Amin SADRI, Ali Asghar BEHESHTI SHIRAZI, Masoomeh ZAMENI

Department of Electrical Engineering, Iran University of Science and Technology

New image registration method based on the physical forces

Abstract*. In this paper, we propose a new method for image registration and template matching in which the registration parameters are translation and rotation. This method is based on the physical forces. The assumption is that images are like charged materials that attract each other, which is a unique feature of the proposed method. In this case, one of the images moves in the same direction as the applied force and the other one is still. The movement of the image continues until the resultant force becomes zero. This approach does not estimate the two parameters separately, but they are estimated simultaneously leading to a better optimized set of registration parameters. We compare the computation cost of this algorithm with other area based methods. In conclusion, we show that the approach is successful and yields to better results than mutual information and correlation-like methods.*

Streszczenie. *Przedstawiono nowa metodę rejestracji obrazu bazująca na sile przyciągania. Założono, że obraz może wytwarzać siłę przyciągania. Przedstawiono symulacje i potwierdzono że proponowana metoda daje lepsze rezultaty niż metody korelacyjne. (**Nowa metoda rejestracji obrazu bazująca na sile przyciągania**)*

Keywords: Image registration, Pattern recognition, Area based methods.
Słowa kluczowe: rejestracja obrazu, rozpoznawanie wzoru.

1.Introduction

Image registration is determining the geometrical transformation that brings two sets of data into coincidence in the same coordinate system. Image registration plays a critically important role as a preprocessing step in many computer vision applications and image processing. Zitova et al. [1] [2] presented a survey of recent image registration techniques covering different application areas. They classified registration methods into two categories: the area-based methods and the feature-based methods. Unlike feature-based methods, area-based methods deal with the images without attempting to detect special objects. In this paper, we proposed a new algorithm that takes the area-base approach.

The first category of area based methods is cross-correlation. This method is a classical area-based method that exploits the image intensities directly [3], without any structural analysis [1].

$$CC(i,j)=\frac{\sum_{W}(W-E(W))(I_{(i,j)}-E(I_{(i,j)}))}{\sqrt{\sum_{W}(W-E(W))^{2}}\sqrt{\sum_{I_{(i,j)}}(I_{(i,j)}-E(I_{(i,j)}))^{2}}} \quad (1)$$

This measure of similarity is computed for window pairs from the reference and sensed images and its maximum is searched. The window pairs for which the maximum is achieved are set as the corresponding ones. The CC based registration can also be successfully applied when slight rotation and scaling are present. There are generalized versions of CC for the more geometrically deformed images. These methods compute the CC for each assumed geometric transformation of the sensed image window [4], and they are able to handle even more complicated geometric deformations than the translation-usually the similarity transform. Berthilsson [5] tried to register in this manner even affinely deformed images. The flatness of the similarity measure maxima (due to the self-similarity of the images) and high computational complexity are two main drawbacks of correlation-like methods. However, despite these limitations, the correlation like registration methods are still often used; particularly, on account of their easy hardware implementation, which makes them useful for real-time applications.

Fourier methods are the second category of area based methods. Fourier methods are preferred when an acceleration of the computational speed is needed or the images are corrupted by frequency-dependent noise. These methods exploit the Fourier representation of the images in the frequency domain [6] [7]. The phase correlation method is based on the Fourier Shift Theorem [6] and it was proposed for the registration of the translated images. Furthermore, it computes the cross-power spectrum of the reference and the sensed images and the location of the peak in its inverse is searched [1].

$$\frac{F(f)F(g)^{*}}{\left|F(f)F(g)^{*}\right|}=e^{2\pi i(ux_{0}+vy_{0})} \quad (2)$$

The method shows strong robustness against the correlated and frequency dependent noise and non-uniform, time varying illumination disturbances. The computational time savings are more significant for large images. Moreover, the application of the phase correlation in 3D is described in [8].

The mutual information (MI) methods are the third group of the area-based methods. They have appeared recently and represent the most powerful technique in multimodal registration. The MI, originating from information theory, is a measure of statistical dependency between two data sets. It is particularly suitable for the registration of images from different modalities. MI between two random variables X and Y is given by

$$\begin{aligned} MI(X,Y)&=H(Y)-H(Y\,|\,X)\\ &=H(X)+H(Y)-H(X,Y) \end{aligned} \quad (3)$$

where H(X) =-Ex (log (P(X))) is entropy of random variable and P(X) represents the probability distribution of X. The method is based on the maximization of MI [9].

Some algorithms in computer science are inspired by natural phenomena such as Genetic Algorithm, Neural Networks, and Ant Algorithm. The proposed algorithm which is introduced in this paper is inspired by the mechanics and the electromagnetic. In this method, we use the electrostatic force between charges and define an equivalent force between the image pixels from the reference and the sensed image. By this definition, similar pixels absorb each other more than dissimilar pixels. By means of this virtual force and other mechanical rules, an algorithm for registration of two images is developed.

The proposed method is described in section 2 and the simulation results are given in section 3. We discuss about the method in section 4. Finally, section 5 is devoted to conclusion.

2. The Proposed Method

2.1. Introduction to the Method

This section presents the main idea of the proposed method that will be discussed in details in the next parts. In this method each of the images is considered as a charged object, supposing that the sensed image is able to slide on the referenced image. In each step, the sensed image moves to its counterpart in the referenced image and gets closer to it (Fig 1 and Fig 2.B).

The question is that, how the amplitude and direction of the force is obtained? In other words, how the translation of the sensed image is estimated?

To explain the answer it should be mentioned that the initial position of the sensed image is chosen randomly. In addition, a physical model is used to estimate the movement vector in each step. In this model, every pixel is considered as a charged point. The pixels of the referenced image apply forces to the pixels of the sensed image. These forces have been defined in a way that the pixels with the same color attract each other. So, by this definition, the sensed image attracts its counterpart and moves towards it. Therefore, the sensed image gets closer to its counterpart in each step and finally they match.

2.2. The Force between the two images

Consider a pair of grayscale images, A and B, where B is the sensed and A is the referenced image. These images are shown in Fig 1. Here, we attempt to find the translation parameters and the rotation angle. First an initial position for the image B is chosen then the forces are computed to detect the correct position of B.

To compute the resultant force between the images the following equation is used:

$$\vec{F}_{AB} = \sum_{p\in A}\sum_{q\in B}\vec{f}(p,q) \tag{4}$$

where f(p,q) is the force between p and q.

So, the resultant force for each point of the image B should be computed, and then all these forces summed up to obtain the resultant force.

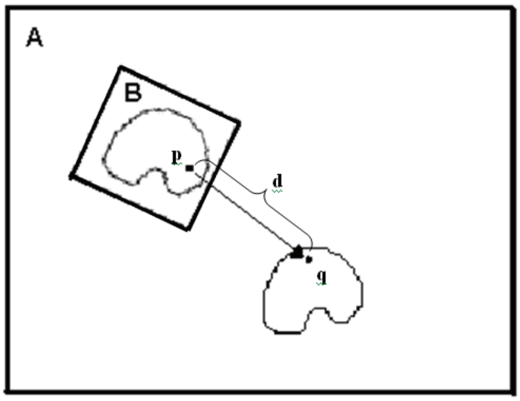


Fig 1. Image B slides on the surface until it matches with its counterpart.

It is supposed that every pixel in an image is a point in its respective object, and the gray level of a pixel is considered as the charge of the respective point. Therefore, based on these assumptions the force between the images was computed.

In electromagnetic, the force between p and q is [10]:

$$F = k\frac{Q_1Q_2}{d^2} \tag{5}$$

where Q1 and Q2 are the charges of the points p and q, k is a constant, and d is the distance between p and q.

Although we use electrostatic force, the same formulations are not applied. Therefore, it is suggested that the magnitude of the force between p and q should be calculated as follows:

$$|F(p,q)| = F_1(p,q)\times F_2(p,q)\times F_3(p,q) \tag{6}$$

where F1 is a function of the pixel values of p and q, F2 is the force that makes the similar objects closer to each other, and finally F3 is the friction force. Direction of F(p,q) is the same as the direction of vector $\vec{pq}$ (Fig 1).

If p and q have the same gray level, F1 must have the maximum value. In this method the following equation was used for F1

$$F_1 = \frac{1}{a_1(C_q - C_p)^2 + a_2} \tag{7}$$

where Cq is the gray level of q, Cp is the gray level of p, and a_1 and a_2 are constant. This equation suggests that if the pixel values of p and q have a significant difference, F1 becomes negligible. Based on this definition, the similar parts attract each other.

According to Eq. (5), the closer charges apply more force to each other. We suggest Eq. (8) for F2. This function increases when the distance between p and q decreases.

$$F_2 = \frac{1}{a_3 + |d|} \tag{8}$$

where a_3 is a positive constant that avoid the division by zero and d is the distance between p and q .

For better convergence of the algorithm, it is supposed that friction increases when the similar parts are close to match. The friction can be useful here because it decreases the movement when the similar parts are close to each other. In order to add the friction to our formula, we assume that there is a surface between the images. In this case, we have vertical (fv) and horizontal (fh) forces which vertical force causes the friction. Fig 2.A shows this situation.

$$f_h = f\sin\alpha \tag{9}$$

$$f_v = f\cos\alpha \tag{10}$$

f and α are shown in Fig 2. From the (4) and (6) equations it can be inferred that fv increases when d decreases and when fv increases friction increases too. So, the following equation for F3 is proposed.

$$F_3 = |d| \tag{11}$$

This means that when the points are close to each other the friction force increases due to the increase of the vertical force - which is against the movement. As previously was discussed, these functions are different from their physical counterparts. They are obtained experimentally. For instance, based on the physical rules the friction is subtracted from the force and resists the movement. However, in our formulation we need a force to stop the movement completely when we get the matching; moreover, dissimilar pixels should not apply force to each other. Therefore, these forces are multiplied.

Now for the purpose of obtaining the resultant force for one point, we should sum all the forces that apply to that point (see (12)).

$$\vec{F}_p = \sum_{q\in B} \vec{F}(p,q) \tag{12}$$

In the Eq. (12), the force that applies to the pixel p of image A is computed.

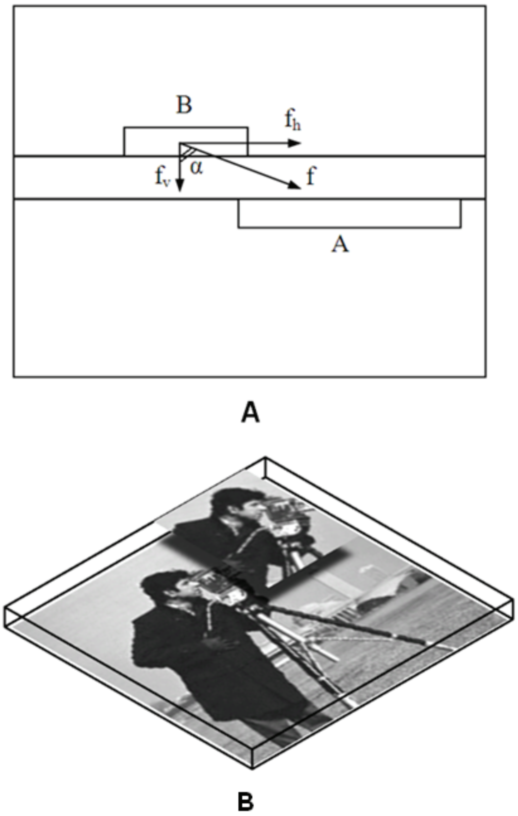


Fig 2. A: Two images with a surface between them; B: The reference and the sensed image and a surface between them

Finally to obtain resultant force for the image, Fp should be calculated.

$$\vec{F}_{AB} = \sum_{p\in A} \vec{F}_p = \sum_{p\in A}\sum_{q\in B} \vec{F}(p,q) \tag{13}$$

In other words, for each point in A such as p, first the resultant force that applies to p is calculated. Then all of these forces are summed to obtain the resultant force that applies to A.

2.3. Estimating the Movement

After finding the resultant force, the next step is to estimate the movement and the rotation angle. Then we slide (move and rotate) image B on the surface until it matches its counterpart. In the proposed method, translation and rotation parameters are calculated using the resultant force. Therefore, we don’t need a full search of rotation and translation to find the best match, because the magnitude and direction of force may give the translation and rotation size.

To estimate the translation, the resultant force between the images should be normalized.

$$\vec{x} = \frac{\vec{F}}{m} \tag{14}$$

$$m = \sum F_1 F_2 \tag{15}$$

where x is the movement vector and m is a scalar used to normalize vector F . White vector in Fig 3.A shows the estimated movement vector and Fig 3.B shows the new position of the sensed image.

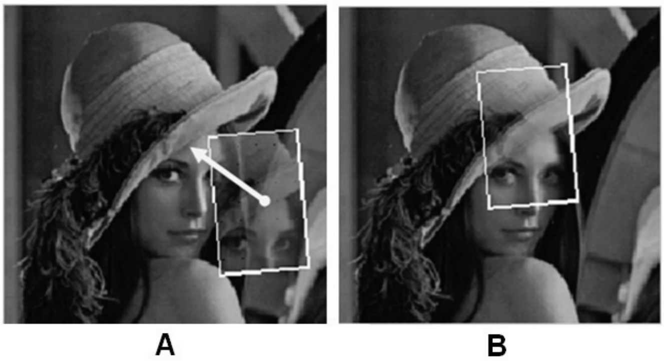


Fig 3. A: Estimated movement vector, B: New position of sensed image after applying the estimated vector

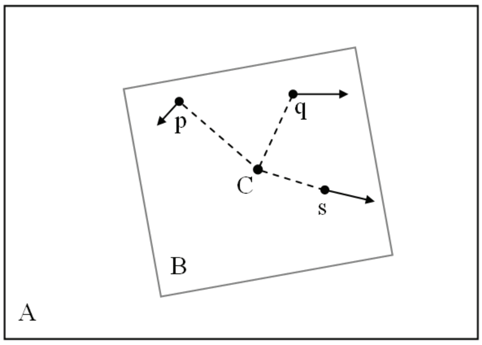


Fig 4. The moment of a force (Torque)

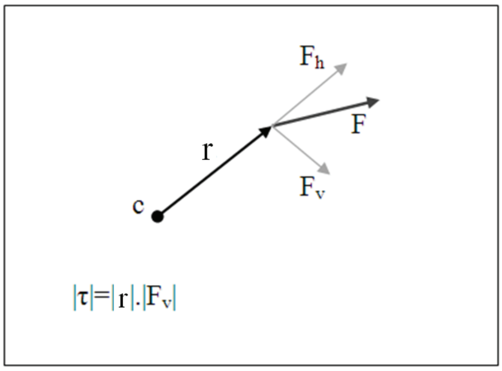


Fig 5. Some pixels of the sensed image (Image B). Point C is the center of the image B.

2.4 Estimating the Rotation Angle

To estimate the rotation angle, the moment of the forces or torque should be calculated. The assumption is that the image rotates around its center point; then the torque is calculated by multiplying the distance by the vertical force (Fig 4). So, after computing the resultant forces of the point the torque is calculated using the following equations:

$$\vec{\tau} = \vec{r} \times \vec{F} \tag{16}$$

$$|\vec{\tau}| = |\vec{r}||\vec{F}|\sin\theta \tag{17}$$

where $\vec{\tau}$ is the torque vector that is perpendicular to the plane, $\vec{r}$ is the lever arm vector (vector from the axis to the point of the force application), and $\vec{F}$ is the force vector. Furthermore, "×" denotes cross product and θ is the angle between the force vector and the moment arm vector. If $\vec{r}$ and $\vec{F}$ has the same direction, $\vec{\tau}$ becomes zero and the image dose not rotate. For example, in Fig 5 the forces that are applied to the pixels p and q, make the image rotate; but the moment of the force that is applied to s is zero.

However, the sum of the moments produces the resultant moment; then the rotation angle is estimated using the resultant moment (Fig 5). So, the Eq. (18) is used to estimate the rotation angle of the image A.

$$\theta = \frac{\left|\sum_{p\in A}\vec{\tau}_p\right|}{J} \tag{18}$$

where τp is the moment of Fp (the resultant force that is applied to pixel p of the image A), θ is the estimated angle, and J is a scalar used to normalize the moment. J is equal to the number of the pixels of the image A.

After estimating the movement and the rotation, image B moves to a new position and again the forces are computed. This process will be repeated until the resultant force is equal to zero. However, the direction and the size of the movement depend on the resultant force. The flowchart of the proposed method is illustrated in Fig 6.

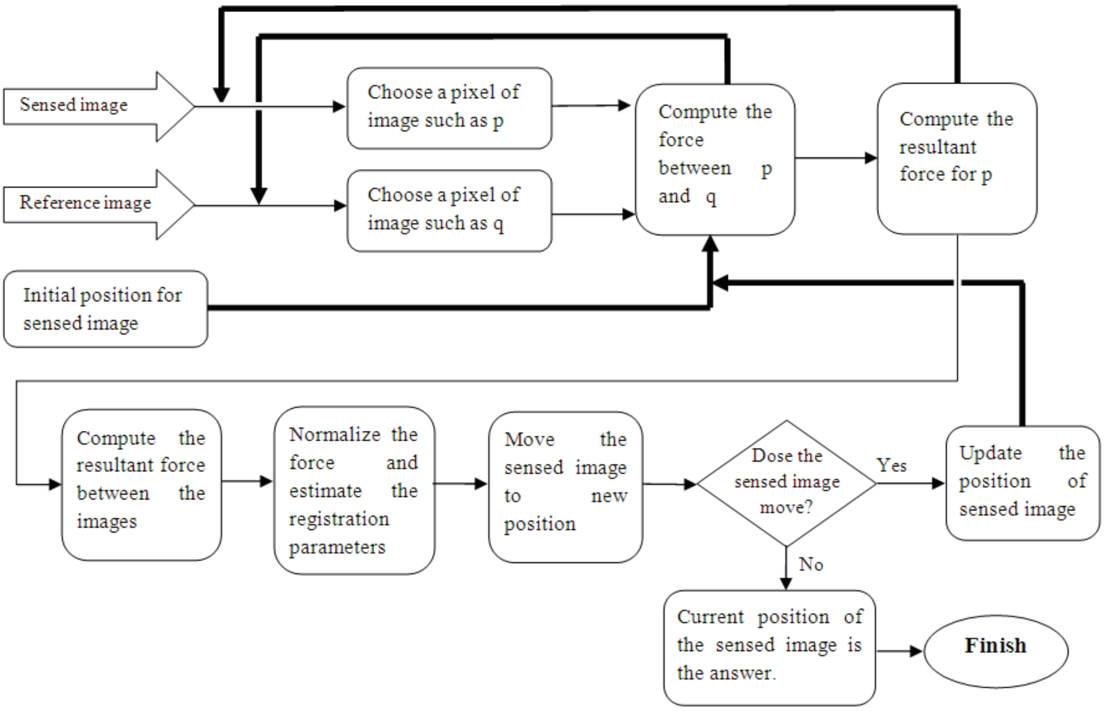

Sensed image
Reference image
Initial position for sensed image
Choose a pixel of image such as p
Choose a pixel of image such as q
Compute the force between p and q
Compute the resultant force for p
Compute the resultant force between the images
Normalize the force and estimate the registration parameters
Move the sensed image to new position
Dose the sensed image move?
Yes
No
Update the position of sensed image
Current position of the sensed image is the answer.
Finish

Fig 6. The flowchart of the algorithm

3. Simulation Results

Having explained the algorithm, the next step is to evaluate the performance of the proposed registration approach and examine the precision of the registration. In this regard, we try to have enough examples to show that the proposed algorithm works properly.

Therefore, the images of the Camera man and Lena are considered as the referenced images. The referenced and the sensed images are shown in Fig 7 and Fig 8. Then, the sensed images with different rotation angles from different parts of the referenced image are chosen. Both of the referenced images are 200×200 and the sensed images 60×100. It should be mentioned that in all of the following experiments, a_1, a_2, and a_3 were 0.04, 0.7, and 2.5 respectively, which they were obtained experimentally.

For each experiment first a reference, a sensed image, and an initial situation are chosen for the sensed image. Then the algorithm is run and the translation and the rotation of the sensed image are calculated in each step. In this case, if the translation is less than 3 pixels or the rotation is less than 3 degrees, the algorithm is terminated. Finally, the number of the steps and registration errors are reported. Tables 1-2 demonstrate twenty experiments that the sensed image, the referenced image, and the initial position of the sensed image are different in each experiment.

The first position of the sensed image is described as initial value. Moreover, x and y show the initial coordinates of the sensed image and θ is the degree that the sensed image should rotate to match its counterpart. The resultant force is calculated and the registration parameters update in each repetition of the algorithm. Consequently, the algorithm is repeated until the variation of the registration parameters becomes negligible (less than 3 pixels for x and y, and less than 3 degree for the rotation angle).

Tables 1-2 indicate that the proposed algorithm works properly. Clearly, the algorithm is converged in all of the twenty experiments and the average of the number of the iterations is 7.5. Despite of the fact that error of 3 pixels and

3 degrees is relatively high value, the proposed method can be quite useful. This is because obtaining the exact position from the approximate position is not so hard.

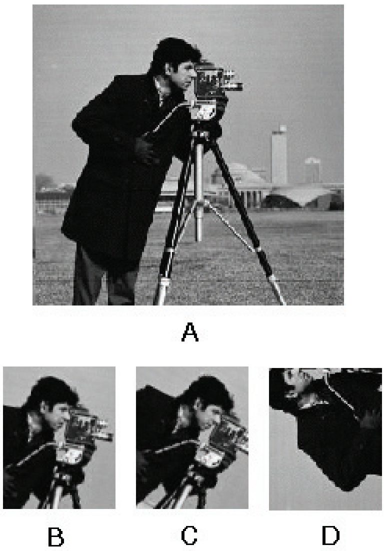


Fig 7. A: The Camera man (Reference Image) B-D: The sensed images

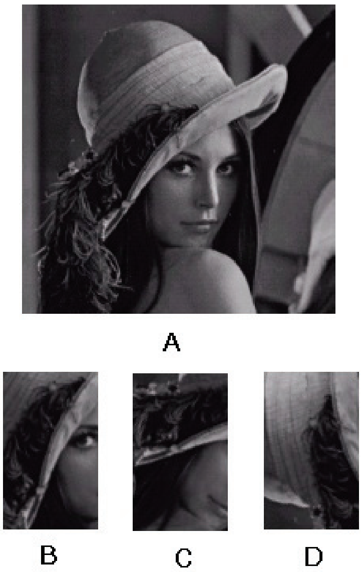


Fig 8. A: Lena (Reference Image) B-D: The sensed images

Some steps of an experiment are illustrated in Fig 9. In this Fig the referenced image and the position of the sensed image are shown after first, fourth and sixth iterations. In this case the rotation angle is 27 and the number of the iterations is 5. If we run the algorithm after the 5th iteration the sensed image does not move anymore.

4. Discussion of proposed method

4.1. Computation Reduction

In this section we try to confirm the proposed method to reduce the computation, and as a result, the algorithm becomes faster. The main idea is that it is not necessary to calculate the force between each pixel of the referenced and the sensed image. Therefore, some pixels are chosen instead of all.

Table 1. The simulation results for images in Fig 7 (Camera Man)

Sensed Image	Initial value			Number of iterations	Error		
	x	y	θ		ex (pixel)	ey (pixel)	eθ (degree)
B	73	50	11	4	2	0	0
B	123	120	11	12	1	0	0
B	150	80	11	3	1	1	0
B	163	105	11	7	1	2	2
C	163	105	27	20	1	1	1
C	123	70	27	7	0	1	0
C	27	50	27	7	0	1	0
D	50	90	90	10	1	3	0
D	42	120	90	9	1	0	0
D	100	60	90	11	2	0	2

Table 2. The simulation results for the images in Fig 8 (Lena)

sensed Image	Initial value			Number of iterations	Error		
	x	y	θ		ex pixel	ey pixel	eθ degree
B	84	73	-15	4	0	0	0
B	113	100	-15	5	1	0	0
B	123	130	-15	9	0	1	1
B	154	40	-15	7	0	1	1
C	124	129	35	8	0	2	0
C	76	85	35	5	0	0	3
C	88	113	35	4	0	0	0
D	81	83	-60	7	2	0	0
D	71	123	-60	8	0	1	1
D	86	130	-60	6	1	0	2

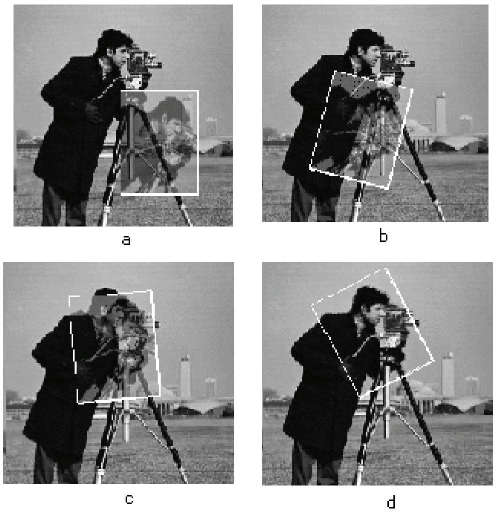


Fig 9. (a) The initial position of the images. The positions of the images after (b) one (c) four (d) six iterations.

This suggestion comes from the fact that the energy of the low frequency in an image is high and we have not so many changes in an image. As a result, adjacent pixels probably have the same color. So, first some pixels from the images randomly are chosen using uniform distribution, and then the forces between these pixels are calculated. Assuming that αA and αB are the percentage of the selected pixels of the reference and the sensed image, respectively, the algorithm becomes (αA*αB)-1 times faster. The reason is that the number of calculated forces reduce to (αA*αB) of its original value.

In addition, pixels which are far from each other apply negligible force; therefore, it is not necessary to calculate the force between these pixels. In this section, the force between two pixels is ignored providing that the distance between them is more than 40 pixels.

The performance of this method is reported for the different percentages of the selected pixel in tables 3 and 4.

The performance of the algorithm under different situations is evaluated. In case that the ratio of the selected pixels is about 8% for the sensed and 2% for the referenced images, the performance is acceptable. Therefore the computation is reduced and the algorithm is about 500 times faster. As it is seen in tables 3-4, the number of the iteration and the errors does not change under this situation so much.

Table 3. Errors of the different percentages of the selected pixels. The reference image is Cameraman and the sensed image is Fig 7.B.

Referenced Image (%)	Sensed Image (%)	Number of iterations	Error		
			ex (pixel)	ey (pixel)	eθ (degree)
20	100	4	1	0	0
10	40	4	1	0	0
5	25	5	1	1	0
3.5	20	5	1	1	0
2	10	5	0	1	1
2	8	5	0	1	1
1	4	6	2	1	1
0.5	2	7	2	2	1
0.2	1	∞	-	-	-

Table 4. Errors of the different percentages of the selected pixels. The reference image is Lena and the sensed image is Fig 8.D.

Referenced Image (%)	Sensed Image (%)	Number of iterations	Error		
			ex (pixel)	ey (pixel)	eθ (degree)
20	100	4	0	0	0
10	40	4	0	1	0
5	25	5	1	1	0
3.5	20	5	1	1	0
2	10	5	0	1	1
2	8	5	0	1	1
1	4	6	2	1	1
0.5	2	7	2	2	1
0.2	1	∞	-	-	-

4.2. The Order of the Algorithm

We begin this section by observing that the computational cost of the proposed algorithm stems mostly from the computation of the force between the two points. This case is the valuable computationally, because it is the most repeated process. However, the other steps in the algorithm have negligible cost.

Suppose that B (sensed image) is an x×y image and A (referenced image) is an m×n image. the force between q and some point in A should be calculated. Eq. (19) gives the total number of the computation.

$$n_{PA} \approx \alpha_A \times n \times m \times \alpha_B \times x \times y \times k \tag{19}$$

where αA and αB represent the ratio of the selected pixels for the reference and the sensed image, k is the number of iterations, and nPA is the number of the calculated forces between the two pixels. Therefore, nPA is proportional to the square of m and n.

Computing the order of the proposed algorithm, the next step is comparing it with another method such as the cross correlation and the mutual information methods. In order to use full search, the CC for each assumed geometric transformation of the image B is computed. Therefore, the similarity measure (CC or MI) is calculated for the window pairs from the sensed and the reference images and also its maximum is investigated through full search. Accordingly, the window pairs for which the maximum is achieved are set as the corresponding ones.

The number of x×y rectangles in an m×n rectangle is $(m-x+1)\times(n-y+1)$ [12]. If the angle between the sensed and the referenced image is not zero, the total number of the geometric transformation of the sensed image is $(m-x+1)\times(n-y+1)\times t_{bins}$. Using CC as similarity measure we have:

$$n_{CC} \approx (m-x+1)\times(n-y+1)\times x\times y\times t_{bins} \tag{20}$$

$$n_{CC} \approx m\times n\times x\times y\times t_{bins} \quad m << x, n << y \tag{21}$$

where tbins is the number of the angle bins and nCC shows approximately the number of the computations.

The cost of a single computation of the MI of two images depends on the number of pixels in images, x×y, and also on the number of the bins used to form the histogram. The computational cost relative to the number of histogram bins, nbins used in the computation, is (nbins)2. When used for image registration, the total cost is then a function of the number of steps where the MI is computed. This value is about (m-x+1)×(n-y+1)× tbins. [11]

$$n_{MI} \approx (x\times y+n_{bins}^2)\times(m-x+1)\times(n-y+1)\times t_{bins} \tag{22}$$

$$n_{MI} \approx (x\times y+n_{bins}^2)\times m\times n\times t_{bins} \quad m << x, n << y \tag{23}$$

The equations (19-21) prove that the computation cost of the proposed algorithm does not depend on tbins unlike the other two algorithms. Furthermore, αA*αB is insignificant (about 0.001) and k is approximately 8.

The comparison of the computation required by the CC, MI, and PA (the proposed algorithm) registration methods is given in table 5 (the assumption is that nbins=32, k=8 and tbins=180). The values are the number of the calculation such as summing and multiplying. As a result, the computation cost of proposed algorithm is acceptable. In the case that the computation cost of the proposed method is more than that of the fullsearch, the proposed method is useless.

Table 5. The number of the operations for the cross correlation (CC), the mutual information (MI), and the proposed algorithm (PA)

Size of Referenced Image	Size of Sensed Image	Number of Operations (×106) (Full Search)		
		CC	MI	PA
200×200	60×100	16120	20160	480
200×200	120×150	12960	14400	1880
100×150	70×70	2116	2980	380
100×200	30×50	2835	6615	240

4.3. Comparison of the Proposed Algorithm and the Steepest Decent

This section looks at the comparison of the algorithm with the two optimization methods, in terms of the computation and convergence.

When we search for the maximum value of the CC or MI, we can use the Steepest Decent instead of the full search. Regarding the search strategy, steepest decent has less computation than full search but the main drawbacks of it is the local maximum. Then, the Steepest Decent is used as the search strategy and the CC and MI as the similarity measures. As a result, the Steepest Decent is faster than our algorithm but the region of the convergence for the algorithm is quite larger. So, the following experiment evaluated the robustness of the algorithm concerning to the starting point.

The following procedures were followed for the purpose of this study: First, a reference and a sensed image were chosen. Then, for each pair of the images, 5 different initial positions were considered and in each case the algorithm was run. So, for each algorithm the computation time was reported. The approach was implemented on the Pentium Core2Due processor, 1 GB RAM, 1.8 GHz PC in a MATLAB environment.

In addition, the size of the step of the Steepest Decent algorithm was 3. In other words, the registration parameters increased and decreased by 3 units in each step, and the best answer was chosen by using the similarity measure.

Tables 6-7 show the simulation results for the Lena and Cameraman images. In these tables, Δθ, Δx, and Δy indicate the differences between the initial position and the correct position of the sensed images. The sign“-” in the tables shows that the algorithm did not converge.

It can be inferred from these two tables that when the sensed image and its counterpart are close to each other, the MI and CC get converge faster than the proposed algorithm. However, when the images are not close to each other the proposed method converges unlike the other two algorithms. Therefore, the advantage of the proposed method over the other methods is the large region of the convergence.

4.4. Convergence

The purpose of this section is to show; intuitively, that the convergence is guaranteed. In other words, after several iterations, the registration parameters are obtained independent of the initial position of the sensed image.

Fig 10 shows the CC graph of the images of Fig 8.B and Lena. As it is seen, there are several local maximums and the Steepest Decent is sensitive to these maximums.

Tables 8 show the simulation results for the Lena and Cameraman images. It is observed from Table 5 that the region of the convergence of the proposed algorithm is quite larger than the other two algorithms.

Table 6. The comparison of the algorithms for the image, Lena, and the images of Fig 8.

Sensed Image	Δx	Δy	Δθ	Computation time (second)		
				CC	MI	PA
Fig 8.B	5	5	15	1.9	1.3	2
Fig 8.B	12	15	15	1.2	-	2.1
Fig 8.B	20	70	15	-	-	4
Fig 8.B	35	15	15	-	-	1.1
Fig 8.B	20	20	15	1.6	2.7	2.5
Fig 8.D	10	27	25	-	-	3.9
Fig 8.D	10	12	25	-	3.7	4
Fig 8.D	5	10	25	1.7	3.2	3.2
Fig 8.D	15	35	25	-	-	3
Fig 8.D	25	35	25	-	-	3.2
Fig 8.C	10	20	35	-	-	5.4
Fig 8.C	10	5	35	4	-	4.8
Fig 8.C	5	5	35	-	-	4.2
Fig 8.C	5	0	35	-	5.0	4.4
Fig 8.C	5	15	35	-	-	5

Table 7. The comparison of the algorithms for the image, Camera man, and the images of Fig 7.

Sensed Image	Δx	Δy	Δθ	Computation time		
				CC	MI	PA
Fig 7.B	5	5	10	1.4	2.1	3.2
Fig 7.B	15	5	10	2.1	3.6	3.8
Fig 7.B	10	15	10	1.7	3	3.8
Fig 7.B	20	25	10	-	-	3.2
Fig 7.B	40	45	10	-	-	4.1
Fig 7.D	10	0	15	1.3	1.8	2.4
Fig 7.D	10	10	15	1.7	-	2.8
Fig 7.D	15	15	15	-	-	2.4
Fig 7.D	5	10	15	1.4	1.7	2.2
Fig 7.D	20	20	15	-	3.1	2.1
Fig 7.C	0	0	27	1.9	3.8	1.9
Fig 7.C	5	10	27	-	-	1.9
Fig 7.C	5	5	27	3.5	4.8	1.9
Fig 7.C	45	50	27	-	-	4.2
Fig 7.C	25	20	27	-	-	2

Fig 11 and 12 show quiver graphs. Every vector in the quiver graphs shows the movement vector when the smaller image situated in the place of the vector. In fact in order to obtain Fig 12 we run the proposed algorithm for 19×20 (the number of the vectors) times and each time the smaller image is situated in a different place. In Fig 12, the movement vectors are obtained from the proposed algorithm; on the contrary, in Fig 11 they are obtained from the Steepest Decent. As it is seen, there is a few local maximum for the proposed algorithm. So, in this example the convergence is guaranteed and it does not depend on the position of the smaller image. In other words, unlike the other two algorithms, it moves to the correct position independent of the initial position of the sensed image.

Table 8. Comparison of the region of convergence

Reference Image	Sensed Image	Region of Convergence (%)		
		CC	MI	PA
Lena	Fig 8.B	44	12	84
Lena	Fig 8.C	40	20	92
Lena	Fig 8.D	36	16	84
Cameraman	Fig 7.B	44	24	76
Cameraman	Fig 7.C	56	20	80
Cameraman	Fig 7.D	48	16	84

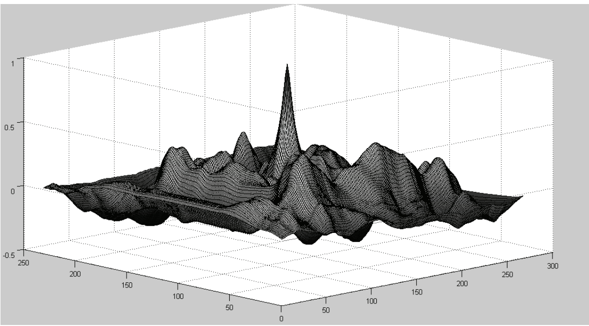
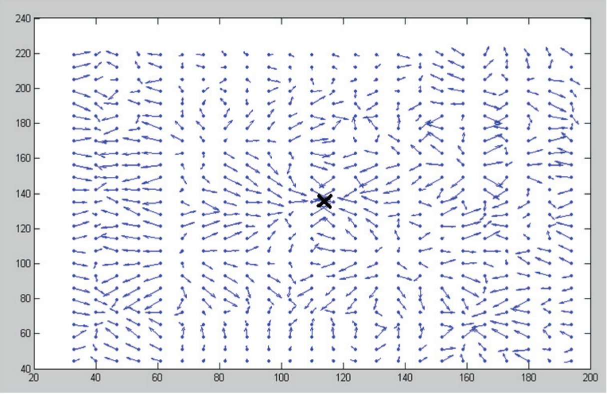
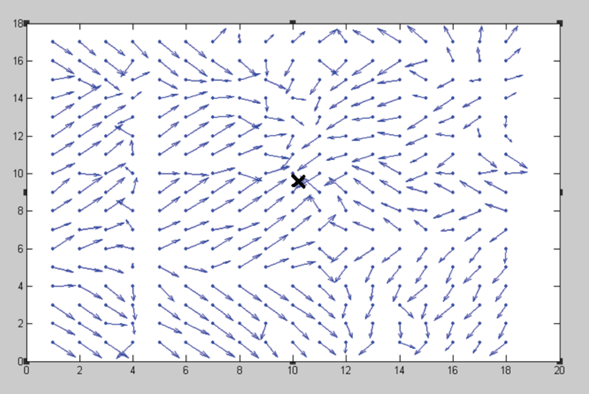


Fig 10. Cross-correlation graph for the sample Images



a



b

Fig 11. Quiver graph for (a) MI algorithm (b) CC algorithm.

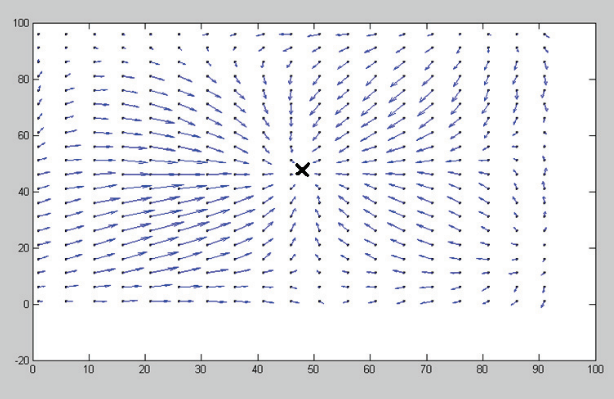


Fig 12. Quiver graph for proposed algorithm

5. Conclusion

This study presented an image registration algorithm based on the physical forces. The method used some of the mechanic principles as friction, mass, and resultant force to estimate the registration parameters.

We assumed these images as charged materials but with the opposite charges. So, they attract each other and the virtual forces helped to find the registration parameters. Consequently, the smaller image moved to match the bigger one. For better convergence of the algorithm we supposed that the friction increases when the images are close to matching the position. The experimental results show that this new approach is very effective in terms of speed and region of convergence.

REFERENCES

[1] Barbara Zitova´, Jan Flusser, “Image registration methods: a survey ,” Image and Vision Computing, Vol 21, pp.977-1000, 2003.

[2] L.M.G. Fonseca, B.S. Manjunath , “Registration techniques for multi-sensor remotely sensed imagery”, photo-grammetric Engineering and Remote Sensing 62 (1996) 1049–1056.

[3] W.K. Pratt, Digital Image Processing, Third Edition, Wiley, New York, 2001.

[4] A.wong, David A.Clausi, “ARRSI:Automated Registration of remote-sensing images”, IEEE Transactions on Geoscience and Remote Sensing , Vol. 45, No 5 ,MAY 2007.

[5] R.Berthilsson, “Affine correlation” , Proceedings of the International Conference on Pattern Recognition ICPR’98, pp. 1458–1461, Brisbane, Australia, 1998.

[6] R.N. Bracewell, The Fourier Transform and Its Applications, McGraw-Hill, New York, 1965.

[7] A. Abche, F.Yaacoub, A.Maalouf, E.Karam, “Image Registration based on Neural Network and Fourier Transform" , IEEE EMBS Annual International Conference, 2006.

[8] L.Lucchese, G. Doretto, G.M. Cortelazzo, “A frequency domain technique for range data registration”, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (2002) 1468–1484.

[9] P.Viola, W.M. Wells, “Alignment by maximization of mutual information”,International Journal of Computer Vision 24 (1997) 137–154.

[10] D. Cheng , Electromagnetic Field and Wave, 2nd Edn, pp. 348–350.

[11] A. Cole-Rhodes, L. Johnson, J. LeMoigne, “Multiresolution Registration of Remote Sensing Imagery by Optimization of Mutual Information Using a Stochastic Gradient”, IEEE Transactions on Image Processing , VOL. 12, NO. 12, DECEMBER 2003

[12] R. P. Grimaldi. Discrete and Combinatorial Mathematics: An Applied Introduction. Addison-Wesley, New York, 2nd edition, June 1989.

Authors:
Amin Sadri, Department of Electrical Engineering, Iran University of Science and Technology, a_sadri@elec.iust.ac.ir
Ali Asghar Beheshti Shirazi, Department of Electrical Engineering, Iran University of Science and Technology, abeheshti@iust.ac.ir
Masoomeh Zameni, Department of Computer Engineering, Iran University of Science and Technology, zameni_m@comp.iust.ac.ir