Firefly-Based Approaches of Image Recognition

Catalina-Lucia Cocianu, **Alexandru Daniel Stan** * and **Mihai Avramescu**

The Department of Economic Informatics and Cybernetics, Bucharest University of Economic Studies, 6 Piata Romana, 1st district, 010374 Bucharest, Romania; catalina.cocianu@ie.ase.ro (C.-L.C.); mihai.avramescu@ie.ase.ro (M.A.)

* Correspondence: stan1daniel10@stud.ase.ro or alexandru.stan1@yahoo.com

check for updates

Abstract: The main aim of the reported work is to solve the registration problem for recognition purposes. We introduce two new evolutionary algorithms (EA) consisting of population-based search methods, followed by or combined with a local search scheme. We used a variant of the Firefly algorithm to conduct the population-based search, while the local exploration was implemented by the Two-Membered Evolutionary Strategy (2M-ES). Both algorithms use fitness function based on mutual information (MI) to direct the exploration toward an appropriate candidate solution. A good similarity measure is the one that enables us to predict well, and with the symmetric MI we tie similarity between two objects A and B directly to how well A predicts B, and vice versa. Since the search landscape of normalized mutual information proved more amenable for evolutionary computation algorithms than simple MI, we use normalized mutual information (NMI) defined as symmetric uncertainty. The proposed algorithms are tested against the well-known Principal Axes Transformation technique (PAT), a standard evolutionary strategy and a version of the Firefly algorithm developed to align images. The accuracy and the efficiency of the proposed algorithms are experimentally confirmed by our tests, both methods being excellently fitted to registering images.

Keywords: image recognition; image registration; evolutionary strategy; Firefly algorithm; memetic algorithms; symmetric uncertainty

1. Introduction

Image registration is one of the well-known techniques belonging to the computer vision field [1–3]. In the last few years, nature-inspired algorithms and metaheuristics have been used to address the image registration problem, becoming a solid alternative to direct optimization methods. Even though image registration using the nature-inspired algorithms represents a niche in the research, a variety of evolutionary algorithm techniques has been proposed, starting from the most used, Genetic Algorithm, and moving to the newest, Coral Reefs Optimization [4–11].

The aim of the research work presented in this paper is to accurately register binary images using evolutionary search techniques. The considered perturbation is of rigid type. The proposed metaheuristics use symmetric uncertainty, designed to maximize the similarity between the computed variants of the sensed image and the target one. The aim is to develop a population-based evolutionary optimization model, in which the individuals evolve toward the chromosome corresponding to the given target image. Note that our approaches can register both gray-scale and colored images after a pre-processing step designed to compute the boundaries.

The rest of the paper is organized as follows. A brief review of the literature regarding the state of the art in the field of metaheuristics for image registration is provided in Section 2. In Section 3, the 2M-ES local search implemented for the rigid perturbation model is briefly outlined. The use of the Firefly algorithm—developed based on the updating of two rules introduced in [12]—to register

binary images is presented in Section 4. Note that the versions of the Firefly algorithm proposed in [12] proved extremely accurate, but very time consuming. In order to reduce the computational effort without decreasing the accuracy, we developed two new methods, provided in Sections 5 and 6.

A two-stage hybrid technique, that involves a population-based Firefly search and a variant of 2M-ES as the local search algorithm, is introduced in Section 5. The hybrid algorithms consist of two stages. In the first stage, a variant of Firefly technique [12] computes a "good" candidate solution, i.e., a chromosome whose fitness is larger than a certain threshold value. The obtained individual is considered as the initial solution of the 2M-ES method. Briefly, the Firefly search technique directs the exploration in an appropriate direction, then the 2M-ES algorithm is used to compute an optimal solution. In Section 6 of the paper, we introduce a memetic algorithm that embeds a version of the Firefly algorithm and 2M-ES.

The accuracy and efficiency of the proposed algorithms are experimentally confirmed by our tests outlined in Section 7. The conclusions of our study are provided in Section 8 of the paper.

2. Literature Review

Taking into account the main application of the methodology and the type of the optimization problem, various straightforward and hybrid methods have been presented in the literature.

In [11], the bio-inspired meta-heuristic Coral Reef Optimization Algorithm with Substrate Layers (CRO-SL) has been introduced. The algorithm is used to solve the real-coding image registration problem focusing on both mono-modal and inter-modal scenarios, the perturbation model being of 3D affine type. It has been experimentally established that CRO-LS is both robust and accurate.

The rigid transformation is studied in [13]. The authors proposed an evolutionary rigid body docking algorithm to register medical images, where the ligands are the sensed images and proteins are references. The aim is to minimize an energy function using genetic algorithms. The experiments have been conducted on images belonging to the Retrospective Image Registration Evaluation project, and the results are encouraging.

To enhance the feature selection in a static image-based facial expressions system, an evolutionary algorithm combined with standard linear discriminant analysis was introduced in [14]. The proposed method proved to be more efficient than convolutional neural network-based approaches, from the point of view of both training time and features size.

In order to solve the problem of deformable image registration, the Gene-pool Optimal Mixing Evolutionary Algorithm (GOMEA) has been applied by introducing a multi-objective real-valued adaptation [15]. The authors introduced a real-valued adaptation of GOMEA by using a prescribed dependency model to incrementally improve parts of solutions. The proposed method proved well-suited to register medical images as CT scans and MRI scans, being also vey efficient from the computational complexity point of view.

The problem of non-rigid multi-modal image registration is addressed in [16]. The authors proposed a novel optimization method, combining the limited memory Broyden–Fletcher–Goldfarb–Shanno with boundaries with cat swarm optimization. The algorithm uses an NMI fitness function to address the problem of the free-form deformations model. Extensive experiments proved that the method is quite accurate and tractable.

In [17], an improved artificial bee colony algorithm hybrid with a differential evolution for image registration has been proposed. The algorithm uses the simple MI measure to evaluate the candidate solutions. The reported results indicate accurate registration.

To solve the quality inspection for free-form surfaces, a design model-inspection method with range image registration has been proposed in [18]. A series of 3D discrete points were used to align images. The first stage of the method computes simplified cloud points, using the Hausdorff distance technique with a new point descriptor. Then, a differential algorithm based-optimizer is applied to evaluate the similarity between the designed model and the measurement model in a recursive manner. The experimental results showed that the model is efficient and effective.

Furthermore, numerous approaches for image reconstruction and modeling using metaheuristics, such as the Bacterial Foraging Optimization Algorithm (BFOA) [19], the Adaptive Invasion–based Mode and distributed Differential Algorithm (AIM-dDE) [20], the Histogram-based Fruit Fly Optimization Algorithm (HFFO) [21], Asynchronous Migration and a Mechanism Multi-Population Recombination (AsAMP) [22], have been reported in recent years. Some of the methodologies are based on single or multiple evolutionary algorithms as a hybrid method, use different codding schemes of solutions (real codding, binary codding, integer codding, etc.), use a parameter or matching approach, address the single or multiple objective optimization, and use different types of images (2D or 3D).

3. 2M-ES Algorithm for Image Alignment and Recognition

The Evolutionary Strategies (ES) class is one of the most popular self-adaptive metaheuristics designed to solve continuous parameter optimization problems [23]. The basic ES algorithm is 2M-ES, a simple local search procedure developed based on the Gaussian mutation operator.

Given a current candidate solution x, the algorithm computes a new vector by adding to each component of x Gaussian noise with zero mean and standard deviation σ. For each dimension, the step size σ is updated using the Rechenberg rule [24].

The 2M-ES algorithm can be implemented to align a certain sensed image I to the target T, as follows. Given the rigid model

$$I(x,y) = T\left(x^1, y^1\right) \tag{1}$$

$$\begin{pmatrix} x^1 \\ y^1 \end{pmatrix} = \begin{pmatrix} a \\ b \end{pmatrix} + s \cdot R \cdot \begin{pmatrix} x \\ y \end{pmatrix} \tag{2}$$

where $R = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$, the aim is to compute a vector (a, b, s, θ) and a function

$$f_{a,b,s,\theta}(x,y) = \frac{1}{s} \cdot R^T \cdot \left[\begin{pmatrix} x \\ y \end{pmatrix} - \begin{pmatrix} a \\ b \end{pmatrix} \right] \tag{3}$$

such that $I\left((f(x,y))^T\right) = T(x,y)$. The rigid transformation is characterized by the translation vector $\begin{pmatrix} a \\ b \end{pmatrix}$, the scale *s* and the rotation matrix R.

The genetic representation of a candidate solution is a four-size vector $c_{sol} = (a_c, b_c, s_c, \theta_c)$.

For each chromosome c, its fitness value reflects the similarity between and $I_c = I\left(f_{a_c,b_c,s_c,\theta_c}\right)$ in terms of the symmetric uncertainty NMI [25],

$$\text{fitness}(c) = \frac{2 \cdot MI(T, I_c)}{MI(T, T) + MI(I_c, I_c)} = \frac{2 \cdot MI(T, I_c)}{H(T) + H(I_c)} \tag{4}$$

$$MI(T, I_c) = H(T) + H(I_c) - H(T, I_c) \tag{5}$$

where $H(\cdot)$ is the Shannon entropy, and $H(\cdot, \cdot)$ is the joint entropy.

Obviously, the maximum value of the fitness function is 1 and it is reached when $I_c = T$.

Note that, since the search landscape of NMI proved more appropriate for evolutionary computation algorithms than simple MI, symmetric uncertainty is very well-suited to measure the similarity between two images.

The 2M-ES algorithm is described below (Algorithm 1). The inputs are: Max, the upper bound of the number of iterations; σ_{ini}, the initial value of σ-parameter; ϑ and ν, the Rechenberg rule parameters; the threshold $\tau \in (0, 1)$; the pair (I = sensed image, T = target image).

Algorithm 1. 2M-ES Algorithm for Image Registration

---

1. **Inputs:** $\sigma_{ini}, \vartheta, \nu, \tau, \text{Max}, \text{I}, \text{T}$
2. $t \leftarrow 0$
3. Randomly generate an initial point x
4. $x_c \leftarrow x; \sigma \leftarrow \sigma_{ini}; \text{it} \leftarrow 0; p \leftarrow 0$
5. **while** $t < \text{Max and fitness}(x_c) < \tau$ **do**
6. 　　Compute $y = x_c + z$, z randomly generated $N(0, \sigma)$
7. 　　**if** $\text{fitness}(x_c) < \text{fitness}(y)$
8. 　　　　$x_c \leftarrow y$
9. 　　　　$p \leftarrow p + 1$
10. 　　**end if**
11. 　　$\text{it} \leftarrow \text{it} + 1$
12. 　　**if** $\text{it} = \nu$
13. 　　　　Apply the Rechenberg rule $\sigma \leftarrow \begin{cases} \frac{\sigma}{\vartheta}, \frac{P}{\nu} > 0.2 \\ \sigma \cdot \vartheta, \frac{P}{\nu} < 0.2 \\ \sigma, \frac{P}{\nu} = 0.2 \end{cases}$
14. 　　　　$\text{it} \leftarrow 0; p \leftarrow 0;$
15. 　　**end if**
16. 　　$t \leftarrow t + 1$
17. **end while**
18. **Output:** I_{x_c}

---

4. Image Registration Using the Firefly Algorithm

The Firefly algorithm (FA) is a swarm intelligence algorithm inspired by the flashing patterns of fireflies and the phenomenon of bioluminescent communication, successfully used to solve various parameter-optimization problems [26–28]. It was developed based on three rules, briefly described in the following. First, it is assumed that all individuals are unisex, each firefly being attracted to all other fireflies. Secondly, the attractiveness associated to a firefly refers to its brightness. Finally, it is assumed that the attractiveness between fireflies is directly influenced by the brightness and the distance. In terms of metaheuristics, the light intensity is proportional to the value of the objective function, and it represents the fitness value of the considered individual.

The attractiveness of a certain firefly denoted by j seen by another firefly i is expressed by

$$\beta_j(r) = \beta_j(0) \cdot e^{-\gamma r^2} \tag{6}$$

where x_j is the current position of j, x_i is the current position of i, $r = x_j - x_i$ is the Euclidian distance between j and i, $\beta_j(0)$ is the brightness of j at $r = 0$, and γ represents the light absorption coefficient. Note that, in the standard Firefly algorithm, $\beta_j(0)$ does not depend on the individual j. We denote by β_0 the attractiveness at the distance 0 [26].

Each firefly i is attracted by a brighter firefly j, updating its corresponding position as follows,

$$x_i(t+1) = x_i(t) + \beta_0 \cdot e^{-\gamma r_{ij}^2} \left(x_j(t) - x_i(t) \right) + \alpha \cdot \varepsilon \tag{7}$$

where $\beta_0 \cdot e^{-\gamma r_{ij}^2} \left(x_j(t) - x_i(t) \right)$ is the attraction to the light intensity, α controls the randomness and ε is randomly drawn from $U(0, 1)$.

In the standard FA, the firefly *i* changes its position according to (7) if the attractiveness produced by $x_i(t+1)$ is higher than the attractiveness of the old position $x_i(t)$. Usually, the termination criterion of the FA is given by the iterations number. In a case where the optimal value of the brightness function is known, the FA search is also over when the best individual is good enough, i.e., its fitness is close to MAXF.

The population diversity in the Firefly algorithm is essentially influenced by the randomization term $\alpha \cdot \varepsilon$ in (7). In most cases, the parameter α is static or decreases linearly in time, each firefly from a certain generation having the same randomness degree. The value of the parameter α should rapidly decrease through the first generations, to explore new search space. During the final iterations, in order to maintain the direction of search, the parameter should slightly vary.

In the following, we briefly describe the variant of the Firefly algorithm used for solving the binary image registration task introduced in [12].

Each firefly is encoded as a D-dimensional real-valued sequence, representing the transformation parameters (a, b, s, θ). In our work $D = 4$, an individual i being defined in terms of its position by $x_i = (a_i, b_i, s_i, \theta_i)$, corresponding to the degradation model (2). The brightness, $\text{fitness}(x_i)$, of the firefly, identified by $x_i = (a_i, b_i, s_i, \theta_i)$, is computed in terms of (4).

We considered the fixed-size model, where each population has n fireflies, $X = \{x_1, x_2, x_3, \ldots x_n\}$,

$$x_i = (x_i(1), x_i(2), \ldots, x_i(D)) \tag{8}$$

$$x_i(k) \in [lb(k), hb(k)], k = 1, \ldots, D \tag{9}$$

Consequently, the search space is given by

$$S = \prod_{k=1}^{D} [lb(k), hb(k)] \tag{10}$$

The initial population, X_0, is randomly generated according to

$$x_i(k) = (hb(k) - lb(k)) \cdot d + lb(k) \tag{11}$$

where d is a draw from uniform distribution $U(0, 1)$.

We denote by c a constant scale factor, and let ε be a number drawn from $U(0, 1)$. The updating rule introduced in [12] is defined based on the fitness of the attractor and the variable ranges, as follows,

$$x_i(k) = x_i(k) + \beta_{ij}(k) \cdot \left(x_j(k) - x_i(k)\right) + \frac{hb(k) - lb(k)}{\max_k (hb(k) - lb(k))} \cdot \varepsilon \cdot c \cdot \exp\left(1 - \text{fitness}(x_j)\right) \tag{12}$$

$$\alpha_2 = \frac{hb(k) - lb(k)}{\max_k (hb(k) - lb(k))} \cdot \exp\left(1 - \text{fitness}(x_j)\right) \cdot c \tag{13}$$

To deal with unfeasible candidate solutions, the following border reflection mechanism is installed

$$\text{If } x_i(k) > hb(k), \text{ then } x_i(k) = U(val, hb(k))$$

Else

$$\text{If } x_i(k) < lb(k), \text{ then } x_i(k) = U(lb(k), val)$$

where

$$val = c_1 \cdot lb(k) + (1 - c_1) \cdot hb(k) \tag{14}$$

$c_1 \in (0, 1)$ and $U(val, hb(k))$ represents a draw from uniform distribution. Taking into account the results reported in [12], the following proposed approaches that use FA are developed based on the updating rule (12).

5. Two-Stage Hybrid Algorithm for Image Recognition

In order to obtain improved image recognition algorithms, we present a two-stage hybrid algorithm that combines the Firefly-based algorithm described in Section 3 with the standard 2M-ES

local search method. The algorithm first uses a variant of the Firefly algorithm to obtain a promising candidate solution, *best*, then applies the local search algorithm, 2M-ES, to reach an optimal solution, $best_c$.

Since the fine tune of solutions obtained by the heuristic population-based methods usually implies high execution times, the main purpose is to obtain a tractable recognition method. More efficient methods, incorporating systematic searches of the neighborhood of good solutions, can be derived by adding a local search mechanism [24].

Our hybrid approach uses the variant of FA described in Section 3 to compute a promising candidate solution that is going to be improved via 2M-ES. Note that a firefly *i* is defined in terms of its position by $x_i = (a_i, b_i, s_i, \theta_i)$. The position x_i corresponds to the rigid degradation model (2). The brightness of the firefly identified by $x_i = (a_i, b_i, s_i, \theta_i)$, $fitness(x_i)$, is computed in terms of (4). The FA component of the hybrid technique is defined by the following parameters: n represents the population size, NMax represents the maximum number of generations, γ represents the light absorption, β_0 is the attractiveness at the distance 0, τ represents the desired quality ($\tau \leq 1$), I is the sensed image and T is the target image. The proposed two-stage hybrid technique is provided below (Algorithm 2).

Algorithm 2. Two-Stage Hybrid Firefly Algorithm

---

1. **Inputs**: $n, NMax, \beta_0, \gamma, \tau, \sigma1_{ini}, \vartheta, \nu, \tau1, Max, I, T$
2. **APPLY FA**
3. $t \leftarrow 0$
4. Randomly generate an initial population $P_t = \{x_1, x_2, x_3, \ldots x_n\}$
5. Compute $fitness(x_i)$, the light intensity of each $x_i \in P_t$
6. Compute best: $fitness(best) = \max_{x \in P_t} fitness(x)$
7. **while** $t < NMax$ and $fitness(best) < \tau$ **do**
8.   **for** $i = 1 \ldots n$
9.   **for** $j = 1 \ldots n$
10.   **if** $fitness(x_i) < fitness(x_j)$
11.   Compute x_{new} by moving firefly x_i toward firefly x_j (12)
12.   For x_{new} use the border reflection mechanism (14)
13.   Compute the brightness of x_{new}, $fitness(x_{new})$
14.   Replace x_i with x_{new} in P_t
15.   **end if**
16.   **end for**
17.   **end for**
18.   Compute $best_c$: $fitness(best_c) = \max_{x \in P_t} fitness(x)$
19.   **if** $fitness(best_c) > fitness(best)$
20. $best = best_c$
21.   **end if**
22. $t \leftarrow t + 1$
23. **end while**
24. **APPLY 2M-ES PROCEDURE**
25. Compute $best_c$: $2M\text{-}ES(best, \sigma1_{ini}, \vartheta, \nu, \tau1, Max, I, T)$
26. **Output**: I_{best_c}

---

6. The Memetic Approaches of Image Registration

The memetic algorithms (MA) are optimization methods in which the evolutionary process is enhanced with deterministic, heuristics or other local search techniques, which reduce the probability of premature convergence. In the memetic approaches, during the evolution process, the information transmission is improved by incorporating local, exact or heuristic methods [29].

In order to reduce the worst-case run times, one has to use a local search method with an updating rule that differs from population-based search techniques variation operator. From the intuitive point of view, in MA variation operators, mutation in particular, generate points lying in various basins of attractions with respect to the local point operator. The diversification is done either using large mutation rates or, even better, by applying mutation operators with a different neighborhood structure.

The standard MA scheme is described below (Algorithm 3) [24].

Algorithm 3. Generic Memetic Algorithm

---

1. Initialize population
2. Evaluate each candidate
3. **while** Not Termination Condition **do**
4. 　Select parents
5. 　Recombine parents to obtain offspring
6. 　Mutate offspring
7. 　Evaluate offspring
8. 　Improve offspring via local search
9. 　Select the next generation
10. **end while**

Usually, the initial population can be generated using a mixed variant of randomly drawn individuals and a local search procedure. This way, the initial population contains not only randomly generated individuals, but also some sub-optimal solutions. Further, procedures underlying mass mutation and selective initialization could be applied to initialize a population. [24,29,30].

The proposed MA combines the variants of the Firefly algorithm described in Section 3 with the 2M-ES local search. Consequently, the neighborhood characteristics of the updating rules depend on the particular local topology, and also on the considered algorithm that operates at a certain moment of the evolution time. The resulting algorithm uses different variation operators, with various neighborhood structures.

We developed the memetic approach, taking into consideration that the number of parents is significantly smaller that the offspring population size. If a variant of the Firefly algorithm with μ individuals is used, the number of updated positions of fireflies could be around μ^2. Therefore, due to excessive computational effort, the local search procedure cannot be applied to each child/updated individual. Moreover, in order to avoid the premature convergence and to preserve the diversity of the population, the local search is applied on a small subset of children [24]. We also used the local search optimization to improve the current best firefly solution, if and only if it is less bright than the global best solution. We applied the 2M-ES search procedure for λ_s offspring randomly selected, where $\lambda_s \ll \lambda$. In our work we considered $\lambda_s = \frac{\mu}{s}, s > 2$.

The memetic algorithm, based on the proposed variant of FA and the 2M-ES local search, is described as follows (Algorithm 4). The parameters of the resulting algorithm are similar to those used to present Algorithm 2.

Algorithm 4. Memetic Firefly Algorithm

---

1. **Inputs:** $n, NMax, \beta_0, \gamma, \tau, s, \sigma1_{ini}, \vartheta, \nu, \tau1, Max, I, T$
2. $t \leftarrow 0$
3. Randomly generate an initial population $P_t = \{x_1, x_2, x_3, \ldots x_n\}$
4. **for** $i = 1 \ldots n/s$
5. 　　Randomly select $x \in P_t$
6. 　　Compute x_{new}: $2M - ES(x, \sigma1_{ini}, \vartheta, \nu, \tau1, Max, I, T)$
7. 　　Replace x with x_{new} in P_t
8. **end for**
9. Evaluate each $x_i \in P_t$
10. Compute best : $\text{fitness(best)} = \max_{x \in P_t} \text{fitness}(x)$
11. **while** $t < NMax$ and $\text{fitness(best)} < \tau$ **do**
12. 　　**for** $i = 1 \ldots n$
13. 　　　　for $j = 1 \ldots n$
14. 　　　　　**if** $\text{fitness}(x_i) < \text{fitness}(x_j)$
15. 　　　　　　Compute x_{new} by moving firefly x_i toward firefly x_j using (12)
16. 　　　　　　For x_{new} use the border reflection mechanism (14)
17. 　　　　　　Evaluate x_{new}
18. 　　　　　　Replace x_i with x_{new} in P_t
19. 　　　　　**end if**
20. 　　　　**end for**
21. 　　**end for**
22. 　　Compute best_c: $\text{fitness(best}_c) = \max_{x \in P_t} \text{fitness}(x)$
23. 　　**if** $\text{fitness(best}_c) \le \text{fitness(best)}$
24. 　　　**for** $i = 1 \ldots n/s$
25. 　　　　Randomly select $x \in P_t$
26. 　　　　Compute x_{new}: $2M - ES(x, \sigma1_{ini}, \vartheta, \nu, \tau1, Max, I, T)$
27. 　　　　Replace x with x_{new} in P_t
28. 　　　**end for**
29. 　　　Compute best_c: $\text{fitness(best}_c) = \max_{x \in P_t} \text{fitness}(x)$
30. 　　　**if** $\text{fitness(best}_c) > \text{fitness(best)}$
31. 　　　　$\text{best} = \text{best}_c$
32. 　　　**end if**
33. 　　**end if**
34. 　　$t \leftarrow t + 1$
35. **end while**
36. **Output:** I_{best}

---

7. Experimental Results

In order to experimentally evaluate the accuracy and the efficiency of the proposed algorithms, a series of experiments were performed on different binary images representing signatures. The tests were conducted on simulated data. We used 16 pairs (sensed image, target image), with each observed image representing a certain rigid perturbation of the original one. The proposed registration methods have been applied 500 times for each pair of images, in order to derive significant conclusions. Note that we used the perturbation model (2) with various translations, scale factors and rotation matrices.

Further, we have conducted a comparative analysis to evaluate the performance of the proposed algorithms against the well-known Principal Axes Transformation (PAT) registration method, a standard ES population-based approach [31] and a variant of the Firefly algorithm reported in [12].

Obviously, one of the most challenging tasks was to derive tractable algorithms, given the complexity of the fitness function and the fact that the searching space is of continuous type.

The chromosome evaluation was performed using only contour pixels, the procedure being outlined in [31].

We evaluated the accuracy of the evolutionary-based algorithms introduced in this paper through the success rate and the Signal-to-Noise-Ratio (SNR) measure. The success rate of an algorithm A, SR(A), is given by [31].

$$SR(A) = \frac{NSuccess(A)}{NRun(A)} \cdot 100\% \tag{15}$$

where NRun(A) is the number of algorithm executions and NSuccess(A) stands for the number of successful runs. A successful run is defined by a solution whose quality is above a certain threshold value. In our work, the threshold was set to 0.89. For each pair of images (S,T), the SNR value is given by:

$$SNR(T,S) = 10 * \log_{10}\left[\frac{\sum_{x=1}^{M}\sum_{y=1}^{N}(S(x,y))^2}{\sum_{x=1}^{M}\sum_{y=1}^{N}(T(x,y)-S(x,y))^2}\right] \tag{16}$$

where (M,N) is the image size.

For each pair of images, we recorded the mean value, the maximum value and the minimum value of the best fitness, the SNR measure and the run time value, respectively. In this study, we express the complexity of a certain algorithm using the mean value of the execution time recorded for each test, no matter if it was a successful run or not. The tests were conducted on the following configuration: Processor—Intel Core I7-7700k 3.6 GHZ, Memory—8 GB DDR4 2400 MHZ, Storage—1TB HDD 7200 RPM SATA 3.

Note that, in order to come to meaningful conclusions concerning the accuracy of the proposed methods, we split the test data into two classes of images, TI1 and TI2. TI1 consists of images correctly recognize by a standard ES algorithm [31], while the rest of the images belong to TI2.

The results of the standard ES algorithm presented in [31] are provided in Tables 1 and 2.

Table 1. The Efficiency of the Standard ES-Based Method.

Input	SR	Run Time Mean Value	Run Time Min Value	Run Time Max Value
TI1	100%	15.29	4.30	34.34
TI2	98%	26.21	5.54	95.85

Table 2. The Accuracy of the Standard ES-Based Method.

Input	SR	FITNESS SNR Mean Value	FITNESS SNR Min Value	FITNESS SNR Max Value
TI1	100%	0.92 26.44	0.89 25.02	0.969 31.16
TI2	98%	0.903 26.21	0.054 10.034	0.927 27.587

In Tables 3 and 4 are displayed the results of the Firefly algorithm version reported in [12]:

Table 3. The Efficiency of the Firefly-Based Method.

Input	SR	Run Time Mean Value	Run Time Min Value	Run Time Max Value
TI1	100%	14.5914	1.7948	25
TI2	100%	16.7172	2.1061	41.7394

Table 4. The Accuracy of the Firefly-Based Method.

Input	SR	FITNESS SNR Mean Value	FITNESS SNR Min Value	FITNESS SNR Max Value
TI1	100%	0.9063 26.2854	0.89 24.6903	0.969 31.1685
TI2	100%	0.9055 26.1922	0.89 21.9707	0.963 30.8899

The run times obtained when the Two-Stage Hybrid Firefly algorithm was applied to each pair (sensed image, target image) belonging to TI1 and TI2, respectively, are reported in Tables 5 and 6. Note that TI1 $=\{P_{1,1}, P_{1,2}, \ldots, P_{1,8}\}$ and TI2 $=\{P_{2,1}, P_{2,2}, \ldots, P_{2,8}\}$. The values of the accuracy measures are presented in Tables 7 and 8.

We used the following parameter settings. The population size in the first stage is n $=$ 20. The attractiveness at the distance 0 is $\beta_0 = 1$, while the light absorption coefficient is $\gamma = 1$. The termination criteria parameters are $\tau = 0.7$ and NMax $=$ 200. In the second stage, the parameters are as follows: $\nu = 0.87$, the initial dispersions vector is $\sigma 1_{ini} = [1, 1, 0.01, 0.01]$, $\vartheta = 50$, Max $=$ 3000 and $\tau 1 = 0.89$.

For each class of images, we summarized the performances of the Two-Stage Firefly 2M-ES Hybrid Algorithm in Tables 9 and 10. Note that the accuracy of this algorithm is experimentally proved to be similar to that corresponding to the Firefly algorithm [12], and better than the accuracy recorded for the standard ES-based method [31]. Moreover, the efficiency measured in run times is significantly improved by this method. In the case of the first class of images, TI1, the run times are significantly decreased, from 14.59 s and 15.29 s, respectively, to 5.27 s, while the computation effort carried out by the proposed hybridization algorithm in the case of images belonging to TI2 decreased from 16.71 s and 26.21 s, respectively, to 6.63 s.

Table 5. The Efficiency of the Two-Stage Firefly 2M-ES Hybrid Algorithm for TI1.

Input	SR	Run Time Mean Value	Run Time Min Value	Run Time Max Value
$P_{1,1}$	100%	4.3325	0.4764	10.2576
$P_{1,2}$	100%	5.1893	0.625	11.0681
$P_{1,3}$	100%	3.3141	0.4046	8.2807
$P_{1,4}$	100%	5.8114	0.734	13.2551
$P_{1,5}$	100%	5.7321	0.9164	12.7348
$P_{1,6}$	100%	6.5153	0.864	14.0022
$P_{1,7}$	100%	5.6918	1.0151	12.5463
$P_{1,8}$	100%	5.6389	0.5895	11.9978

Table 6. The Efficiency of the Two-Stage Firefly 2M-ES Hybrid Algorithm for TI2.

Input	SR	Run Time Mean Value	Run Time Min Value	Run Time Max Value
$P_{2,1}$	100%	8.5842	0.8736	17.5454
$P_{2,2}$	99.8%	7.5911	1.0809	15.0537
$P_{2,3}$	100%	5.4495	0.6411	11.4637
$P_{2,4}$	99.8%	4.1837	0.7385	30.6742
$P_{2,5}$	100%	6.2053	1.3296	11.9818
$P_{2,6}$	100%	4.9761	0.7192	11.4633
$P_{2,7}$	100%	4.7902	0.8598	11.3452
$P_{2,8}$	100%	11.3124	1.5711	24.5833

Table 7. The Efficiency of the Two-Stage Firefly 2M-ES Hybrid Algorithm for TI1.

Input	SR	FITNESS SNR Mean Value	FITNESS SNR Min Value	FITNESS SNR Max Value
$P_{1,1}$	100%	0.9113 25.931	0.89	0.9665
$P_{1,2}$	100%	0.9109 26.2857	0.89	0.9687
$P_{1,3}$	100%	0.904 27.5375	0.8901	0.9304
$P_{1,4}$	100%	0.9039 25.8263	0.89	0.9293
$P_{1,5}$	100%	0.9045 25.8614	0.89	0.9316
$P_{1,6}$	100%	0.9044 26.1817	0.89	0.9366
$P_{1,7}$	100%	0.9052 25.6632	0.89	0.9365
$P_{1,8}$	100%	0.9013 26.5873	0.89	0.9271

Table 8. The Efficiency of the Two-Stage Firefly 2M-ES Hybrid Algorithm for TI2.

Input	SR	FITNESS SNR Mean Value	FITNESS SNR Min Value	FITNESS SNR Max Value
$P_{2,1}$	100%	0.9133 26.5621	0.89	0.9669
$P_{2,2}$	99.8%	0.9021 26.06244	0.7673	0.9289
$P_{2,3}$	100%	0.9036 26.22287	0.89	0.9313
$P_{2,4}$	99.8%	0.9048 27.23462	0.1789	0.943
$P_{2,5}$	100%	0.9031 26.26862	0.89	0.929
$P_{2,6}$	100%	0.9051 26.97139	0.8901	0.9338
$P_{2,7}$	100%	0.9043 27.28921	0.8901	0.9441
$P_{2,8}$	100%	0.9011 22.55602	0.89	0.9236

Table 9. The Efficiency of the Two-Stage Firefly 2M-ES Hybrid Algorithm.

Input	SR	Run Time Mean Value	Run Time Min Value	Run Time Max Value
TI1	100%	5.278	0.404	14.002
TI2	99.95%	6.636	0.6411	30.674

Table 10. The Accuracy of the Two-Stage Firefly 2M-ES Hybrid Algorithm.

Input	SR	FITNESS SNR Mean Value	FITNESS SNR Min Value	FITNESS SNR Max Value
TI1	100%	0.9057 26.23431	0.89 24.444	96.8796 31.523
TI2	99.95%	0.90472 26.1459	0.1789 13.6355	96.697 31.042

We implemented the proposed Memetic Firefly algorithm with the following parameter values. The population size is n $=$ 20. The attractiveness at the distance 0 is $\beta_0 = 1$, while the light absorption coefficient is $\gamma = 1$. The termination criteria parameters are $\tau = 0.89$ and NMax $=$ 200. The initial population is randomly generated, and 30% of its individuals are improved via 2M-ES. During the evolution process, the local search is used only when the best new individual, represented by the position of an updated firefly, is below the quality of the current best candidate solution. In such a case, we randomly optimize 20% of the new individuals by applying the 2M-ES procedure. The values of parameters in the 2M-ES procedure are set as follows: $\nu = 0.87$, Max $=$ 300, $\sigma_{ini} = [1.0, 1.0, 0.01, 0.01]$. The mutation size is updated every 40 iterations.

The run times of the Firefly 2M-ES Memetic Algorithm are shown in Tables 11 and 12, while the recorded values of the fitness function and SNR are displayed in Tables 13 and 14. To compare the obtained results against the standard ES-based method [31] and the Firefly algorithm in [12], we summarized the runtime and accuracy values in Tables 15 and 16. Note that the accuracy of this algorithm is almost 100% for both classes of images, while the computation effort is significantly decreased as compared to both the standard ES-based method and the considered version of the Firefly algorithm.

The results obtained when the PAT method was applied are less accurate, the mean value of the NMI defined by (4) being below 0.8 in most of the cases. Further, the mean value of SNR is around 23, lower than the values corresponding to the proposed methods.

Table 11. The Performances of the Firefly 2M-ES Memetic Algorithm for TI1.

Input	SR	Run Time Mean Value	Run Time Min Value	Run Time Max Value
$P_{1,1}$	100%	7.2073	3.0007	19.1900
$P_{1,2}$	100%	7.7114	2.9078	24.4584
$P_{1,3}$	100%	6.0070	2.1924	19.0050
$P_{1,4}$	100%	8.1369	2.9153	27.0790
$P_{1,5}$	100%	8.3016	3.1708	25.9697
$P_{1,6}$	100%	15.9387	2.4286	59.3580
$P_{1,7}$	100%	9.9297	2.7894	34.0189
$P_{1,8}$	100%	10.7730	2.9607	33.6675

Table 12. The Performances of the Firefly 2M-ES Memetic Algorithm for TI2.

Input	SR	Run Time Mean Value	Run Time Min Value	Run Time Max Value
$P_{2,1}$	100%	10.2538	2.7983	41.112
$P_{2,2}$	100%	20.0726	2.9867	68.346
$P_{2,3}$	100%	10.316	2.6688	32.1672
$P_{2,4}$	99.8%	8.5464	2.6749	239.634
$P_{2,5}$	100%	9.9095	2.9316	34.5194
$P_{2,6}$	100%	8.4755	2.8107	34.1838
$P_{2,7}$	100%	12.579	2.5451	40.4873
$P_{2,8}$	100%	12.4921	4.1513	43.4075

Table 13. The Accuracy of the Firefly 2M-ES Memetic Algorithm for TI1.

Input	SR	FITNESS SNR Mean Value	FITNESS SNR Min Value	FITNESS SNR Max Value
$P_{1,1}$	100%	0.9114 25.93298	0.89	0.9631
$P_{1,2}$	100%	0.9125 26.395	0.8902	0.968
$P_{1,3}$	100%	0.9042 27.55638	0.8901	0.9291
$P_{1,4}$	100%	0.904 25.838	0.89	0.9332
$P_{1,5}$	100%	0.9035 25.813	0.89	0.9318
$P_{1,6}$	100%	0.9032 26.12029	0.89	0.9324
$P_{1,7}$	100%	0.9051 25.67207	0.89	0.9395
$P_{1,8}$	100%	0.9012 26.58445	0.89	0.9249

Table 14. The Accuracy of the Firefly 2M-ES Memetic Algorithm for TI1.

Input	SR	FITNESS SNR Mean Value	FITNESS SNR Min Value	FITNESS SNR Max Value
$P_{2,1}$	100%	0.9122 26.48436	0.89	0.97
$P_{2,2}$	100%	0.9015 26.0435	0.8902	0.9264
$P_{2,3}$	100%	0.9039 26.25327	0.89	0.9296
$P_{2,4}$	99.8%	0.9055 27.20916	0.8809	0.9455
$P_{2,5}$	100%	0.903 26.25712	0.89	0.9296
$P_{2,6}$	100%	0.9062 27.02602	0.89	0.9354
$P_{2,7}$	100%	0.9048 27.31797	0.89	0.9426
$P_{2,8}$	100%	0.9 22.54077	0.89	0.9225

Table 15. The Performances of the Firefly 2M-ES Memetic Algorithm.

Input	SR	Run Time Mean Value	Run Time Min Value	Run Time Max Value
TI1	100%	9.2507	2.1924	59.358
TI2	99.9975%	11.5806	2.5451	239.634

Table 16. The Accuracy of the Firefly 2M-ES Memetic Algorithm.

Input	SR	FITNESS SNR Mean Value	FITNESS SNR Min Value	FITNESS SNR Max Value
TI1	100%	0.90569 26.2390	0.89 24.513	0.9680 31.3535
TI2	99.9975%	0.90474 26.14152	0.8809 21.87044	0.97001 31.3641

8. Conclusions

The main aim of the reported work is to solve the registration problem for recognition purposes. We proposed two new methods, a two-stage hybrid algorithm and a memetic technique, respectively consisting of the Firefly population-based search method followed by or combined with a local search scheme.

We investigated the potential of using a version of the Firefly algorithm to conduct the population-based search, while the local exploration was performed using the Two-Membered Evolutionary Strategy. The proposed strategies use the mutual information similarity measure, defined in terms of symmetric uncertainty, to evaluate the quality of the candidate solutions. The fitness function computation scheme is designed to obtain tractable algorithms.

A long series of tests, designed to establish the performances of the proposed methods versus other commonly used classes of techniques for image registration in cases of rigid perturbation model, have been conducted. It was experimentally proved that the proposed methods outperform PAT registration, from both accuracy and efficiency points of view. Our tests also pointed out that the introduced techniques are very accurate, and significantly faster than both the ES-based optimization method reported in [31] and the version of the Firefly algorithm [12].

The results indicated that the new methods are excellently fitted to register binary images, their corresponding accuracy being almost 100%. Further, the obtained algorithms are very efficient in terms of the recorded run times.

The experimentally established conclusions are very promising, entailing future research toward more general perturbation models, not necessarily corresponding to registration problems. Moreover, the registration process in cases of more complex images could be designed using similar approaches.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Goshtasby, A.A. *Image Registration: Principles, Tools and Methods*; Springer Science & Business Media: London, UK, 2012. [CrossRef]
2. Modersitzki, J. *Numerical Methods for Image Registration*; Oxford University Press: New York, NY, USA, 2004. [CrossRef]
3. Zitova, B.; Flusser, J. Image registration methods: A survey. *Image Vis. Comput.* **2003**, *21*, 977–1000. [CrossRef]
4. Vemuri, B.; Chen, Y. Joint Image Registration and Segmentation. In *Geometric Level Set Methods in Imaging, Vision, and Graphics*; Springer: New York, NY, USA, 2003.
5. Sarvamangala, D.R.; Kulkarni, R.V. A Comparative Study of Bio-inspired Algorithms for Medical Image Registration. *Adv. Intell. Comput.* **2018**, *44*, 27–44. [CrossRef]
6. Abdul Khalid, N.; Md Ariff, N.; Yahya, S.; Mohamed, N. A Review of Bio-inspired Algorithms as Image Processing Techniques. In *International Conference on Software Engineering and Computer Systems 2011, ICSECS*; Springer: Berlin, Germany, 2011; Volume 179, pp. 660–686. [CrossRef]
7. Valsecchi, A.; Dubois-Lacoste, J.; Stutzle, T.; Damas, S.; Santamaria, J.; Marrakchi-Kacem, L. Evolutionary medical image registration using automatic parameter tuning. In Proceedings of the 2013 IEEE Congress on Evolutionary Computation, Cancun, Mexico, 20–23 June 2013. [CrossRef]
8. Cao, X.; Yang, J.; Wang, L.; Xue, Z.; Wang, Q.; Shen, D. Deep Learning Based Inter-modality Image Registration Supervised by Intra-modality Similarity. In *International Workshop on Machine Learning in Medical Imaging*; Springer: Granada, Spain, 2018; pp. 55–63. [CrossRef]

9. Zhu, Q.; Shi, Q. Application of Improved Genetic Algorithm in Medical Image Registration. In *Proceedings of the 2012 International Conference on Communication, Electronics and Automation Engineering*; Springer: Berlin, Germany, 2012; Volume 181, pp. 1063–1071. [CrossRef]

10. Santamaría, J.; Damas, S.; García-Torres, J.; Cordón, O. Self-Adaptive evolutionary image registration using differential evolution and artificial immune systems. *Pattern Recognit. Lett.* **2012**, *33*, 2065–2070. [CrossRef]

11. Bermejo, E.; Chica, M.; Damas, S.; Salcedo-Sanz, S.; Cordón, O. Coral Reef Optimization with substrate layers for medical Image Registration. *Swarm Evol. Comput.* **2018**, *42*, 138–159. [CrossRef]

12. Cocianu, C.L.; Stan, A. New Evolutionary-Based Techniques for Image Registration. *Appl. Sci.* **2019**, *9*, 176. [CrossRef]

13. Panda, R.; Agrawal, S.; Sahoo, M.; Nayak, R. A novel evolutionary rigid body docking algorithm for medical image registration. *Swarm Evol. Comput.* **2017**, *33*, 108–118. [CrossRef]

14. Boubenna, H.; Lee, D. Image-Based emotion recognition using evolutionary algorithms. *Biol. Inspired Cogn. Archit.* **2018**, *24*, 70–76. [CrossRef]

15. Bouter, A.; Alderliesten, T.; Bosman, P. A novel model-Based evolutionary algorithm for multi-objective deformable image registration with content mismatch and large deformations: Benchmarking efficiency and quality. In Proceedings of the Medical Imaging 2017: Image Processing, Orlando, FL, USA, 24 February 2017. [CrossRef]

16. Yang, F.; Ding, M.; Zhang, X.; Hou, W.; Zhong, C. Non-rigid multi-modal medical image registration by combining L-BFGS-B with cat swarm optimization. *Inf. Sci.* **2015**, *316*, 440–456. [CrossRef]

17. Qin, Y.; Hu, H.; Shi, Y.; Liu, Y.; Gao, H.; Chen, J.; Zhao, Q. An Artificial Bee Colony Algorithm Hybrid with Differential Evolution for Multi-temporal Image Registration. In Proceedings of the 35th Chinese Control Conference (CCC), Chengdu, China, 27–29 July 2016; pp. 2734–2739.

18. Li, T.; Pan, Q.; Gao, L.; Li, P. Differential evolution algorithm-Based range image registration for free-Form surface parts quality inspection. *Swarm Evol. Comput.* **2017**, *36*, 106–123. [CrossRef]

19. Bermejo, E.; Cordón, O.; Damas, S.; Santamaría, J. A comparative study on the application of advanced bacterial foraging models to image registration. *Inf. Sci.* **2015**, *295*, 160–181. [CrossRef]

20. Falco, I.D.; Cioppa, A.D.; Maisto, D.; Scafuri, U.; Tarantino, E. Using an Adaptive Invasion-based Model for Fast Range Image Registration. In Proceedings of the GECCO'14—2014 Genetic and Evolutionary Computation Conference, Vancouver, BC, Canada, 12–16 July 2014; pp. 1095–1102. [CrossRef]

21. Li, T.; Pan, Q.; Gao, L.; Li, W.; Li, P.; Shen, W.; Liu, X.; Yang, C.; Barthes, J.; Luo, J.; et al. Normal Histogram-Based Fruit Fly Optimization Algorithm for Range Image Registration. In Proceedings of the IEEE 20th International Conference on Computer Supported Cooperative Work in Design (CSCWD), Nanchang, China, 4–6 May 2016; pp. 357–362. [CrossRef]

22. Falco, I.D.; Scafuri, U.; Tarantino, E.; Cioppa, A.D.; Yetongnon, K.; Dipanda, A.; DePietro, R.; Gallo, L. Fast Range Image Registration by an Asynchronous Adaptive Distributed Differential Evolution. In Proceedings of the 2016 12TH International Conference on Signa-Image Technology & Internet-Based Systems (SITIS), Naples, Italy, November 28–December 1 2016; pp. 643–651. [CrossRef]

23. Edelkamp, S.; Schrödl, S. Heuristic search: Theory and applications. *Morgan Kaufmann Amst.* **2012**. [CrossRef]

24. Eiben, A.; Smith, J. *Introduction to Evolutionary Computing*; Springer: Berlin, Germany, 2015. [CrossRef]

25. Kvålseth, T. On Normalized Mutual Information: Measure Derivations and Properties. *Entropy* **2017**, *19*, 631. [CrossRef]

26. Yang, X. *Nature-Inspired Metaheuristic Algorithms*; Luniver: Frome, UK, 2010.

27. Ritthipakdee, A.; Thammano, A.; Premasathian, N.; Jitkongchuen, D. Firefly Mating Algorithm for Continous Optimization Problems. *Comput. Intell. Neurosci.* **2017**, *2017*. [CrossRef] [PubMed]

28. Memari, A.; Ahmad, R.; Akbari Jokar, M.R.; Abdul Rahim, A.R. A New Modified Firefly Algorithm for Optimizing a Supply Chain Network Problem. *Appl. Sci.* **2019**, *9*, 7. [CrossRef]

29. *Handbook of Metaheuristics*; Gendreau, M.; Potvin, J.Y. (Eds.) Springer: New York, NY, USA, 2010. [CrossRef]

30. Hart, W.E.; Krasnogor, N.; Smith, J.E. (Eds.) *Recent Advances in Memetic Algorithms*; Springer: New York, NY, USA, 2005. [CrossRef]

31. Cocianu, C.; Stan, A. New Attempts in Binary Image Registration. In *The Proceedings of 5th International Conference on Control, Decision and Information Technologies*; IEEE: Thessaloniki, Greece, 2018; pp. 253–258. [CrossRef]