Spectrum of Engineering Sciences
ISSN (e) 3007-3138 (p) 3007-312X
Volume 4, Issue 4, 2026

COMPARISON OF RIGID REGISTRATION WITH DIFFERENT OPTIMISATION TECHNIQUES

Aizaz Hussain*[1], Muhammad Yousuf Tufail[2], Saima Gul[3], Anum Zaib[4]

*[*1,4]MS Scholar, Department of Mathematics, NED University Karachi, Sindh, Pakistan*
*[*2]Associate Professor, Department of Mathematics, NED University Karachi, Sindh, Pakistan*
[3]Assistance Professor, Department of Mathematics, NED University Karachi, Sindh, Pakistan

[*1]Aizazhussain574@gmail.com, [*2]tufail@neduet.com, [3]sagul@neduet.com, [4]anumzaib015@gmail.com
DOI: https://doi.org/

Keywords
Algorithms, Coarse Search, Gradient Descent, Image Registration, Rigid Group.

Article History
Received: 04 March 2026
Accepted: 15 April 2026
Published: 01 May 2026

Corresponding Author: *
Aizaz Hussain

Muhammad Yousuf Tufail

Abstract
The process or technique of matching the appearance of two or more images by determining an alignment between them is known as image registration. This is basically aligns two images geometrically. In this study, two-dimensional image registration is presented using the rigid group. This group is a finite dimensional group under composition; it is four-dimensional in this particular case. The dimensions of the rigid group include scaling, rotation, and translations along the axes. This paper presents methods that use a discretized objective function to construct rigid transformations. Based on SSD (sum of the squares of the distances between the pixel intensities), this objective function calculates the difference between the images. In this paper we compare two image registration optimization algorithms and applied them to six different image registration examples. One is the coarse search, and the other is the gradient descent, implemented using MATLAB'S least-squares optimisation (lsqnonlin). The coarse search algorithm is used to explore the transformation domain and provide an initial estimate, which is then refined using the gradient descent method. The proposed algorithms is implemented on a variety of images, mostly our own captured images. The numerical examples demonstrate that the combined use of coarse search and gradient descent improves registration accuracy in several cases, although both methods may still be affected by local minima in certain situations..

1. INTRODUCTION

The process or technique of matching the appearance of two or more images by determining an alignment between them is known as image registration. This essentially aligns two images geometrically [1-6, 10]. It has been extensively researched and applied in a number of fields, such as medical imaging [9] and biology (in the context of morphometric) [7-8]. The collection of acceptable transformations that are used to deform the images is one of the fundamental image registration choices [11]. In this paper, we use the rigid group for two-dimensional image registration. This paper focuses on the use of a coarse search and gradient descent Algorithms for objective function optimization. The proposed methods are applied to several examples in order to demonstrate its effectiveness in image registration.

1.1 Objective Function

Consider the couple of images I_1 and I_2, usually referred to as source and target, are defined in a particular domain $\boldsymbol{\mu}$ (typically, $\boldsymbol{\mu} \subset \mathbb{R}^2$), for 2D grey scale images), image registration can be described as a minimizing objective or error function that evaluates the discrepancy between pictures with a deformation variation of $\boldsymbol{\varphi}^{-1}$ which implement on any of the images. (Use of

$\boldsymbol{\varphi}^{-1}$ is clarified in [11]). Objective function is:

$$\mathrm{E}_{I_1,I_2}(\varphi) = \int_{\mu} \|\mathrm{I}_1 \circ \varphi^{-1}(x_1, x_2) - \mathrm{I}_2(x_1, x_2)\|^2 \, dx_1 dx_2, \qquad \forall (x_1, x_2) \in \mu \qquad (1.1)$$

After applying the transformation φ and the optimization approach, the norm $\|\cdot\|$ is used to calculate the discrepancy between the target picture and the transformed source. The above equation (1.1) is used to register two-dimensional greyscale pictures. However in this paper, we consider the rigid group, which is formed by translation along the x-axis and y-axis, rotations, and scalings, consisting of four parameters in two dimensions.

2. REGISTRATION OF IMAGES WITH RIGID REGISTRATION

In image registration, One of the most often utilized group is the rigid group [11]. For this group the major indicators are scalings, rotations, and arbitrary translations (in 2-dimensional) rigid group [11-12]. The transformation (which is now called the rigid transformation) associated with rigid group is explained below:

2.1 Rotation

Suppose an arbitrary $\begin{pmatrix} x \\ y \end{pmatrix} \in \mathbb{R}^2$. Let $\theta \in [0, 2\pi)$ be the angle of rotation transformation $\boldsymbol{\varphi_1}$ to generate the new point $\begin{pmatrix} x' \\ y' \end{pmatrix} \in \mathbb{R}^2$ as defined by:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \varphi_1 \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

2.2 Scaling

Suppose an arbitrary $\boldsymbol{\lambda} \in \mathbb{R}^2$ is a non-zero number so uniform scaling transformation $\boldsymbol{\varphi_2}$ over $\begin{pmatrix} x \\ y \end{pmatrix}$ is defined below:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \varphi_2 \begin{pmatrix} x \\ y \end{pmatrix} = \lambda \begin{pmatrix} x \\ y \end{pmatrix}$$

2.3 Translation

Suppose $\boldsymbol{t_x}, \boldsymbol{t_y} \in \mathbb{R}$ are the translations in the horizontal and vertical axis respectively. The translation transformation $\boldsymbol{\varphi_3}$ over $\begin{pmatrix} x \\ y \end{pmatrix}$ is defined below:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \varphi_3 \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x + t_x \\ y + t_y \end{pmatrix}$$

The rigid transformation is the combination of the all above mentioned transformations. The compact form of the rigid transformation $\boldsymbol{\varphi}$ is defined below:

$$\begin{pmatrix} \mathrm{x}' \\ \mathrm{y}' \end{pmatrix} = \varphi \begin{pmatrix} \mathrm{x} \\ \mathrm{y} \end{pmatrix}$$
$$= (\varphi_3 \circ \varphi_2 \circ \varphi_1) \begin{pmatrix} \mathrm{x} \\ \mathrm{y} \end{pmatrix}$$

$$= \lambda \begin{pmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix}, \quad \theta \in [0, 2\pi), 0 \neq \lambda \in \mathbb{R} \tag{2.1}$$

The collection of all such $\boldsymbol{\varphi}$ forms a group under composition, which is known as rigid

Group. (From Equation 2.1) The inverse of a rigid transformation is:

$$\begin{pmatrix} x \\ y \end{pmatrix} = \frac{1}{\lambda}\begin{pmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{pmatrix}^{-1}\left\{\begin{pmatrix} x' \\ y' \end{pmatrix} - \begin{pmatrix} t_x \\ t_y \end{pmatrix}\right\}$$
$$= \frac{1}{\lambda}\begin{pmatrix} cos\theta & sin\theta \\ -sin\theta & cos\theta \end{pmatrix}^{-1}\left\{\begin{pmatrix} x' \\ y' \end{pmatrix} - \begin{pmatrix} t_x \\ t_y \end{pmatrix}\right\} \tag{2.2}$$

Since $\boldsymbol{\varphi}$ and $\boldsymbol{\varphi^{-1}}$ are both diffeomorphisms, the rigid group is a subgroup of all diffeomorphisms of $\mathbb{R}^2$.

Definition 2.1. Let's suppose that $\boldsymbol{I_1}$ *and* $\boldsymbol{I_2}$ are two greyscale images. We define a matching functional:

$$\mathrm{E}(\varphi) = \int_{\mu} (\mathrm{I}_1 \circ \varphi^{-1}(\mathrm{x}) - \mathrm{I}_2\,(\mathrm{x}))^2 dx\, dy, \qquad \mathrm{x} = (\mathrm{x}, \mathrm{y})^T \in \mu \subset \mathbb{R}^2 \tag{2.3}$$

If φ denotes a rigid group element, then the technique of determining a **φ** that minimises Equation (2.3) is known as rigid registration.

The continuous form of the objective function is Equation (2.3). We must discretise this continuous form in order to perform numerical computations. Therefore, we must specify a discrete domain, which we denote as $\boldsymbol{K}$, which is basically a domain of discrete pictures. Suppose $\boldsymbol{L} \in \boldsymbol{Z}^{+}$ and let,

$$K = \{ ([0, L-1]/(L-1) - 0.5) \times ([0, L-1]/(L-1) - 0.5) \}$$

Is a grid of squares with $\boldsymbol{L}^2$ grid nodes. Figure (2.1) shows an example with $\boldsymbol{L = 100}$ grid nodes. We shall utilise $\boldsymbol{L = 100}$ all over this thesis, unless otherwise specified.

Discrete objective function is defined as follows (in which $\boldsymbol{x_{ij}}$ indicate the picture component at location $(\boldsymbol{i}, \boldsymbol{j})$ of an image matrix):

$$\mathrm{E}(\varphi) = \sum_{i=1}^{L}\sum_{j=1}^{L}\{(\mathrm{I}_1 \circ \varphi^{-1})(x_{ij}) - \mathrm{I}_2\,)\}^2 \tag{2.4}$$

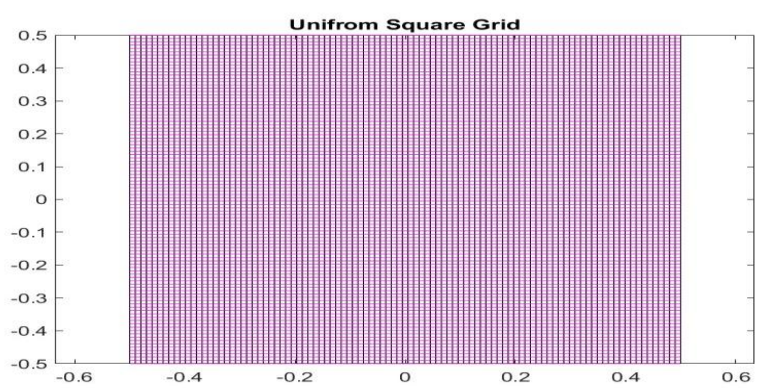


Figure 2.1: The grid domain has 10,000 node points in [-0.5, 0.5] × [-0.5, 0.5] range along with L = 100

ISSN (e) 3007-3138 (p) 3007-312X Volume 4, Issue 4, 2026

(there are 100 node points on each side).

The calculation for $\boldsymbol{I_1} \circ \boldsymbol{\varphi}^{-1}(\boldsymbol{x_{ij}})$ are as follows:

1. If $\boldsymbol{\varphi}^{-1}(\boldsymbol{x_{ij}}) \in \boldsymbol{\mu}$, we apply bilinear interpolation.
2. We utilise a constant background value if $\boldsymbol{\varphi}^{-1}(\boldsymbol{x_{ij}}) \notin \boldsymbol{\mu}$.

The images in the registration examples in this paper are always assumed to have constant backgrounds of 0 (black) or 1 (white), although in some cases the image does not reach its background value entirely before the boundaries of the image are reached.

Our goal is to minimize E(φ) for φ ∈ G. Equation (2.4) represents the general form of the least-squares optimisation function [8] for any choice of planar transformation group (that is, G is a Lie group that acts on the plane; in this paper, G will be a rigid group). This is a numerical optimisation problem [7]. Optimisation is a vast field with numerous known algorithms [9], the application of which is determined by the nature of the objective function. Our goal here is not to survey the topic, but rather to show some easy applications. In this paper, we focus on a two methods, One is coarse search method and other one is gradient descent solving the optimization.

3. COARSE SEARCH METHOD

In the light of recent optimisation studies, it may appear that it is not worth bothering with. In reality, it is still worth thinking about. First, calculating the derivative is optional, i.e., the function $\boldsymbol{\varphi}$ does not necessarily have to be differentiable. Second, and perhaps more importantly, the function $\boldsymbol{\varphi}$ may have several local minima dispersed over the domain and even nested in a fractal-like manner. This can make finding the global minimum problematic for local, derivative-based approaches like gradient descent [6]. Coarse search at least surveys the entire domain [6], or a large part of it, and can provide a solid beginning guess or guesses for more complex methods.

Algorithm (1) coarse search optimization:

Input: I_1 and I_2: source and target images

Output: War φ^{-1} and deformed image $I_1 \circ \varphi^{-1}$

for $\theta = -\pi : \pi/9 : \pi$ **do**
  for $\lambda = 0.1 : 0.1 : 1.5$ **do**
    for $t_x = -0.5 : 0.1 : 0.5$ **do**
      for $y_x = -0.5 : 0.1 : 0.5$ **do**
        use bi-linear interpolation to compute transformed version of the source, $I_1 \circ \varphi^{-1}(x_{ij}) \forall x_{ij} \in S$ computes
        $d = \| I_1 \circ \tilde{\varphi}^{-1} I_2(x_{ij} \|^2 \ \forall x_{ij} \in S$

for the minimum value of d, computes $\varphi^{-1} = \tilde{\varphi}^{-1}$

compute $I_1 \circ \varphi^{-1}$

In the coarse search method the objective function is evaluated at many points $\boldsymbol{\varphi} \in \boldsymbol{G}$; coarse search Algorithm presents an overview of the method, which consists of four nested loops. A decision needs to be made as to how many points $\boldsymbol{\varphi}$ to check, and how they should be distributed in $\boldsymbol{G}$. The rigid group is a 4-dimensional. The coarse search can take $\boldsymbol{\varphi}$ to lie in a grid of values of $\boldsymbol{\theta}, \boldsymbol{\lambda}, \boldsymbol{t_x}$ and $\boldsymbol{t_y}$ as illustrated in the coarse search Algorithm. In this Algorithm we follow the similar approach as proposed by Tufail in [11]. The number of values computed in the algorithm are: $M_\theta = 19$, $M_\lambda = 15$, $M_x = 11$, $M_y = 11$, Hence the search is over the product of these sizes, requiring a total of 34,485 evaluations of the objective function. Thus, although the coarse search method is extremely simple, it is necessary to search over relatively small numbers of values for each parameter or else the computational cost becomes prohibitive.

4. RIGID REGISTRATION THROUGH COARSE SEARCH METHOD

In this section, we present the application of the coarse search Algorithm on real-world data. The source images are transformed using known rigid transformations to create the synthetic target images. Three examples are examined.

Example 4.1: For this example, we consider a pair of non-smooth own captured images illustrated in Figure (4.1). To generate the target image, a rigid transformation is applied to the source image. The transformation parameters used in this example are: $\theta = \pi/3$, $\lambda = 1.2$, $t_x = -0.2$, $t_y = 0.2$ coarse search Algorithm is then applied. The optimized parameters obtained at the end of the optimizations process are: $\theta_{opt} = \pi/3$, $\lambda_{opt} = 1.2$, $t_{x_{opt}} = -0.2$, $t_{y_{opt}} = 0.2$. These optimised parameters are then used to transform the source image. Figure (4.2) shows the image registration results. A perfect registration is obtained.

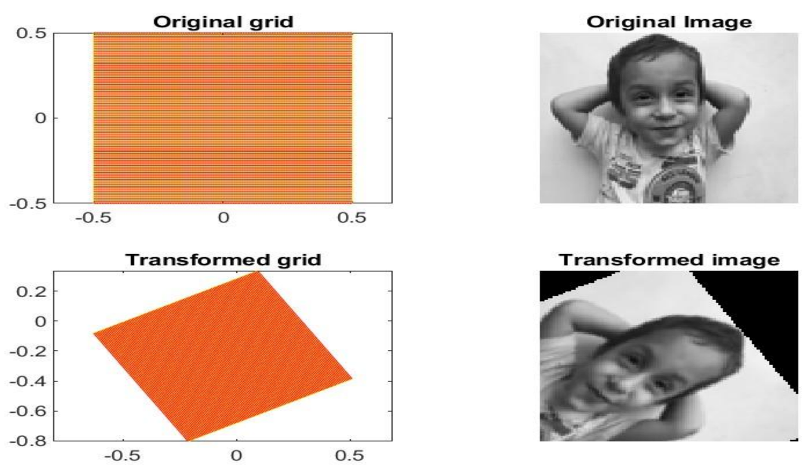
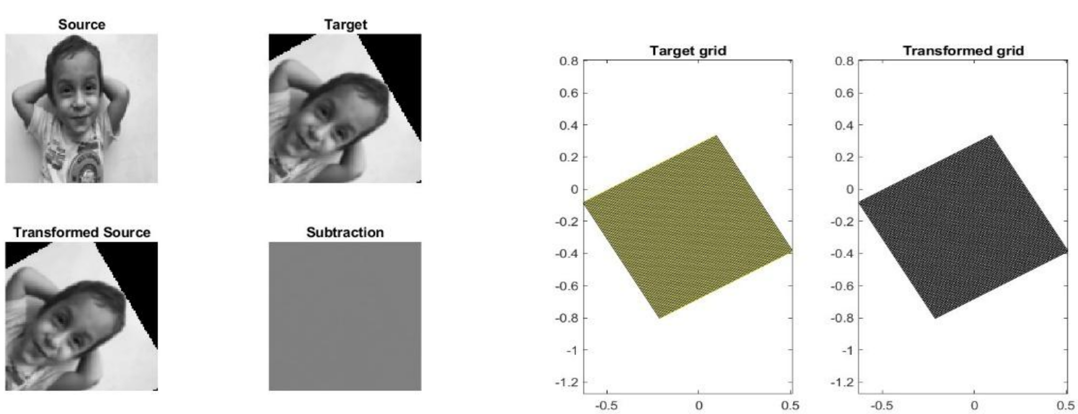
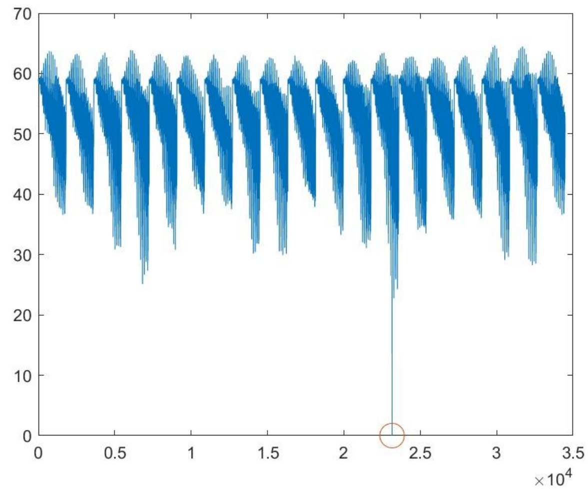


Figure 4.1: In Ex: (4.1), the non-smooth images have been used. The transformed image is generated from the original image using rigid transformation.



ISSN (e) 3007-3138 (p) 3007-312X Volume 4, Issue 4, 2026

(a) (b)



(c)

Figure 4.2: (a): A perfect registration[1] is obtained as a result of rigid registration along with coarse search for Ex: (4.1). (b): Contains corresponding grid plots of the transformed source and target. (c): All objective function values for Ex: (4.1).

Example 4.2: Another non-smooth picture pair can be seen in Figure (4.3). Where the target is obtained from the source image using a rigid transformation with: $\boldsymbol{\theta} = \frac{\pi}{9}$, $\boldsymbol{\lambda} = \mathbf{1}$, $\boldsymbol{t_x} = \mathbf{0.1}$, $\boldsymbol{t_y} = \mathbf{-0.1}$ parameters for this example. We will now apply coarse search Algorithm. The optimized parameters obtained at the end of the optimization process are: $\boldsymbol{\theta_{opt}} = \frac{\pi}{9}$, $\boldsymbol{\lambda_{opt}} = \mathbf{1}$, $\mathbf{t_{x_{opt}}} = \mathbf{0.1}$, $\mathbf{t_{y_{opt}}} = \mathbf{-0.1}$. The optimised parameters are then used to transform the source. A perfect registration is accomplished yet again. The image Registration results are shown in Figure (4.4).

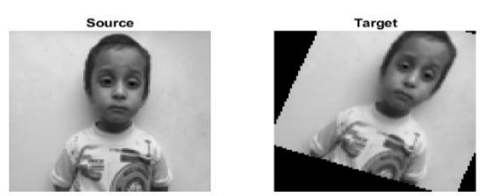
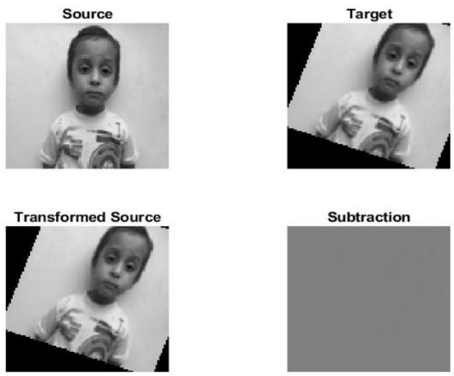
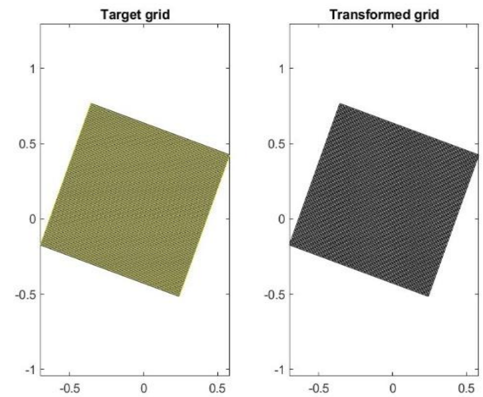


Figure 4.3: In Ex: (4.2), the non-smooth images have been used. The target picture is produced by employing rigid transformation to original image.

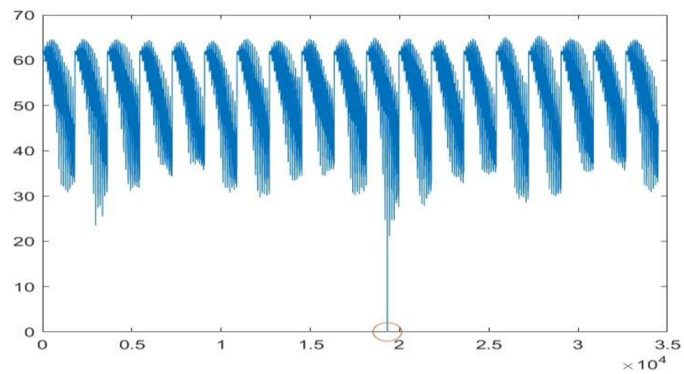
[1] A perfect registration we mean no mismatch between the transformed source and the target.



(a)



(b)



(c)

Figure 4.4: (a) The result is a perfect registration. For Ex: (4.2), Rigid registration with coarse search Algorithm. (b): Contains corresponding grid plots of the transformed source and target. (c): All objective function values for Ex: (4.2)

Example 4.3: In this example, we will look at some other set of non-smooth pictures where the target is obtained from the source image through a rigid transformation shown in Figure (4.5) The parameters for this transformation are as follows: $\boldsymbol{\theta} = \frac{\pi}{2}(\mathbf{1.5708})$, $\boldsymbol{\lambda} = \mathbf{1.4}$, $\boldsymbol{t_x} = \mathbf{-0.15}$, $\boldsymbol{t_y} = \mathbf{0.13}$. The optimised parameters are returned by coarse search Algorithm that are: $\boldsymbol{\theta_{opt}} = \mathbf{1.7453}$, $\boldsymbol{\lambda_{opt}} = \mathbf{1.4}$, $\boldsymbol{t_{x_{opt}}} = \mathbf{-0.1}$, $\boldsymbol{t_{y_{opt}}} = \mathbf{0.1}$. Although these are distinct from those that were use to build the target. The issue would be that the range of values allowed for coarse search does not contain the actual values that were used to build the target. Figure (4.6) shows the image registration results, which show acceptable but not perfect registration.

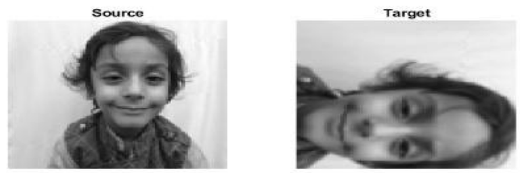
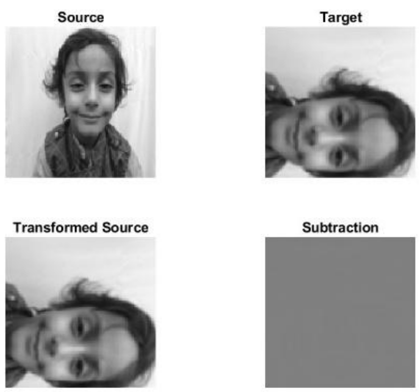
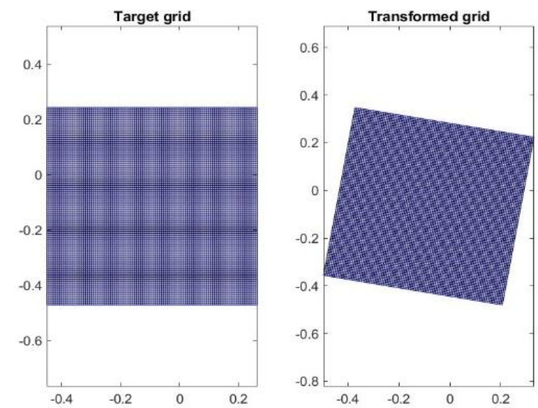


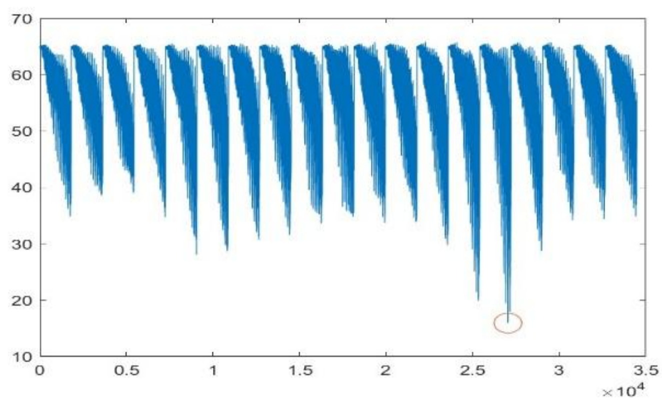
Figure 4.5: The non-smooth images have been used for this Ex (4.3). The target picture is produced by employing rigid transformation to source image.



(a)



(b)



(c)

Figure 4.6: (a): Result of rigid registration by coarse search in above example. A good but imperfect registration has been received. (b): Contain grid plots of the transformed source and target pictures (c): All objective function values. It's worth noting that the y-axis values just go downward to 15.9289, rather than 0 just like in prior cases.

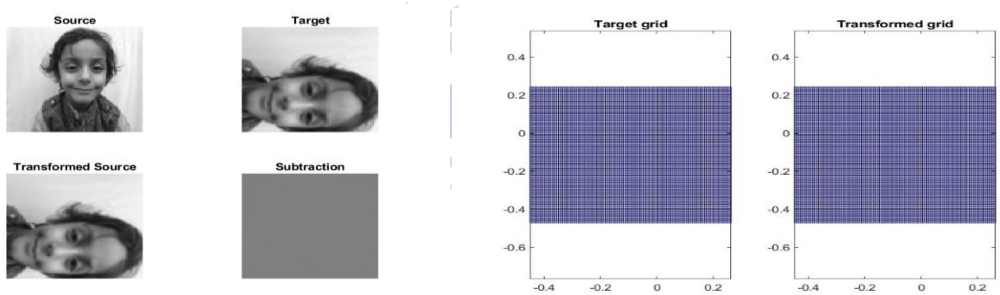
5. GRADIENT DESCENT METHOD

Using the MATLAB optimisation toolbox optimiser lsqnonlin (this tool will be utilised throughout this thesis). This is a nonlinear least-squares optimiser derived from gradient descent techniques and is capable of finding a local minimum of a function $f(x) = (f_1(x), f_2(x), \dots, f_n(x))$ by calculating:

$$min_x \|f(x)\|_2{}^2 = min_x (f_1(x)^2 + f_2(x)^2 + \cdots + f_n(x)^2)$$

Indeed, lsqnonlin has a number of built-in algorithms, such as trust-region-reflective and Levenberg–Marquardt. The first approach is incapable of solving underdetermined systems and is only applicable when the number of unknowns equals the number of equations. Levenberg–Marquardt often works effectively for underdetermined systems. The problem is well-posed in our situation, because the target is derived from the source; thus, we utilise the trust-region-reflective algorithm for image registration. So Starting point (initial guess) is required for the optimisation in order to implement lqnonlin for registration of images. Let us begin gradient descent method.

Example 5.1: In this example, we replicate (Ex: 4.3) and employ the gradient descent method by using the initial guess, such that the starting point will be the outcomes of coarse search optimisation that are (see in Ex: 4.3**)** $\theta_{ini} = 1.7453$, $\lambda_{ini} = 1.4$, $t_{x_{ini}} = -0.1$, $t_{y_{ini}} = 0.1$ Which are closer to the global minimum. When the optimisation process is done, the optimizer manages to discover the parameters that correspond to the global minimum, and that parameters are $\theta_{opt} = 1.5708$, $\lambda_{opt} = 1.4$, $t_{x_{opt}} = -0.15$, $t_{y_{opt}} = 0.13$. The outcomes of this registration process are given in below figure, which show that a perfect registration is achieve.



(a) (b)

Figure 5.1: (a): The outcome of least-square rigid registration for (Ex: 4.3), By implementing lsqnonlin with optimal coarse search outcome as the initial guess, Which yields perfect registration. (b): Corresponding grid plots of the transformed source and target.

Example 5.2: We have another couple of non-smooth pictures, where the target picture is developed the source picture by applying rigid transformation. The following are the parameter values for this transformation: $\boldsymbol{\theta = \pi/4(0.7854)}$, $\boldsymbol{\lambda = 1.25}$, $\boldsymbol{t_x = -0.27}$, $\boldsymbol{t_y = 0.13}$ shown in Figure (5.2). Now Now we implement coarse search algorithm over this example. When the optimisation procedure has been completed, we acquired the optimised parameters, which are: $\boldsymbol{\theta_{opt} = 0.6981}$, $\boldsymbol{\lambda_{opt} = 1.3}$, $\boldsymbol{t_{x_{opt}} = -0.3}$, $\boldsymbol{t_{y_{opt}} = 0.2}$. Now we transform the source by using these optimised parameters. Again a imperfect registration is obtained, The outcomes of the registration depict in Figure (5.2).

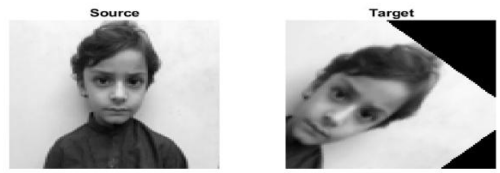
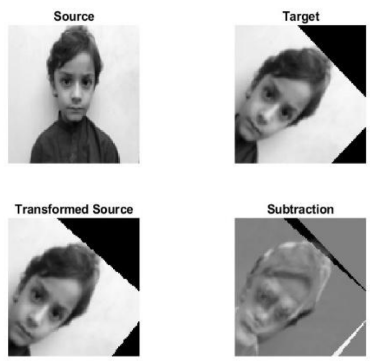
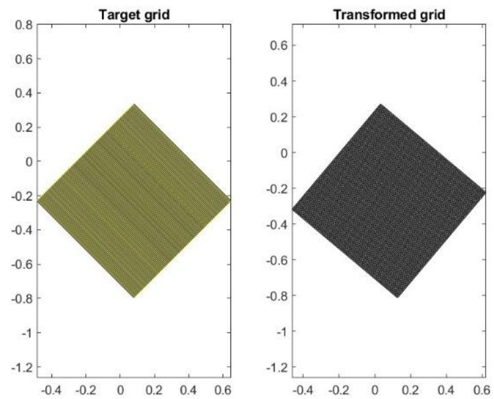
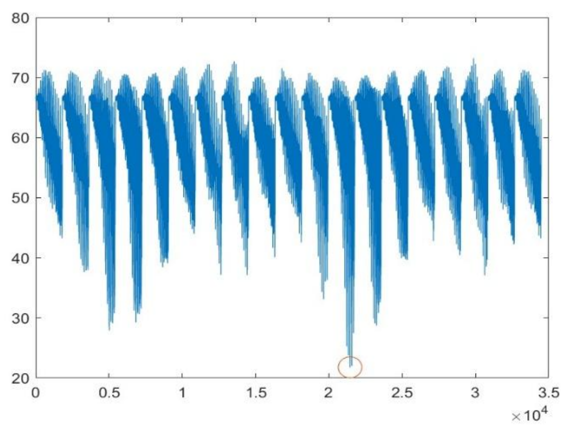


Figure 5.2: (a): For this example, non-smooth images were used. The target picture is produced by employing rigid transformation to source image.





(a) (b)



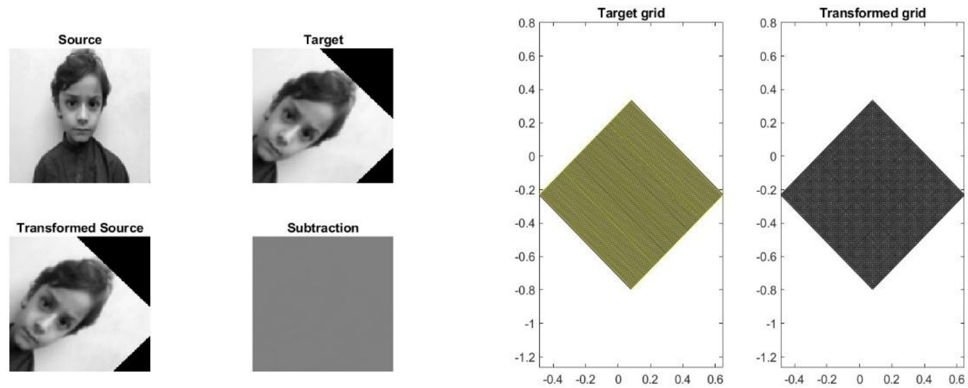
(c)

Figure 5.3: (a): For (Ex: 5.2), the result of rigid registration using coarse search is an acceptable but imperfect registration achieve. (b): Corresponding grids of the transformed source and target pictures. (c): The entire set of values for the objective function for (Ex: 5.2). Again, the values of y-axis only comes down to 21.7735, not to zero.

We'll go over (Ex: 5.2) again, this time with the lsqnonlin optimizer. Since the initial

ISSN (e) 3007-3138 (p) 3007-312X Volume 4, Issue 4, 2026

guess is require in order to implement lsqnonlin. The initial guess will be the outcomes of coarse search optimisation and the values of that parameters are: $\boldsymbol{\theta_{ini}} = \mathbf{0.6981}$, $\boldsymbol{\lambda_{ini}} = \mathbf{1.3}$, $\mathbf{t_{x_{ini}}} = \mathbf{-0.3}$, $\mathbf{t_{y_{ini}}} = \mathbf{0.2}$. Once the optimisation process has been completed, the optimizer is able to identify the optimised parameters that are: $\boldsymbol{\theta_{opt}} = \mathbf{0.7854}$, $\boldsymbol{\lambda_{opt}} = \mathbf{1.25}$, $\mathbf{t_{x_{opt}}} = \mathbf{-0.27}$, $\mathbf{t_{y_{opt}}} = \mathbf{0.13}$. Figure (5.4). The outcomes are shown in the image below. A perfect registration has been obtained;



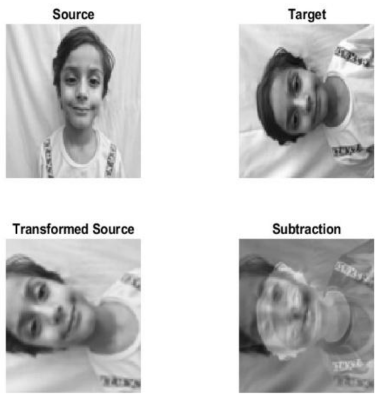
(a) (b)

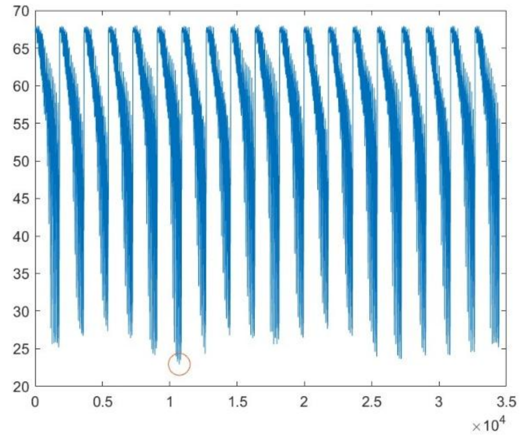
Figure 5.4: (a): The outcome of least-square rigid registration in (Ex: 5.2) Using lsqnonlin with optimal coarse search outcome as the initial guess, which yields perfect registration. (b): Corresponding grid plots of the transformed source and target.

Example 5.4: In this final example, two images are captured from different viewpoints, namely the source and the target. The target is captured 90 degrees of clockwise rotation to the source image, and our goal is to fix the source over the target. Figure (5.5) depicts both pictures, Source and target.

Figure 5.5: For this example, non-smooth pictures were utilised. Note: Here target image is not generated from the source, as in the preceding examples.

For image registration we implemented coarse search algorithm. When the optimization procedure is finished, we have set of parameters $\boldsymbol{\theta_{opt}} = \mathbf{-1.3963}$, $\boldsymbol{\lambda_{opt}} = \mathbf{1.4}$, $\boldsymbol{t_{x_{opt}}} = \mathbf{-0.1}$, $\boldsymbol{t_{y_{opt}}} = \mathbf{0.1}$. Corresponding to the minimum value. Then we transformed the source with rigid transformation by using these optimised parameters. Figure (5.6) depicts the image registration findings, which show adequate but not perfect registration.





(a) (b)

Figure 5.6: (a): For Ex: (5.3), the result of rigid registration using coarse search is an acceptable but imperfect registration achieve. (b): The entire set of values for the objective function in Ex: (4.6). It's worth noting that the y–axis numbers only go down to 27.9462, not zero.

For image registration, we replicate (Ex: 5.3) using the Matlab-based optimizer lsqnonlin. For optimization, we require an initial guess. We begin with the outcomes of the coarse search optimisation that are$\boldsymbol{\theta_{ini}} = \boldsymbol{-1.3963}$, $\boldsymbol{\lambda_{ini}} = \boldsymbol{1.4}$, $\boldsymbol{t_{x_{ini}}} = \boldsymbol{-0.1}$, $\boldsymbol{t_{y_{ini}}} = \boldsymbol{0.1}$. After the optimisation process is finished, the optimiser able to identify the optimised parameters which are:$\boldsymbol{\theta_{opt}} = \boldsymbol{-1.4455}$, $\boldsymbol{\lambda_{opt}} = \boldsymbol{1.2652}$, $\boldsymbol{t_{x_{opt}}} = \boldsymbol{-0.0850}$, $\boldsymbol{t_{y_{opt}}} = \boldsymbol{0.0466}$. The outcomes are shown in the Figure 5.7 below; unfortunately, the registration is unsatisfactory once again.

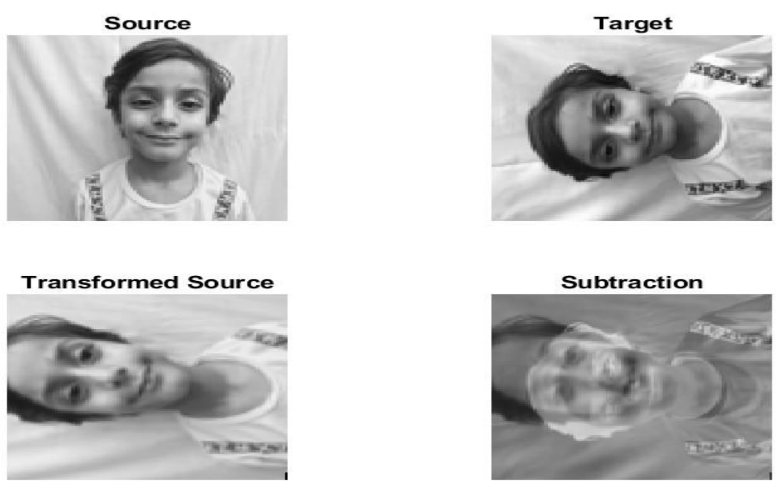


Figure 5.7: The outcome of rigid registration for (Ex:5.4), By applying lsqnonlin with optimal coarse search outcome as the initial guess, Unfortunately again yields imperfect registration.

6. CONCLUSION

In this paper, we compare two image registration optimisation algorithms and apply them to six different image registration examples. One is coarse search, and the other is gradient descent (a MATLAB-based optimiser). Initially, the coarse search method was applied to three image registration examples. Perfect registration was obtained in two examples, while one example yielded poor registration. Method one, on the other hand, has some shortcomings and thus provides poor registration for the remaining examples. As a result, coarse search is not a good option for these examples. However, coarse search has the advantage of surveying the entire domain and providing a good initial guess. To overcome this limitation, after the coarse search, we apply the gradient descent method to the examples in which coarse search produced poor results, using the coarse search outputs as an initial guess. The gradient descent method then yields perfect registration for these examples. As a result, the gradient descent approach is preferred over coarse search; however, coarse search provides a better initial guess, allowing us to employ the gradient descent method effectively. Moreover, gradient descent also has a shortcoming, as it again provides imperfect registration in the last example (see **Ex: 5.3**). It does not guarantee finding the global minimum; although it may find it in some cases, it often converges to a local minimum near the initial guess. This is because there are likely to be multiple local minima in the search space, which can lead the gradient descent to get trapped in local minima. Although finding the global minimum is desirable, it is very difficult without understanding the search space. However, more research into how to eliminate false local minima, particularly shallow ones, could lead to improved registrations. It might be especially useful in examples when the images are not completely aligned.

REFERENCES

[1] B. Zitová and J. Flusser, “Image registration methods: A survey,” Image and Vision Computing, vol. 21, no. 11, pp. 977–1000, 2003..

[2] M. Y. Tufail and S. Gul, “Image registration using the rigid group,” Scientific Inquiry and Review, vol. 7, no. 1, pp. 71–86, 2023.

[3] S. Marsland, R. I. McLachlan, and M. Y. Tufail, “Conformal image registration based on constrained optimization,” The ANZIAM Journal, vol. 62, no. 3, pp. 235–255, 2020.

[4] M. Y. Tufail and S. Gul, “Conformal image registration using the discretized Cauchy–Riemann equations,” The ANZIAM Journal, vol. 67, p. e8, 2025.Thomson, J. A. (1917). On growth and form.

[5] S. Marsland, R. I. McLachlan, and M. Y. Tufail, “Unconstrained optimisation for conformal diffeomorphic image registration,” Bulletin of the Korean Mathematical Society, vol. 63, no. 2, pp. 385–405, 2026.

[6] M. Y. Tufail and S. Gul, “An optimisation over the Möbius group for an optimal solution in image registration,” Punjab University Journal of Mathematics, vol. 57, no. 6, 2025.

[7] D. W. Thompson, On Growth and Form. Cambridge, U.K.: Cambridge University Press, 1917.

[8] M. Y. Tufail and S. Gul, “Conformal patterns in the growth of human skulls,” Acta Biotheoretica, vol. 73, no. 3, p. 13, 2025.

[9] M. I. Miller, A. Trouvé, and L. Younes, "Geodesic shooting for computational anatomy," Journal of Mathematical Imaging and Vision, vol. 24, no. 2, pp. 209–228, 2006.

[10] S. R. Lele and J. T. Richtsmeier, An Invariant Approach to Statistical Analysis of Shapes. Boca Raton, FL, USA: CRC Press, 2001.

[11] M. Y. Tufail, "Image registration under conformal diffeomorphisms," Ph.D. dissertation, Massey University, Palmerston North, New Zealand, 2017.

[12] L. G. Brown, "A survey of image registration techniques," ACM Computing Surveys, vol. 24, no. 4, pp. 325–376, 1992.

[13] C. T. Kelley, Iterative Methods for Optimization. Philadelphia, PA, USA: SIAM, 1999.

[14] J. Glaunès, M. Vaillant, and M. I. Miller, "Landmark matching via large deformation diffeomorphisms on the sphere," Journal of Mathematical Imaging and Vision, vol. 20, no. 1, pp. 179–200, 2004.

[15] M. Uehara, "Pictures by conformal mapping," Forma, vol. 16, no. 1, pp. 1–15, 2001.

[16] A. J. Booker, J. E. Dennis, P. D. Frank, D. B. Serafini, V. Torczon, and M. W. Trosset, "A rigorous framework for optimization of expensive functions by surrogates," Structural Optimization, vol. 17, no. 1, pp. 1–13, 1999.