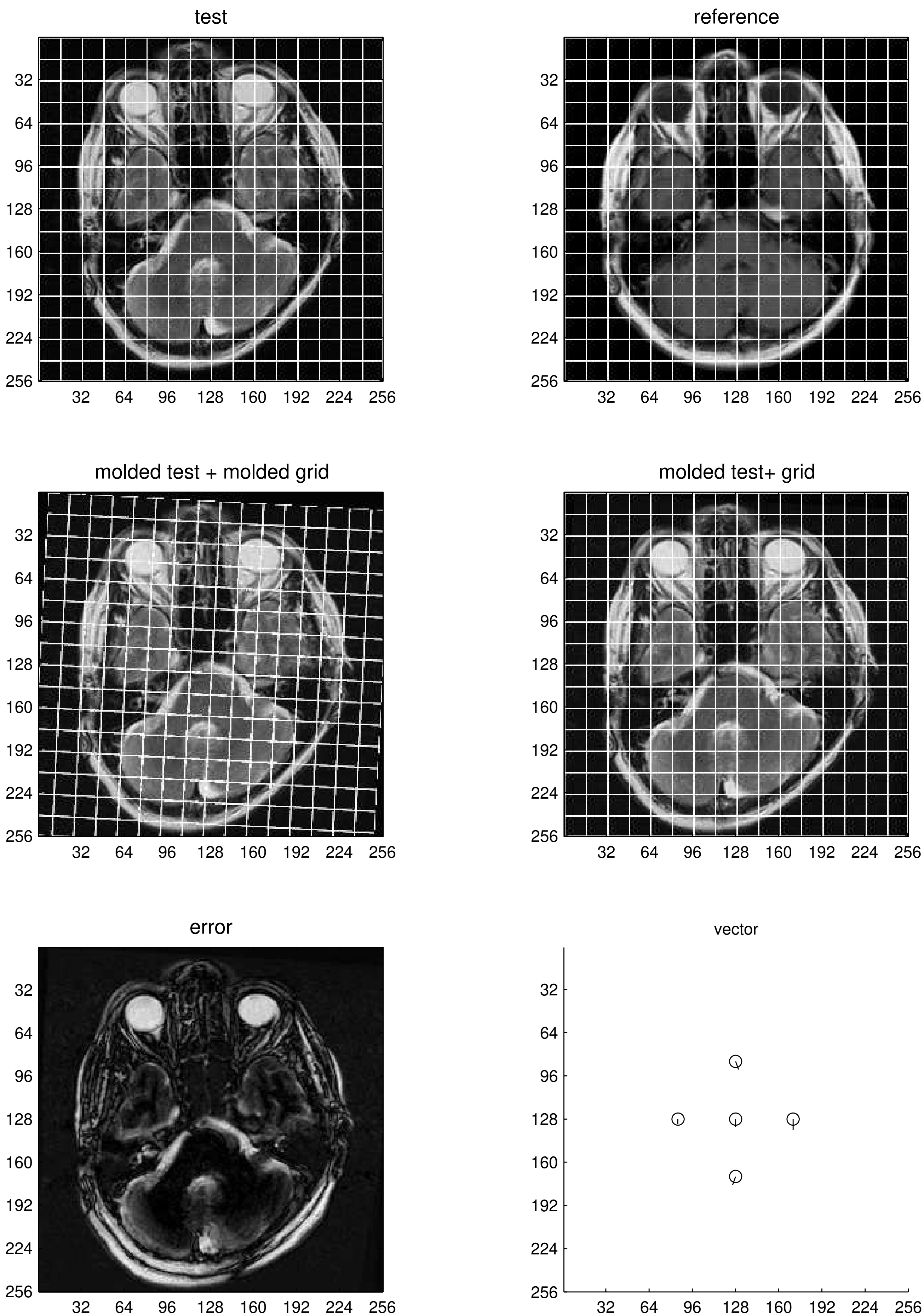


Figure 5. T1 Reference image and a randomly warped T2 test image

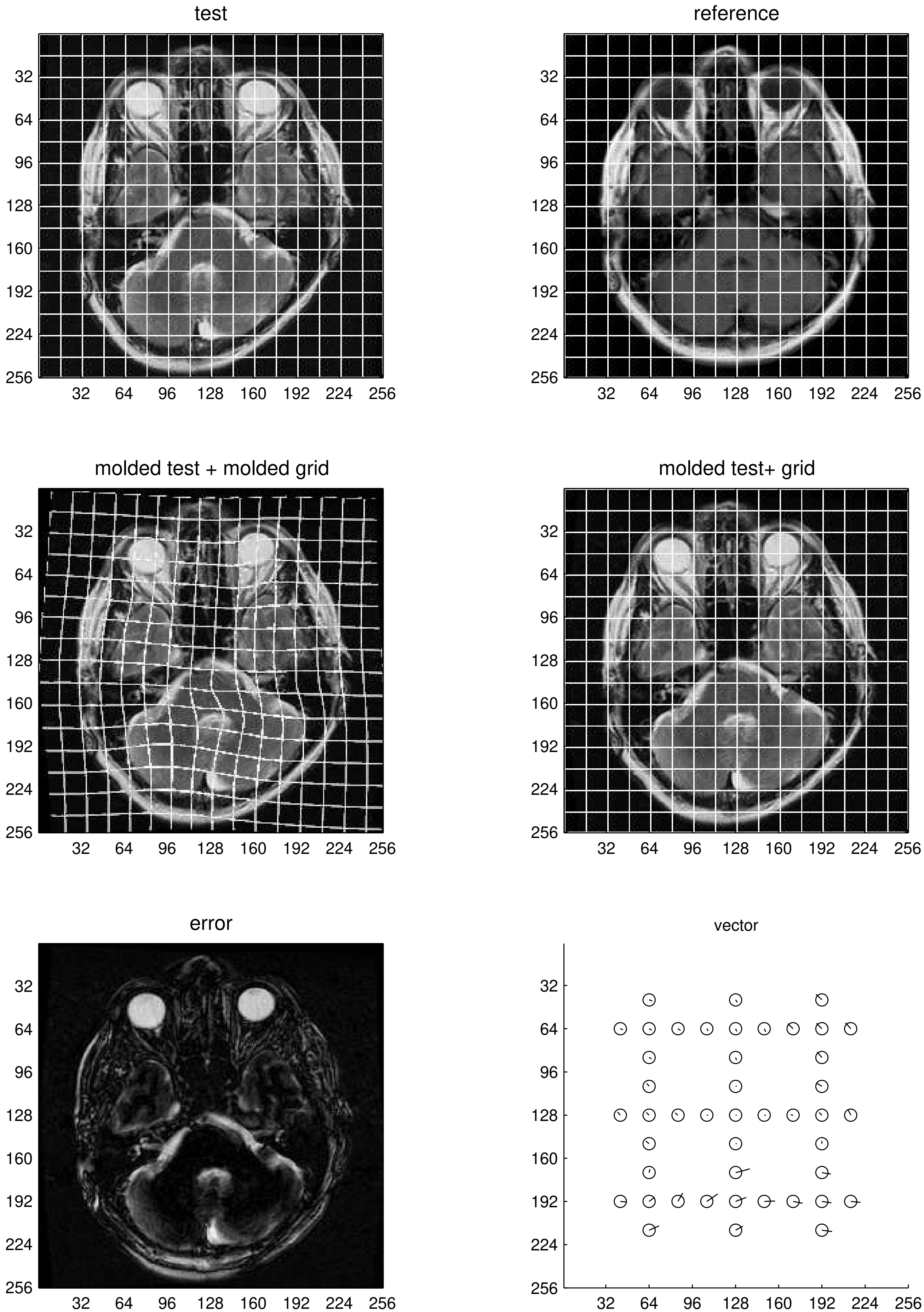
Now, with this randomly warped test image, we are able to run the alignment program for the T1 reference image and the warped T2 test image. At each step in the multi-scale algorithm, the test image should converge toward the reference image alignment.

The following diagrams illustrate each step of the multi-scale hybrid alignment method. In this run of the hybrid method, $NumRotations = 6$, $NumPeaks = 2$, and $Range = 1$. Each page represents one iteration of the multi-scale method. The image marked “reference” is the T1 image that we are trying to fit the image marked “test” into. The test image is warped based on its rotation and translation parameters for each block found with the hybrid method. The “molded test + molded grid” shows where the blocks of the test image where moved from the iteration. The “error” image is just the difference between the molded test and the reference image. The lower right-hand image titled “vector” displays the rotation and translation movement for each block of the template file. The movement of each block is indicated by the length and direction of the lines. The next few pages illustrate each of the five steps in the multi-scale alignment.

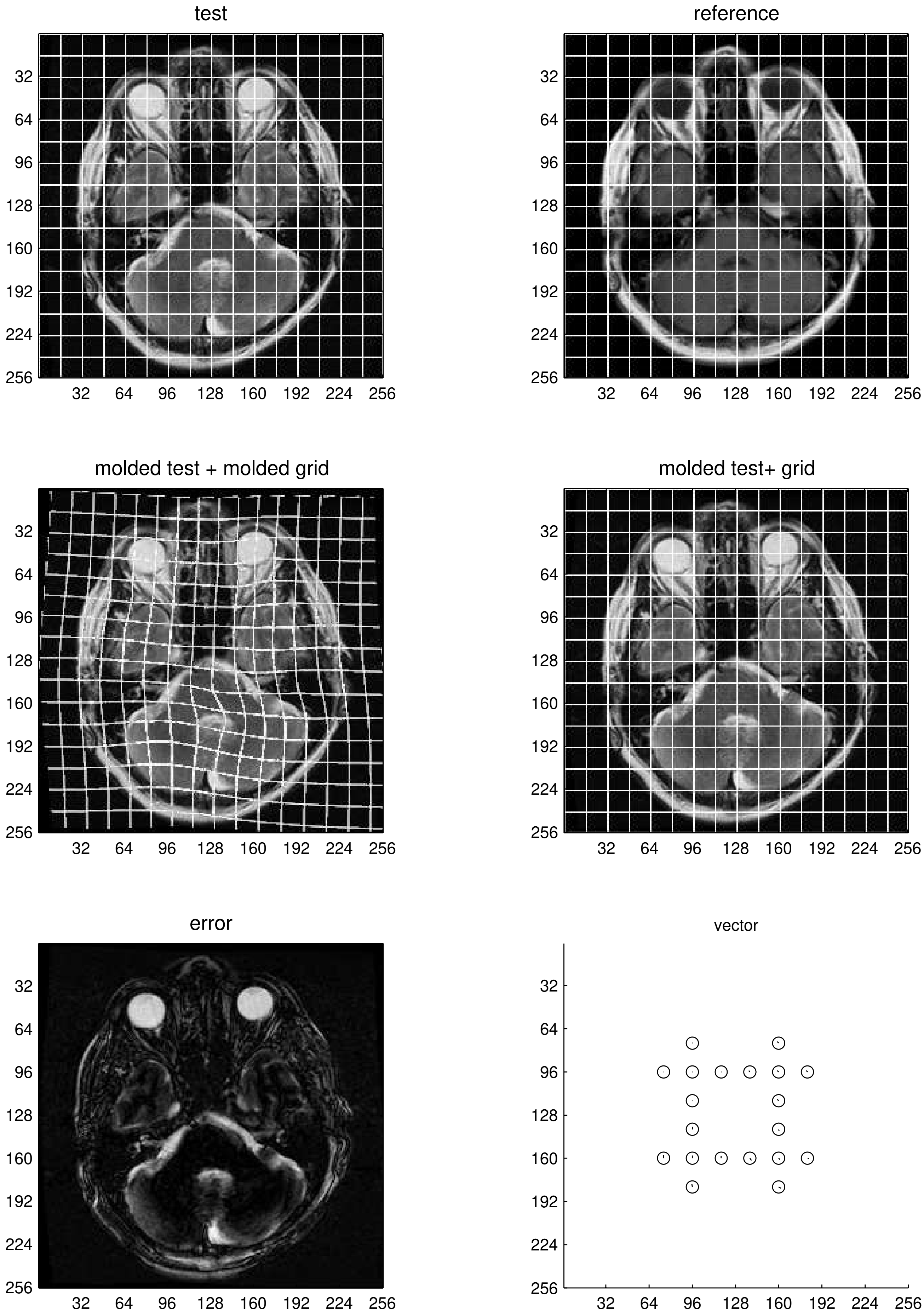
Hybrid (numpeaks:2 range: 1), Iteration: 1, block size: 256, using Template 1



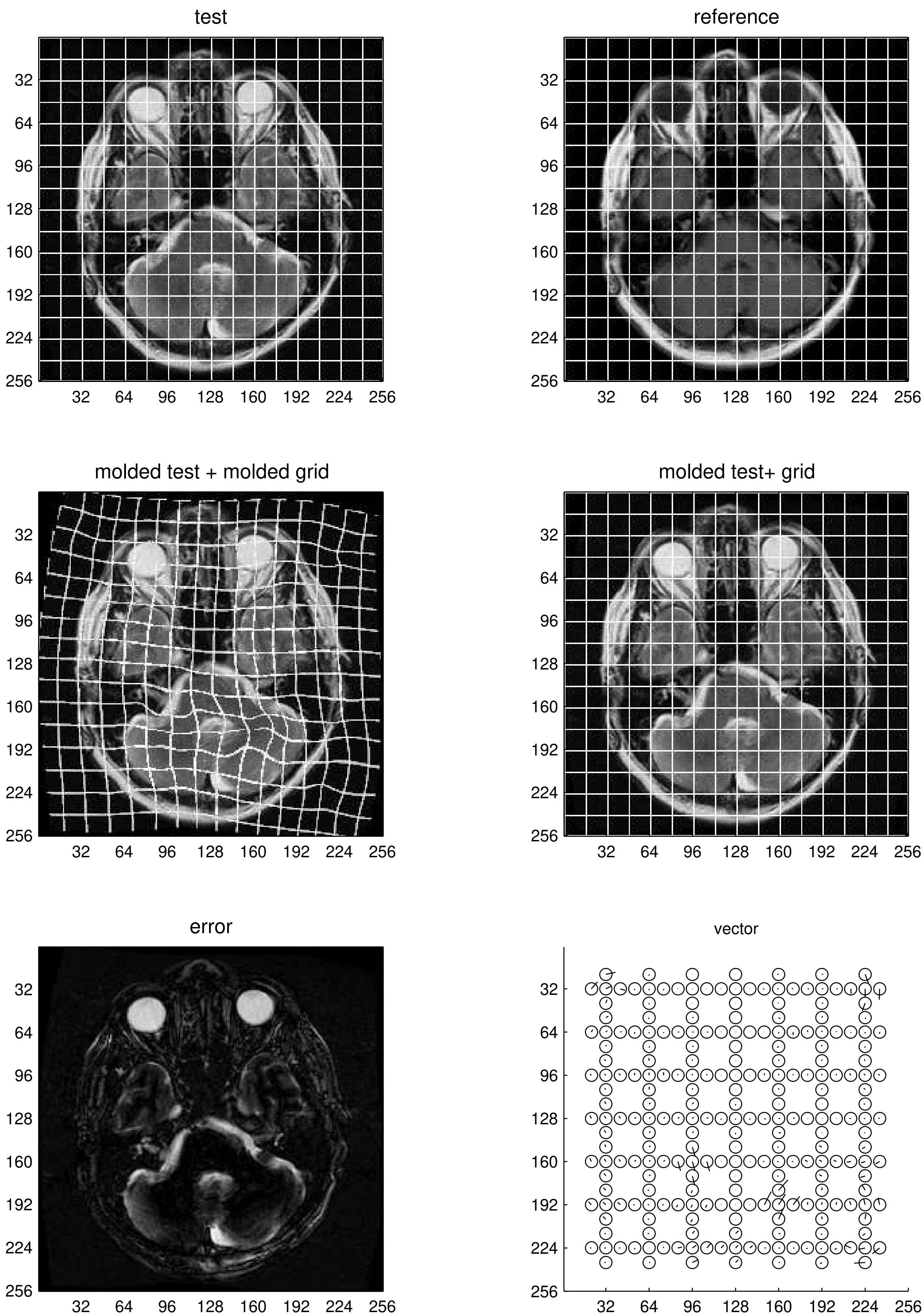
Hybrid (numpeaks:2 range: 1), Iteration: 2, block size: 128, using Template 2a



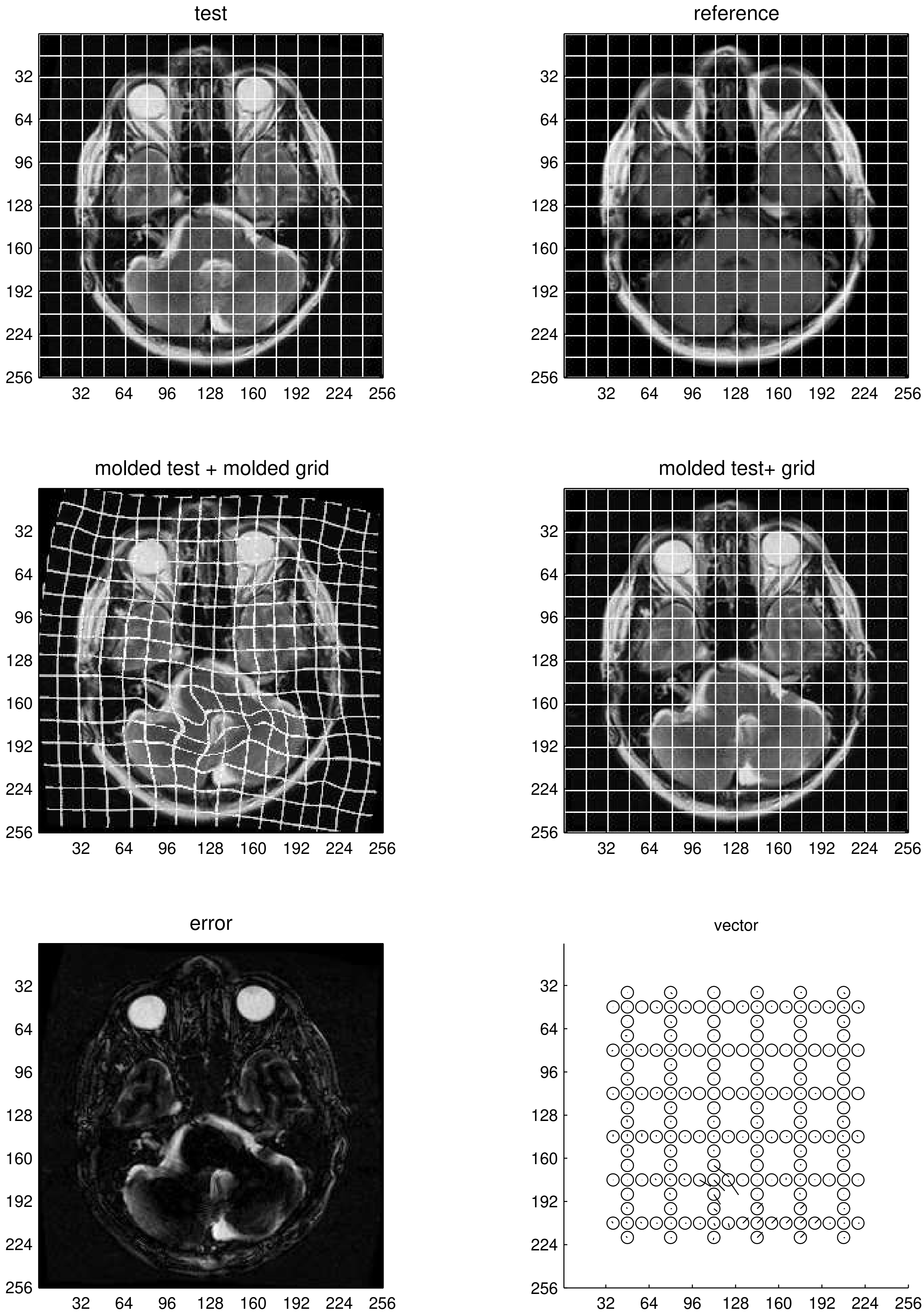
Hybrid (numpeaks:2 range: 1), Iteration: 3, block size: 128, using Template 2b



Hybrid (numpeaks:2 range: 1), Iteration: 4, block size: 64, using Template 3a



Hybrid (numpeaks:2 range: 1), Iteration: 5, block size: 64, using Template 3b



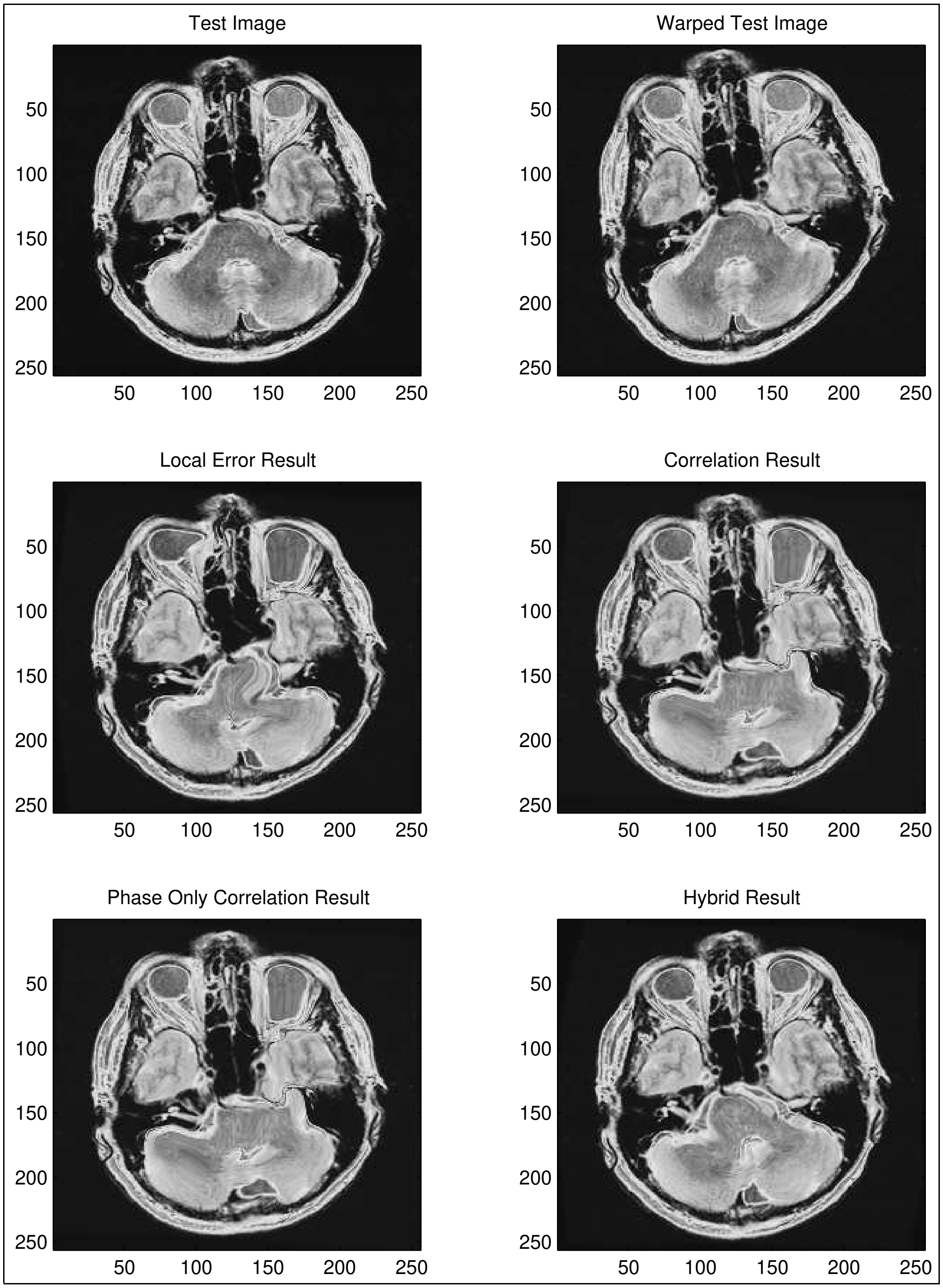


Figure 6. Original reference image and the molded result from each method

Figure 6 shows the final warped test image after the 5th iteration of the multi-scale alignment using the listed method. We can visually identify that the hybrid method performed the best alignment. Our original T2 image is the ideal result for aligning the warped test image with the reference image. Since the warped T2 image is of the same modality as its non-warped version, we can use the error between these two images to evaluate our algorithm. The Root Mean Squared Error (RMSE) and the Peak Signal to Noise Ratio (PSNR) are the error measures. We compare our results from the hybrid method at each stage with local error, correlation, and POC alignment methods. A lower RMSE between two images implies a better match and the inverse is true for the PSNR. All of the values are the same after the first iteration since the correlation is used to perform a rigid alignment of the two images.

Table 1. RMSE values between T2 warped test and original T2 image

	Alignment Method			
Iteration #	Local Error	Correlation	Phase Only Correlation	Hybrid
1	177.261	177.261	177.261	177.261
2	170.308	170.136	170.136	167.916
3	170.373	170.396	170.396	166.580
4	169.000	171.009	171.009	165.834
5	168.959	170.596	170.596	165.499

Table 2. PSNR values between T2 warped test and original T2 image

	Alignment Method			
Iteration #	Local Error	Correlation	Phase Only Correlation	Hybrid
1	3.15853	3.15853	3.15853	3.15853
2	3.50611	3.5149	3.5149	3.62897
3	3.50279	3.50159	3.50159	3.69834
4	3.57307	3.47044	3.47044	3.73731
5	3.57518	3.49144	3.49144	3.75492

The RMSE and PSNR values identify the hybrid method as having found the best possible alignment for our tested methods. We can see that the correlation and POC, which computed identical answers, begin to fail after the third iteration. The local error technique does fairly well, but the hybrid method seems to perform the best.

1. MATHEMATICAL BASIS

In this section, we define the necessary mathematical concepts that allow us to detect translations and rotations for aligning images. Properties of the Discrete Fourier Transform (DFT) and derivations from information theory mathematics are the building blocks of our alignment methods.

5.1 DFT Properties

The Fourier transform provides a change of basis from spatial domain to the frequency domain [3]. Properties of the DFT are used to find possible translations and rotations between the test and reference images. This section reviews these DFT concepts. All expressions are shown as summations instead of integrals since our applications are in the discrete case.

5.1.1 DFT Definitions

Let $f(x,y)$ be a two-dimensional function, such as a matrix of pixel values. The Discrete Fourier Transform $\Im\{.\}$ and its inverse $\Im^{-1}\{.\}$ are defined as follows

$$\Im\{f(x,y)\} = F(u,v) = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)e^{-j2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)} \tag{1}$$

where $j = \sqrt{-1}$.
For $u = 0,1,2,...,M-1$ and $v = 0,1,2,...,N-1$.

$$\Im^{-1}\{F(u,v)\} = f(x,y) = \sum_{u=0}^{M-1}\sum_{v=0}^{N-1} F(u,v)e^{j2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)} \tag{2}$$

For $x = 0,1,2,...,M-1$ and $y = 0,1,2,...,N-1$.

5.1.2 Magnitude, Phase, & Power Spectrum

The Fourier transform of a 2-D real function is complex. We can express it as

$$F(x,y) = R(u,v) + jI(u,v) \tag{3}$$

where $R(u,v)$ and $I(u,v)$ are the real and imaginary components of $F(u,v)$, respectively. Equation (3) is often expressed in the following exponential form

$$F(u,v) = |F(u,v)|e^{j\phi(u,v)} \tag{4}$$

where

$$|F(u,v)| = \sqrt{R^2(u,v) + I^2(u,v)} \tag{5}$$

and

$$\phi(u,v) = \tan^{-1}\left[\frac{I(u,v)}{R(u,v)}\right] \tag{6}$$

The magnitude function $|F(u,v)|$ is called the Fourier spectrum of $f(x,y)$ and $\phi(u,v)$ its phase angle. The square of the spectrum

$$P(u,v) = |F(u,v)|^2 = R^2(u,v) + I^2(u,v) \tag{7}$$

is commonly referred to as the Power Spectrum or spectral density of $f(x,y)$ [3]. Since the range of the Fourier spectra is usually higher than display device can handle, the log of the power spectrum is often taken to compress the values.

5.1.3 Translation

The magnitude of the Fourier transform is not affected by shifts in the signal, which is known as the Fourier Shift Theorem. We will later exploit this property to detect rotations of a function f independent of translations.

Suppose $f(x,y)$ is translated by (x_0, y_0). Then,

$$\Im\{f(x-x_0, y-y_0)\} = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x-x_0, y-y_0)e^{-j2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)} \tag{8}$$

For $u = 0,1,2,...,M-1$ and $v = 0,1,2,...,N-1$

Let $\alpha = x - x_0$, $\beta = y - y_0$. Direct substitution yields

$$\Im\{f(x-x_0, y-y_0)\} = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(\alpha,\beta)e^{-j2\pi\left(\frac{u(\alpha+x_0)}{M}+\frac{v(\beta+y_0)}{N}\right)} \tag{9}$$

$$= e^{-j2\pi\left(\frac{ux_0}{M}+\frac{vy_0}{N}\right)}\frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(\alpha,\beta)e^{-j2\pi\left(\frac{u\alpha}{M}+\frac{v\beta}{N}\right)} \tag{10}$$

$$= e^{-j2\pi\left(\frac{ux_0}{M}+\frac{vy_0}{N}\right)}F(u,v) \tag{11}$$

From equation (11), we can see that the effect of translating f by (x_0, y_0) is to multiply $F(u,v)$ by $e^{-j2\pi\left(\frac{ux_0}{M}+\frac{vy_0}{N}\right)}$ meaning

$$f(x-x_0, y-y_0) = F(u,v)e^{-j2\pi\left(\frac{ux_0}{M}+\frac{vy_0}{N}\right)} \tag{12}$$

Note that this multiplication does not affect the magnitude since

$$F(u,v) = \left|F(u,v)e^{-j2\pi\left(\frac{ux_0}{M}+\frac{vy_0}{N}\right)}\right| \tag{13}$$

5.1.4 Rotation

A key property of the Fourier transform, which allows us to detect rotations in images independently of translations, is that rotations in the spatial domain are preserved in the frequency domain [3].

Let $f(x, y)$ be a two-dimensional function rotated by an amount θ_0 in the counter-clockwise direction. Then the following holds:

Let $\alpha = x\cos\theta_0 + y\sin\theta_0$ $\beta = -x\sin\theta_0 + y\cos\theta_0$. Then $x = \alpha\cos\theta_0 - \beta\sin\theta_0$, $y = \alpha\sin\theta_0 + \beta\cos\theta_0$.

$$\Im\{f(x,y)\} = F(u,v) =$$
$$\frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x\cos\theta_0 + y\sin\theta_0, -x\sin\theta_0 + y\cos\theta_0)e^{-j2\pi\left(\frac{ux_0}{M}+\frac{vy_0}{N}\right)} \quad (14)$$

Substitution gives

$$\Im\{f(\alpha,\beta)\} = \frac{1}{MN}\sum_{\alpha=0}^{M-1}\sum_{\beta=0}^{N-1} f(\alpha,\beta)^{-j2\pi\left(\frac{u(\alpha\cos\theta_0 - \beta\sin\theta_0)}{M}+\frac{v(\alpha\sin\theta_0 + \beta\cos\theta_0)}{N}\right)} \quad (15)$$

$$= F(u\cos\theta_0 - v\sin\theta_0, u\sin\theta_0 + v\cos\theta_0) \quad (16)$$

From equation (16) we can see that a clockwise rotation by θ_0 in the spatial domain equals a counter-clockwise rotation by θ_0 in the Fourier frequency domain; therefore, rotation is preserved in both domains [3].

5.2 Expected Value & Entropy

We will briefly define some information theory concepts that are tied to the derivation of Mutual Information. Let X be a discrete random variable with alphabet $\Re$ and probability mass function $p(x) = \Pr\{X = x\}, x \in \Re$. The probability mass function is denoted by $p(x)$ rather than $p_X(x)$. The expected value $E(X)$ is defined by

$$E(X) = \sum_{x \in \Re} x p(x) \tag{17}$$

Entropy is a measure of uncertainty of a random variable [3]. The entropy of a discrete random variable X with a probability mass function $p(x)$ is defined by

$$H(X) = -\sum_{x \in \Re} p(x) \log p(x) \tag{18}$$

The entropy of X can also be interpreted as the expected value of $\log \frac{1}{p(X)}$, where X is drawn according to the probability mass function $p(x)$ [3]. In summary, this means that $H(X)$ can be defined as

$$H(X) = E \log \frac{1}{p(X)} \tag{19}$$

5.2.1 Joint Entropy

The joint entropy is a statistic that summarizes the degree of dependence of two random variables. The joint entropy $H(X,Y)$ of a pair of discrete random variables (X,Y) with a joint distribution $p(x,y)$ is defined as

$$H(X,Y) = -\sum_{x \in \Re} \sum_{y \in \mathrm{Y}} p(x,y) \log p(x,y) \tag{20}$$

which can also be expressed as

$$H(X,Y) = -E \log p(X,Y) \tag{21}$$

5.2.2 Conditional Entropy

We define the conditional entropy of one random variable given another as the expected value of the entropies of the conditional distributions, averaged over the conditioning random variable [4]. This statistic summarizes the randomness of Y given knowledge of X. The conditional entropy $H(Y \mid X)$ is expressed as

$$H(Y \mid X) = -\sum_{x \in \Re} p(x) H(Y \mid X = x) \tag{22}$$

$$= -\sum_{x \in \Re} p(x) \sum_{y \in \mathrm{Y}} p(y \mid x) \log p(y \mid x) \tag{23}$$

$$= -\sum_{x \in \Re} \sum_{y \in \mathrm{Y}} p(x,y) \log p(y \mid x) \tag{24}$$

$$= -E \log p(Y \mid X) \tag{25}$$

Given these derivations (20-25), the entropy of a pair of random variable can be shown to be the entropy of one plus the conditional entropy of the other

$$H(X,Y) = H(X) + H(Y \mid X) \tag{26}$$

6 DETERMINING ROTATIONS AND TRANSLATIONS

As mentioned before, the goal of registration is to find the "best" possible rotation and translation for aligning two images. In this section, we describe the way in which the mathematical concepts from Section 5 can be applied to find optimal rotations and translations.

6.1 Finding Optimal Translation via Cross Correlation

Let f_1 and f_2 denote two images, with f_2 a shifted copy of f_1 giving $f_2(x,y) = f_1(x - x_0, y - y_0)$. This section describes procedures for determining the translation shift (x_0, y_0).

6.1.1 Cross Correlation

An efficient method for finding the translation follows from the Fourier correlation theorem. We calculate the correlation in $O(S \log S)$ time (versus $O(S^2)$) using the DFT (calculated via the FFT) to convert cyclic convolution into point-wise multiplication [7]. The correlation theorem can be stated as follows

$$\frac{1}{MN} \sum_{i}^{M-1} \sum_{j}^{N-1} f_1^*(i,j) f_2(x+i, y+j) = \frac{1}{MN} F^{-1}\{F_1^*(u,v) F_2(u,v)\} \tag{27}$$

Where * denotes the complex conjugate, and M and N represent the dimensions of the image. The location of the maximum peak should represent the best possible translation shift (x_0, y_0). Since our test image is most likely not exactly a translated and rotated version of the reference image, we take the largest peak as holding the "best" translation parameter for registering f_2 and f_1.

In summary, the cross correlation can be used in finding the shift, (x_0, y_0), with two main steps:

1. Calculate the 2-dimensional cross correlation of f_1 and f_2
2. Find the x-axis and y-axis location of the maximum peak in the correlation. These coordinates give the desired shift (x_0, y_0).

Figure 7 below shows the process of finding the shift (x_0, y_0) using the cross correlation:

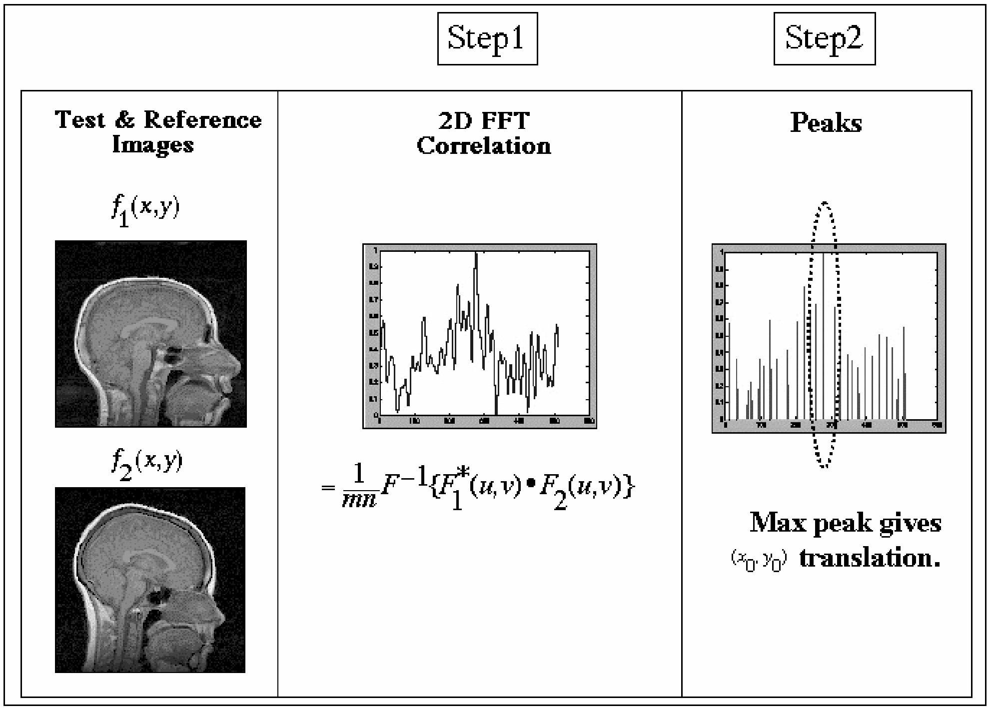


Figure 7. Determining translations using the cross correlation

6.1.2 Phase Only Correlation

Phase only correlation (POC) makes use of the observation that the translations are only contained within the phase, defined in equation (6), of the Fourier transform. The amplitudes of each Fourier coefficient are normalized, effectively serving as a high-pass filter over the image. High-pass filters eliminate low-frequency components of the image. By de-emphasizing low frequency components, we are able to focus on the most distinct components of the image.

$$\frac{1}{MN} F^{-1}\left(\frac{F_1^*(u,v)}{|F_1(u,v)|} \frac{F_2(u,v)}{|F_2(u,v)|} \right) \tag{28}$$

As in the steps used for the cross correlation, the location of the peak in the POC identifies the best possible translation shifts (x_0, y_0), and. the maximum peak is chosen as the translation parameter. The peaks obtained using the POC are sharper and more prominent than those produced with the cross correlation, which makes them easier to find in images with dissimilar contrast.

6.2 Finding Optimal Rotation via Cross Correlation

We explain an approach to find rotations by converting the power spectrum (7) into polar coordinates in order to reduce the rotation problem into one of translation. Once the problem is given in terms of finding a translation, we can apply the cross correlation method explained in Section 6.1.1 to obtain an answer.

Some basic properties of the Fourier Transform are used to separate out the possible rotation angles, θ_0, for the test image independently from the (x_0, y_0) translation. The technique to find the rotation is based on the translation invariant feature of the power spectrum. Let f_1 and f_2 denote two images, with f_2 being a rotated and shifted copy of f_1,

$$f_2(x,y) = f_1((x\cos\theta_0 + y\sin\theta_0) - x_0, (-x\sin\theta_0 x + \cos\theta_0) - y_0) \tag{29}$$

So f_2 is f_1 rotated by θ_0 and shifted by (x_0, y_0). Let $F_i = \Im\{f_i\}$. By properties of the Fourier Transform (1), we see that the Power Spectrum of both images is

$$\left|F_2(u,v)\right|^2 = \left|F_1(u\cos\theta_0 + v\sin\theta_0, -u\sin\theta_0 + v\cos\theta_0)\right|^2 \tag{30}$$

In polar coordinates (setting $u = r\cos\psi$, $v = r\sin\psi$) the (30) becomes

$$\left|F_2(r,\psi)\right|^2 = \left|F_1(r,\psi - \theta_0)\right|^2. \tag{31}$$

From (31), we can also see that rotations of $f_1(x,y)$ by θ_0 appear as a vertical shift along the ψ axis, while the magnitudes of F_1 and F_2 remain the same.

Thus, given two blocks f_1 from the test images and f_2 from the reference image, the rigid rotation and translation parameters can be determined by first finding θ_0, then (x_0, y_0). The following steps detail how to find the rotation angle:

1. Window test and reference blocks $f_1(x,y)$ and $f_2(x,y)$
2. Obtain Power Spectrums for both blocks $F_1(u,v)$ and $F_2(u,v)$
3. Convert the Power Spectrum to polar coordinates $F_1(\omega,\theta_0)$ and $F_2(\omega,\theta_0)$ using correlation.
4. The coordinates of the max correlation value give the rotation angle for aligning $f_1(x,y)$ and $f_2(x,y)$.

Figure 8 below details the process of finding rotations.

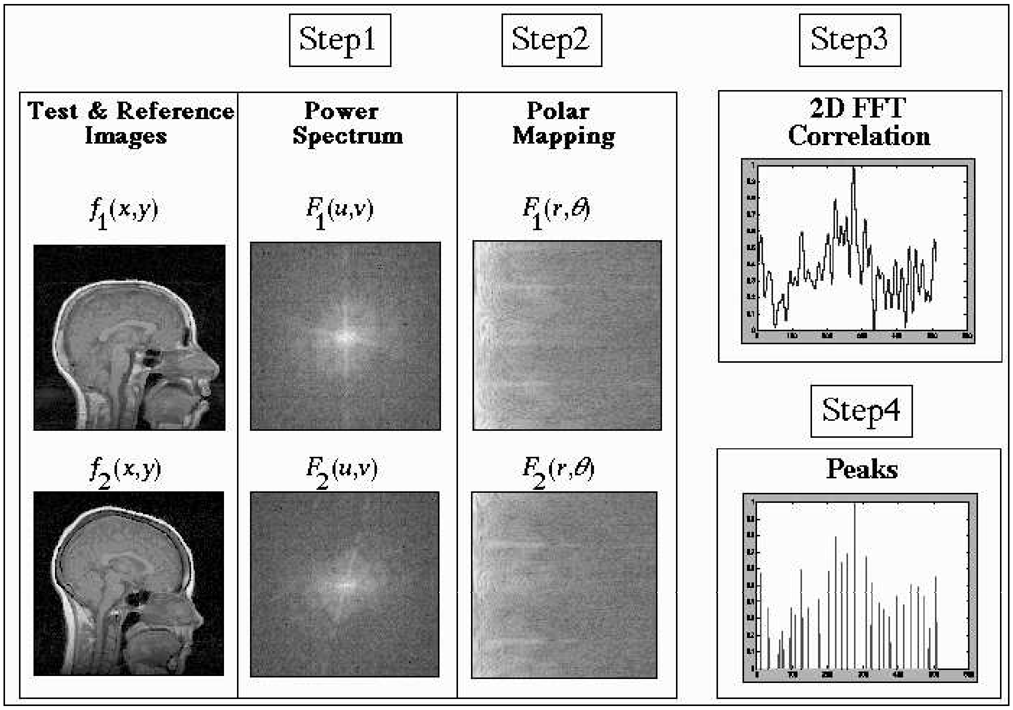


Figure 8. Determination of rotations using Cross Correlation

6.3 Method for registration by maximizing Mutual Information

Maximizing the mutual information between two images, X and Y, has been shown as an effective means to register images independent of contrast [1,2]. The mutual information (MI) of X and Y can be expressed as follows

$$MI(X,Y) = H(Y) - H(Y \mid X) \qquad (32)$$

Note: $H(Y)$ is the entropy of Y and $H(Y \mid X)$ is the conditional entropy of Y, equations (18,25).

Therefore, mutual information serves to describe the reduction in the uncertainty of X due to knowledge of Y. Since $H(X,Y) = H(X) + H(Y \mid X)$ as shown in equation (26) the MI can be expressed as

$$MI(X,Y) = H(X) + H(Y) - H(X,Y) \qquad (33)$$

By maximizing mutual information, we reduce the uncertainty of Y given X. For clarity, the diagram below depicts the discussed relationships between mutual information and entropy in a set theory form.

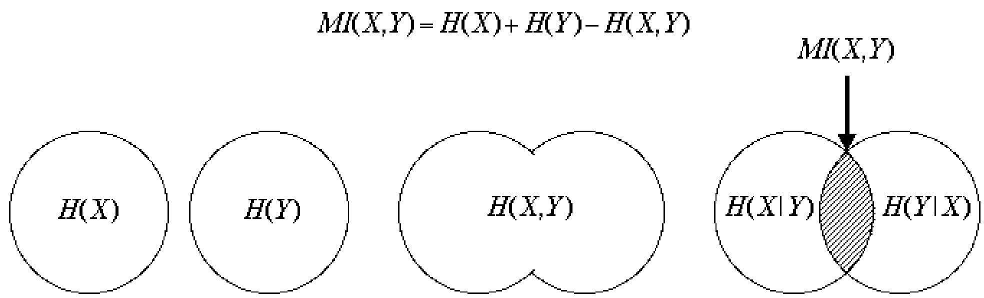


Figure 9. Relationship between mutual information and entropy

Using an exhaustive mutual information approach to find the rotations and translations for registering images X and Y, the steps are as follows:

1. Calculate the mutual information of X and Y for each rotation (360 MI calculations or fewer depending on the angle increments).
2. For each rotation, calculate the mutual information for all the possible translations of the test image to the reference image.
3. Find the overall maximum MI value. The coordinates of this max value give the answer for the rotation θ_0 and translation (x_0, y_0).

In pseudocode, the steps to use mutual information to align two $n \times m$ images are as follows:

```
// angle_incr determines the step for possible rotations
For rot = 1 to 360, n+= angle_incr {
   // Rotate the Test image by rot degrees
   Temp = Rotate(Test,rot)

   // Calculate MI for all translations for the rotation
   For dy = 1 to N {
     For dx = 1 to M {

        // Translation the temp image by dx, dy
        Temp = Translate(Temp, dx, dy)

        // Calculate MI of the Temp & Reference images
        MI_ans(rot,dx,dy)= MI(Temp,Ref)
     }
   }
}

// The answer for the optimal rot and x,y shift is where
// the MI is maximized
Max(MI_ans,rot,dx,dy)
```

The test image can be rotated and translated in $O(S)$ time, where S is the full $N \times M$ image matrix. Calculating the MI of two $N \times M$ images requires $O(S)$ operations. Our pseudocode suggests that the running time of the mutual information maximization algorithm is $O(S^2)$. This upper bound on the running time is significant for high-resolution medical images.

7 CONCLUSIONS

We have presented a method for performing a multi-scale registration for a test and reference image that employs both correlation and mutual information techniques. The goal in combining these methods is to arrive at a technique that is more contrast independent than a local error approach and less computationally intensive than exhaustive mutual information calculations. From the initial results presented, it appears that the hybrid method gives good results with our test data of T1 and T2 class images. Continued testing should be performed with a wider data set to stretch the limits of the method further. Possible modifications to the current method include trials that use the phase only correlation for finding the rotations, modifications of the range value, incorporation of different rotation/translation detection techniques, and adjustments to the template files which determine the block size and position for each iteration.

8 ACKNOWLEDGEMENTS

We would like to thank Senthil Periaswamy , Dartmouth College, for his help and guidance throughout the project.

References

[1] F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, and P. Suetens, Multi-Modality Image Registration by Maximization of Mutual Information, in IEEE Proceedings of Mathematical Methods in Biomedical Image Analysis, (1996), 14-22.

[2] P. Viola and W. M. Wells III, Alignment by Maximization of Mutual Information, Fifth International Conference on Computer Vision, IEEE, Cambridge, MA (1995), 16-

[3] R. C. Gonzalez and R. E. Woods, Digital Image Processing, Addison-Wesley Publishing Company, Reading, (1993), 81-127.

[4] T. M. Cover and J. A. Thomas, Elements of Information Theory, John Wiley & Sons Inc., New York, (1991), 5-33.