Self-Adaptive Evolution Toward New Parameter Free Image Registration Methods

José Santamaría, *Member, IEEE,* Sergio Damas, *Member, IEEE,* Oscar Cordón, *Senior Member, IEEE,* and Agustín Escámez

***Abstract*—Image registration (IR) is a challenging topic in both the computer vision and pattern recognition fields; its main aim is to find the optimal transformation to provide the best overlay or fitting between two or more images. Usually, the success of well-known algorithms, such as iterative closest point, highly depends on several assumptions, e.g., the user should provide an initial near-optimal pose of the images to be registered. In the last decade, a new family of registration algorithms based on evolutionary principles has been contributed in order to overcome the latter drawbacks. However, their performance highly depends on carefully tuning (usually by hand) the control parameters of the algorithm, which is an error-prone and a time-consuming task. In this paper, we propose a new self-adaptive evolution model to deal with IR problems. To our knowledge, this is the first time a self-adaptive approach has been used for tuning the control parameters of evolutionary algorithms tackling computer vision tasks. Specifically, we introduce a novel design of the proposed self-adaptive approach facing pair-wise range IR problem instances, which is a challenging real-world optimization problem. In addition, several classical approaches, as well as state-of-the-art evolutionary IR methods, have been considered for numerical comparison.**

***Index Terms*—3-D modeling, evolutionary algorithms (EAs), image registration (IR), range images, self-tuning.**

I. Introduction

IMAGE REGISTRATION (IR) is a fundamental task in computer vision aimed at finding either a spatial *transformation* (e.g., rotation, translation) or a correspondence (matching of similar image features) among two or more images acquired under different conditions [1]. Over the years, IR has been applied to tackle many real-world problems ranging from remote sensing to medical imaging, artificial vision, and computer-aided design (CAD). Likewise, different techniques facing the IR problem have been studied, resulting in a large body of research. Several recent contributions reviewing the state of the art on IR methods can be found in [1]–[3].

In the last few years, specialized communities have experienced a growing interest in using improved techniques to build high-quality 3-D models of real-world objects and scenes acquired by using range scanners [4]. The goal is not to require humans to manually produce these models using laborious and error-prone CAD-based approaches. Usually, the iterative closest point (ICP) algorithm [5], [6] is the *de-facto* standard for doing pair-wise IR of range images to build the 3-D models in a process called 3-D modeling or reconstruction. However, the optimization nature of this class of methods (i.e., solution estimation using the least-squares approach) needs some further assumptions. For example, they assume that a near-optimal pose estimation is initially provided to guarantee convergence to the optimal solution. Otherwise, the IR process will be likely to get trapped in a local optima solution [7].

Alternatively, *approximate* or *heuristic* optimization methods (also named *meta-heuristics* [8]) belonging to the field of evolutionary computation (EC) [9] constitute a very interesting choice since they are able to achieve good quality outcomes for complex optimization problems. Specifically, the adoption of this optimization paradigm has caused an outstanding interest in the IR community in the last decade [3], [10]. In particular, evolutionary algorithms (EAs) as genetic algorithms (GAs) have been successfully applied for tackling the IR problem.

One of the main strengths of EC-based IR methods is that they do not need an accurate estimation of the initial pose of the images to operate. Nevertheless, these evolutionary techniques require carefully tuning many control parameters (e.g., probability of either mutation or crossover operators in GAs) in order to achieve the best possible performance in each specific application tackled. Usually, determining good control parameters is a very time-consuming task, which is carried out by hand. Users often do not waste time for tuning these crucial parameter values. Hence, the automatic control of the values of various EA parameters is one of the most challenging issues in this field. The better the values of the control parameters are tuned, the better the EA performance will be when handling complex optimization problems [11].

In the last few years, advanced strategies have arisen in the EC field in order to provide new optimization algorithms with an adaptive behavior of control parameters [11]. In this paper, we aim to contribute to the IR community with a new

Manuscript received October 31, 2011; revised April 15, 2012; accepted June 17, 2012. Date of publication July 23, 2012; date of current version July 25, 2013. This work was supported in part by the Spanish Ministerio de Educación y Ciencia (Ref. TIN2009-07727) with funding from the European Regional Development Fund, and by the University of Jaén (Ref. R1/12/2010/61) with funding from Caja Rural de Jaén.

J. Santamaría is with the Department of Computer Science, University of Jaén, Jaén 23700, Spain (e-mail: jslopez@ujaen.es).

S. Damas is with the European Centre for Soft Computing, Mieres, Asturias 33600, Spain (e-mail: sergio.damas@softcomputing.es).

O. Cordón is with the University of Granada, Granada 18071, Spain (e-mail: ocordon@decsai.ugr.es).

A. Escámez is with the Telefónica Research and Development Division, Granada 18005, Spain (e-mail: escamez@tid.es).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TEVC.2012.2209890

evolutionary IR method capable of self-tuning its control parameters. Notice that previous self-adaptive frameworks such as [12] are highly sensitive to local minima. To overcome this drawback, the specific design of our self-adaptive evolutionary optimization method, *SaEvO*, takes advantage of the synergy between two different EAs: a memetic algorithm (MA) [13], [14] based on differential evolution (DE) [15] and variable neighborhood search (VNS) [16], and an artificial immune system (AIS) [17]. While the former memetic stage is aimed to optimize problem solutions, the latter one is focused on tuning the control parameters of both the global and local search steps (i.e., DE and VNS).

We will carry out an experimental study of the proposed method performance when facing pair-wise IR problem instances of real-world objects 3-D modeling. We compare our proposal with other state-of-the-art EC-based IR algorithms, as well as with classical IR methods by using several range image datasets.

The structure of this paper is as follows. Section II presents some basics on the IR problem of range data and the application of EC principles to solve it. Our self-adaptive evolutionary approach is introduced in Section III. Section IV performs an experimental study by considering the previously introduced proposal and several state-of-the-art EC-based IR methods. Section V draws the most relevant conclusions achieved in this paper. Section VI gives conclusions and future works.

II. Range Image Registration

A. Background

Range scanner devices are able to capture 3-D images, named range images, from different viewpoints of the sensed object. Every range image partially retrieves the complete geometry of the scanned object, then placing each of them in a different coordinate system. Thus, it is mandatory to consider a reconstruction technique to perform the accurate integration of the images in order to achieve a complete and reliable model of the physical object (see Fig. 1). This framework is usually called 3-D model reconstruction [4] in which IR plays a crucial role.

Usually, there are two reconstruction approaches to integrate multiple range images [4]. The *cumulative* pair-wise approach accomplishes successive applications of the IR method, considering pairs of range images each time in order to reconstruct the original model. Alternatively, the application of a global or multiview IR stage next to the previous one is aimed to obtain a more accurate reconstruction result by simultaneously considering all these preregistered images. Regardless of the reconstruction approach considered (i.e., cumulative or multiview) the quality of the reconstructed model is highly dependent on the success of the pair-wise range IR (RIR) algorithm to be used.

There is no universal design for a hypothetical (pair-wise) IR method that could be applicable to all registration tasks, since various considerations on the particular application must be taken into account [1]. However, IR methods usually require the following four components (see Fig. 2): two input images named scene $I_s = \{\vec{p}_1, \vec{p}_2, \ldots, \vec{p}_n\}$ and model

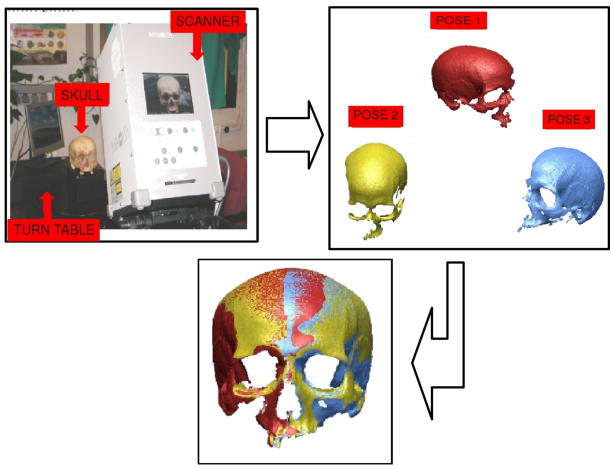

Fig. 1. Real case of 3-D model reconstruction.

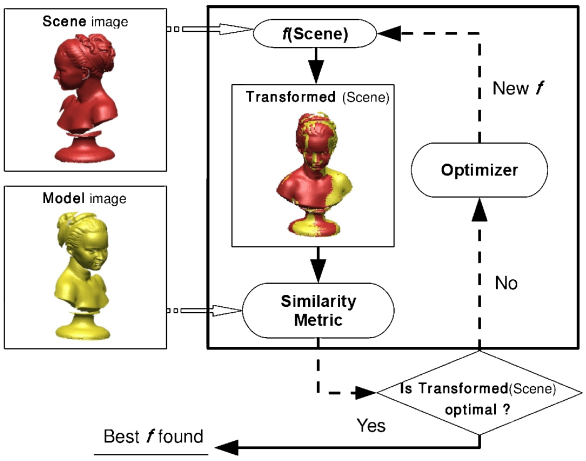

Fig. 2. General framework of the IR optimization process.

$I_m = \{\vec{p}_1{}', \vec{p}_2{}', \ldots, \vec{p}_m{}'\}$, with $\vec{p}_i$ and $\vec{p}_j{}'$ being image points; a *registration transformation* f, being a parametric function relating the two images; a *similarity metric function* F in order to measure a qualitative value of closeness or degree of fitting between the transformed scene image, noted $f'(I_s)$, and the model image; and an *optimizer* that looks for the optimal transformation f inside the defined solution search space.

Likewise, an iterative process is often followed until convergence, for instance, within a tolerance threshold of the concerned similarity metric. Specifically, the ICP algorithm [5], [6] was originally proposed to achieve an accurate estimation of the rigid pose of pairs of range images by using an optimization approach based on least-squares estimation of the registration transformation f. However, the relative rotation and traslation of the pair of images must be small for the method to converge to a good alignment.

A highly detailed survey on IR methods, as well as recognition and 3-D modeling techniques, is given in [4].

B. Evolutionary Image Registration

Since the proposal of the original ICP algorithm, many contributions have been introduced extending it [7], [18]. Despite the latter, the pair of images must usually be prealigned

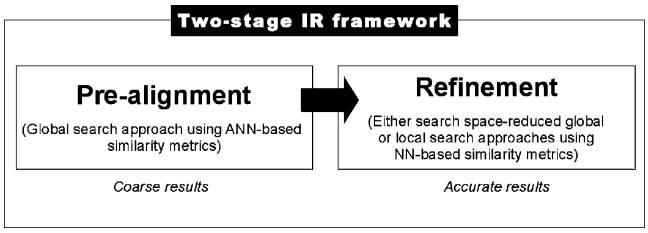

Fig. 3. Nearest neighbor (NN) and the approximate NN approaches are usually adopted for computing the closest point rule in the refinement and the prealignment stages.

manually before running any ICP variant. In order to address this shortcoming, some authors proposed using a two-stage IR approach [18] (see Fig. 3) in which a first coarser and time-consuming prealignment stage makes use of EAs, and the subsequent refinement one applies their Trim-ICP variant. Recently, it has been demonstrated that outcomes based on ICP can be outperformed using a different refinement approach, named the interpenetration measure [19].

We focus our attention on the prealignment RIR step using the EC paradigm. Specifically, EAs [9] make use of computational models of evolutionary processes as key elements in the design and implementation of computer-based problem-solving systems. In the last decade, there has been an increasing interest in applying EC principles to complex optimization tasks due to their capability to escape from local optima. Optimization procedures using stochastic schemes as those based on EAs are empirically found to provide global near-optimal solutions for complex optimization problems, including some examples from the computer vision and the computer graphics fields. GAs are the more extensively adopted EAs facing optimization problems. The first attempts to face the IR problem using EC can be found in the 1980s [20], where a GA was developed for tackling rigid IR of 2-D angiography images. Since then, evolutionary IR has become a very active area due to the successful results obtained and several well-known EAs have been considered to tackle the IR optimization process. Two detailed surveys on the application of evolutionary IR methods to 3-D modeling and medical IR are given in [3] and [10], respectively.

C. Evolutionary RIR Methods for 3-D Modeling

As said, the 3-D model reconstruction pipeline involves the application of several pair-wise alignments of two adjacent range images (see Fig. 1) in order to obtain the final 3-D model of the physical object [21]. Therefore, every pair-wise IR method aims to find the Euclidean motion that brings the *scene* view (I_s) into the best possible alignment with the *model* view (I_m). This Euclidean motion is usually considered based on a 3-D rigid transformation (f) determined by six or seven real-coded parameters when using either Euler or axis plus angle representation for rotation, respectively. In this contribution, we define the rigid transformation as a rotation $R = (\theta,\ \vec{Axis}_x, \vec{Axis}_y, \vec{Axis}_z)$ and a translation $\vec{t} = (t_x, t_y, t_z)$, with θ and $\vec{Axis}$ being the angle and axis of rotation, respectively. The transformed points of the scene view are denoted by

$$f(\vec{p}_i) = R(\vec{p}_i) + \vec{t} \quad i = \{1, \ldots, n\} \tag{1}$$

where n is the number of points of the I_s image. Hence, the evolutionary pair-wise RIR procedure can be formulated as a numerical optimization problem (in which RIR solutions are 7-D real-coded vectors $x = \langle \theta, \vec{Axis}_x, \vec{Axis}_y, \vec{Axis}_z, t_x, t_y, t_z \rangle$ developed to search for the Euclidean transformation f^* achieving the best alignment of both $f(I_s)$ and I_m

$$f^* = \arg\min_f \ F(I_s, I_m; f) \ \text{ s.t. } f^*(I_s) \cong I_m \tag{2}$$

according to the similarity metric, F, being optimized. Among others, the median square error (MedSE) is usually considered as the F function in 3-D modeling [22] due to its robustness in the presence of outliers (e.g., acquired noisy range images) in the RIR process. It can be formulated as

$$F(I_s, I_m; f) = MedSE(d_i^2), \quad \forall i = \{1, \ldots, n\} \tag{3}$$

where $MedSE()$ corresponds to the median value of all the squared Euclidean distances, $d_i^2 = \|f(\vec{p}_i) - \vec{p}'_j\|^2$ ($j = \{1, \ldots, m\}$), between the transformed scene point, $f(\vec{p}_i)$, and its corresponding closest point, $\vec{p}'_j$, in the model view I_m. Notice that it can be said that both the F function and either the fitness or the objective function (see Section II-B) have the same meaning within the optimization process.

In order to speed up the computation of the closest or nearest point (see Fig. 3) of every $f(\vec{p}_i)$ point, advanced indexing structures, such as kd-trees [19] or the grid closest point (GCP) transform [23], are often used.

III. Self-Adaptive Evolutionary Image Registration Proposal

Before running any EA, there is a need to choose the most appropriate components (see Section II-B), i.e., how to generate both the initial population and the subsequent trial solutions. Moreover, it is well known that each of these components may have several parameters that need to be carefully tuned. For instance, we found the mutation and crossover probabilities, and the tournament selection size, among others [9]. The values of these parameters strongly influence the performance of the optimization process, determining whether the algorithm will efficiently find a near-optimum solution. However, the election of the right parameter values is a time-consuming task.

The parameter tuning procedure has a potential of adjusting the optimization algorithm to the problem domain being solved. The automatic control of the values of various parameters has been one of the most challenging issues since the origins of EAs. Thus, some relevant contributions have been proposed in the last few years [11], [24], [25].

Our contribution is two-fold: 1) the proposal of a new self-adaptive evolutionary optimization (SaEvO) method for tackling a computer vision task, in particular RIR problems, and 2) simultaneously achieving high-quality registration solutions even better than those obtained by cutting-edge EA-based IR algorithms from the state of the art.

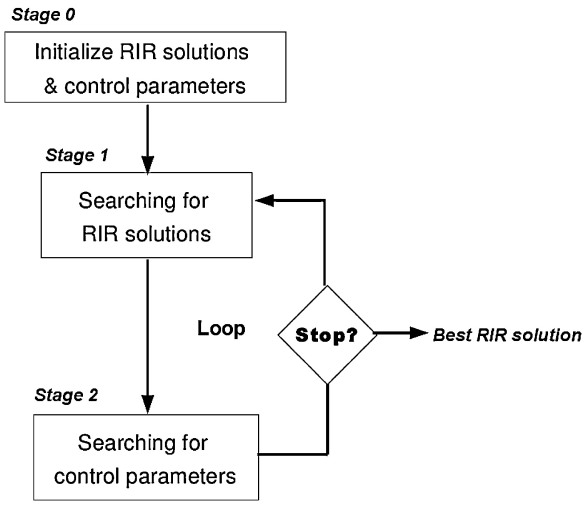

Fig. 4. Proposed self-adaptive RIR optimization framework.

Specifically, our proposed framework resembles the meta-optimization approach adopted in [26] (earlier termed *meta-GA* [27]) and it is inspired, but not based, on the two-stage co-operative approach introduced in [28]. Fig. 4 depicts our design for tackling the optimization task within the RIR problem.

After randomly initializing both the registration solutions and the control parameters (*stage0*), the optimization procedure (*stage1*, based on the DE global search scheme) searches for near-optimal registration solutions. Then, the second one (*stage2*, based on the AIS) searches for near-optimal values of the control parameters of the previous stage. Later, the two linked optimization stages iteratively evolve by using the same fitness function [see (3)]. Thus, the more adapted the control parameters of the first optimization stage are, the more accurate the registration solutions will be.

Unlike [28], our proposal extends the capabilities of the co-operative scheme based on the DE [15] algorithm and the AIS [17] by incorporating a local search (LS) strategy to the former in order to find more accurate RIR solutions. In particular, we used the VNS method [16] as the LS component. Similarly to DE, VNS is easy to implement and requires few control parameters. This hybrid approach, also known as MA [14] in the specific literature, has been successfully applied in IR [19], [29].

Contrary to previous IR contributions, SaEvO performs the self-adaption of the control parameters of both the global and the LS phases of the *stage1*. In particular, it is well known that the adaptation of the steplength parameter is central to EAs [30], [31], being one of the main novelties of our proposal in which such a control parameter of the LS procedure is also self-tuned.

The Appendix provides a more detailed algorithmic description of SaEvO in order to make its understanding easier for its further development.

IV. Experimental Results

This section is aimed at presenting a number of experiments to study how robust and accurate the results obtained by the proposed self-adaptive optimization framework are. As a benchmark, the results obtained by our SaEvO-based prealignment algorithm (see Fig. 3) will be compared against those obtained by four state-of-the-art IR methods also using evolutionary approaches, as follows:

1) Santamaria *et al.*'s proposal (Santamaria09) [29], a recent memetic contribution based on the scatter search algorithm [32];
2) de Falco *et al.*'s method (deFalco08) [33], which makes use of a basic implementation of the DE method;
3) Silva *et al.*'s contribution (Silva05) [19], in which a steady-state GA variant is developed;
4) Wachowiak *et al.*'s proposal (Wachowiak04) [34], in which the authors considered a particle swarm optimization (PSO) algorithm [35], [36] for carrying out 2-D to 3-D medical IR. In particular, we included the PSO variant proposed by the authors that achieved the best performance facing the latter IR problem, called *PSO7*.

In addition, we used the improved variant of the original ICP method proposed by Zhang [37], I-ICP, to carry out the refinement step (see Fig. 3). This method has shown a more accurate performance compared with other ICP variants (see the minimum values shown in Table III).

The proposed algorithm (SaEvO), the four baseline IR methods, and the I-ICP algorithm are implemented in C++ and compiled with the GNU/g++ tool. We adapted all the tested methods by using the same representation of the rigid transformation (f) and objective function [see (3) in Section II-C]. Other similarity metrics can also be used as fitness functions depending on the application being faced.

In order to perform a fair comparison among the prealignment RIR methods included in this paper, we considered the run (CPU) time as the stop criterion. The refinement stage considers a maximum value of 100 iterations for the I-ICP algorithm.

A. Range Image Datasets and Problem Scenarios

In order to ease the comparison with the results reported in other contributions in the field [2], [19], our experiments correspond to a number of pair-wise RIR problem instances, using different range datasets obtained from the well-known public repository of the Signal Analysis and Machine Perception Laboratory (SAMPL, http://sampl.ece.ohio-state.edu/data/3DDB/RID/index.htm). Specifically, Fig. 5 shows the eight range datasets we considered, named in previous contributions [19]: *Bird*, *Duck*, *Frog*, *Tele*, *Angel*, *Buddha*, *Bunny*, and *Lobster*. The datasets range from 8000 to 15 000 points in size.

The second group of datasets are specific ones acquired by the Physical Anthropology Laboratory, University of Granada, Granada, Spain, using their Konica-Minolta VI-910 laser scanner. They are named *Skull*, *Sculpture*, and *Tooth* (see Fig. 6). This second group of datasets ranges from 30 000 to 70 000 points size.

Concerning the views misalignment, 20 and 40 overlapping degrees of the turn table have been commonly used in the literature for testing RIR proposals [2], [19]. In our case,

TABLE I
RIR PROBLEM SCENARIOS CONSIDERING 20°, 40°, 45°, AND 60° OF OVERLAPPING

	SAMPL Datasets								Specific Datasets		
Overlap	*Bird*	*Duck*	*Frog*	*Tele*	*Angel*	*Buddha*	*Bunny*	*Lobster*	*Skull*	*Sculpture*	*Tooth*
20°	✓	✓	✓	✓							
40°	✓	✓	✓	✓	✓	✓	✓	✓			
45°									✓		
60°										✓	✓

Each scenario collects one or more pairs of range images from each of the two considered groups of datasets: SAMPL and specific.

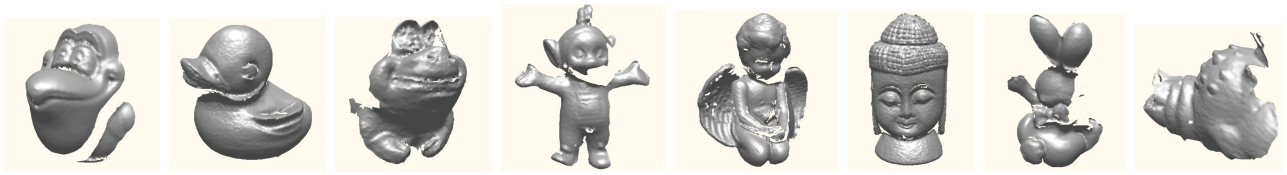

Fig. 5. Range image datasets available at the SAMPL repository. From left to right: *Bird*, *Duck*, *Frog*, *Tele*, *Angel*, *Buddha*, *Bunny*, and *Lobster* images.

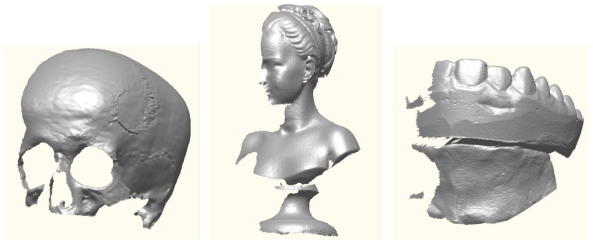

Fig. 6. Specific range image datasets. From left to right: *Skull*, *Sculpture*, and *Tooth* images.

TABLE II
STATISTICAL RESULTS OF SILVA05 CONSIDERING DIFFERENT RUNTIME LIMITS AND MAINTAINING THE SAME NUMBER OF SAMPLED POINTS (40 000)

Runtime (s)	Min.	Max.	Mean	St. dev.
20	0.6741	0.9742	0.9169	0.0628
60	0.3041	0.8986	0.7401	0.1638
120	0.2500	0.8929	0.6656	0.1888

four different pair-wise RIR problem scenarios have been set up in this experiment. Specifically, we considered increasing complexity scenarios using 20°, 40°, 45°, and 60°. As stated, the lower the rotation degree of the turn table is, the more overlapped the adjacent images and the easier the RIR problem will be (see Section II-A). Table I details how each dataset has been assigned to every RIR problem scenario.

We used both the GCP structure (see Section II-C) and the subsampled version of each range image in order to speed up the computation of the objective function [see (3)]. In particular, Table II shows some numerical results reported by Silva05, considering 40 000 points randomly chosen (using a uniform distribution) from the original *Skull* dataset and three different runtime limits 20, 60, and 120 s. The importance of achieving a suitable tradeoff between the number of points and the runtime in order to provide proper convergence conditions is remarkable. Hence, we have randomly chosen 5000 points and 20 s as a good threshold, allowing all the methods to achieve good results.

Notice that feature-based approaches [1], [3], [10] can also be adopted to achieve a reduced and a characteristic subset of image points. However, the latter procedure usually needs the intervention of expert users to obtain high-quality features.

B. Parameter Settings

In order to avoid execution dependence, 30 different runs have been performed for each of the five tested prealignment RIR algorithms when facing each of the two kinds of problem scenarios considering 20° and 40° of image overlapping in the classical datasets. All tested algorithms start from an initial population of random solutions. Then, each run concerns applying a rigid transformation, randomly generated using a uniform distribution to the scene image $f_r(I_s)$ and using the RIR method to search for the optimal transformation f^* between the proposed image $f_r(I_s)$ and the model image I_m. Every rigid transformation is randomly generated as follows: each of the three rotation axis parameters will be in the range $[-1, 1]$, the rotation angle will range in $[0°, 360°]$, and the range of three translation parameters is $[-40\,\text{mm}, 40\,\text{mm}]$.

All the pair-wise RIR methods are run on a PC in an Intel Pentium IV 2.6 MHz processor and 2 GB RAM. We considered the values of the control parameters of each of the four compared algorithms (Santamaria09, deFalco08, Silva05, and Wachowiak04) as those used in their original contribution. Regarding SaEvO, we initially experimented using different population sizes for the optimization procedure of the *stage1* and noticed that it achieved a stable performance. Specifically, we considered $l = \{50,\ 65,\ 80,\ 95\}$ as population sizes for testing SaEvO using the *Bird* dataset with 20° of overlapping (see Section IV-D) and nonsignificant differences were obtained (considering 30 different runs) with respect to the mean value of the F function, i.e., $\{0.1812,\ 0.1864,\ 0.1814,\ 0.1824\}$. Such an attractive behavior is partially supported by the self-adaptive nature of SaEvO, which demonstrates its capability for finding optimal values of the control parameters regardless of the considered population size. Nevertheless, stable results

TABLE III
STATISTICAL RESULTS OF SILVA05, I-ICP, AND LIU-ICP ALGORITHMS FACING 30 DIFFERENT PREALIGNMENT SCENARIOS

Algorithm	Min.	Max.	Mean	St. dev.
Silva05	1	13 118	2695	3941
I-ICP	4	14 822	6016	5028
Liu-ICP	163	12 993	5319	4033

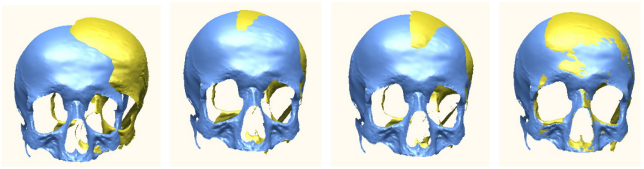

Fig. 7. Outcomes of each of the three methods tackling the same problem instance (run extracted from Table III). From left to right: the initial pose and the poses estimated by I-ICP, Liu-ICP, and Silva05.

will depend on the dimensionality of the optimization problem being considered, thus other semi-automatic approaches can be adopted for tuning the population size [38]. Thus, we decided to use a reduced set of solutions, $l = 50$, in order to speed up the convergence of SaEvO.

C. Prealignment: ICPs Versus EAs

Before performing the whole experimental comparison, it is interesting to first demonstrate the hypotheses formulated in Sections II-A and II-B about the necessity of using EAs to achieve coarse alignments prior to applying any refinement procedure. Some preliminary experiments were conducted using an evolutionary IR method, Silva05, and two different ICP variants proposed by Zhang [37] (I-ICP) and Liu [39] (Liu-ICP). Table III shows the statistical results (computed as suggested in [3] using the ground truth of the images in order to compare both the evolutionary and the non-evolutionary RIR methods) of 30 independent runs using a subsampled version of the *Skull* datasets of 40 000 points. The reported results show the low performance obtained by the two ICP variants according to the mean value. Moreover, there are scenarios in which more accurate results can be achieved using EAs taking a look at the minimum value. Fig. 7 remarks this fact showing an instance in which the ICP variants are invalid for prealignment. A deeper analysis of this behavior can be found in [3] and [10].

D. SAMPL Case Study

Tables IV and V show statistical results of the minimized fitness function F [see (3)] corresponding to the 30 runs carried out by each of the five evolutionary prealignment methods when facing the two RIR problem scenarios, i.e., 20 and 40° of overlapping. In particular, each column of these tables refers to the range dataset, the algorithm, and the minimum, maximum, mean, median, and standard deviation values of the F function.

According to accuracy and robustness, i.e., minimum and mean or median values of F, respectively, we can see how SaEvO achieves the best optimization performance compared

TABLE IV
PREALIGNMENT RIR RESULTS FOR THE 20° OF OVERLAPPING PROBLEM SCENARIO

Dataset	Algorithm	Min.	Max.	Mean	Median	St. dev.
Bird	SaEvO	0.1124	0.5998	0.1812	0.1151	0.1572
	Santamaria09	0.1132	0.8881	0.2075	0.1155	0.2015
	deFalco08	0.1245	0.8429	0.4793	0.5040	0.2157
	Silva05	0.1152	0.9178	0.3506	0.1365	0.3112
	Wachowiak04	0.4337	0.9613	0.8638	0.9036	0.1180
Duck	SaEvO	0.1307	0.8689	0.1917	0.1329	0.1795
	Santamaria09	0.1313	0.8402	0.2053	0.1341	0.1749
	deFalco08	0.1651	0.7159	0.4487	0.4481	0.1568
	Silva05	0.1345	0.9070	0.4329	0.3306	0.2579
	Wachowiak04	0.1936	0.9521	0.6983	0.7293	0.2159
Frog	SaEvO	0.1189	0.5296	0.1789	0.1226	0.1337
	Santamaria09	0.1194	0.8120	0.2029	0.1239	0.1756
	deFalco08	0.1322	0.7345	0.4374	0.4624	0.1615
	Silva05	0.1249	0.8555	0.4329	0.4394	0.2415
	Wachowiak04	0.3633	0.9407	0.7792	0.8373	0.1448
Tele	SaEvO	0.0735	0.1071	0.0780	0.0750	0.0080
	Santamaria09	0.0736	0.7867	0.1639	0.0753	0.2192
	deFalco08	0.0755	0.6578	0.3193	0.3215	0.1819
	Silva05	0.0750	0.9234	0.3728	0.0981	0.3366
	Wachowiak04	0.2510	0.9319	0.7801	0.8556	0.1622

The unit length is squared millimeters. The best minimum, mean, and median results are underlined.

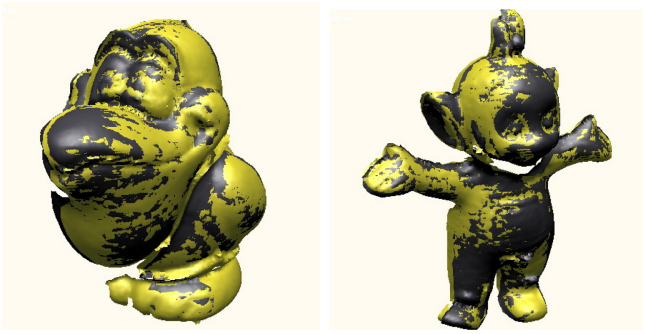

Fig. 8. Best prealignment RIR results (according to the minimum value in Table IV) of the SaEvO algorithm when facing the 20° problem scenario in the *Bird* and the *Tele* datasets.

to the other evolutionary proposals when facing every dataset of the less complex RIR scenario (20°). As an illustrative example, Fig. 8 shows two different renderings of accurate RIR results achieved by SaEvO in the *Bird* and the *Tele* image datasets.

Regarding the more complex scenario (40°), SaEvO maintains the previous outstanding robust behavior being again the best algorithm in all the eight addressed datasets with respect to the mean values. According to individual accuracy, the self-tuned proposal obtains the most accurate results (see Table IV) in five of the eight addressed datasets (i.e., *Bird*, *Frog*, *Tele*, *Angel*, and *Bunny*), while Santamaria09 achieves better results in the remaining three (i.e., *Duck*, *Buddha*, and *Lobster*). Fig. 9 shows four different renderings of these RIR results obtained by SaEvO in the *Angel*, *Frog*, *Duck*, and *Lobster* datasets.

The median values (see Tables IV and V) have been included to carry out a deeper analysis on robustness. We can see how SaEvO shows the highest scores (the lowest values) in all cases of both the 20 and 40° problem scenarios.

TABLE V
PREALIGNMENT RIR RESULTS OF THE 40° OF OVERLAPPING PROBLEM SCENARIO

Dataset	Algorithm	Min.	Max.	Mean	Median	St. dev.
	SaEvO	0.2028	0.9269	0.4451	0.2120	0.3052
	Santamaria09	0.2052	0.9373	0.4626	0.2131	0.3175
Bird	deFalco08	0.2955	0.9350	0.7358	0.7772	0.1852
	Silva05	0.2159	0.9425	0.5795	0.5809	0.3158
	Wachowiak04	0.3230	0.9653	0.9091	0.9377	0.1128
	SaEvO	0.1585	0.4822	0.2355	0.1693	0.1135
	Santamaria09	0.1582	0.8216	0.3228	0.2760	0.1398
Duck	deFalco08	0.2666	0.8245	0.6297	0.6248	0.1348
	Silva05	0.1717	0.9450	0.5207	0.5259	0.1825
	Wachowiak04	0.2171	0.9680	0.8174	0.8826	0.1848
	SaEvO	0.2536	0.7725	0.3991	0.2612	0.1963
	Santamaria09	0.2548	0.7812	0.4700	0.2759	0.2271
Frog	deFalco08	0.3997	0.8000	0.6937	0.7012	0.0876
	Silva05	0.2735	0.9474	0.6923	0.7583	0.1750
	Wachowiak04	0.2697	0.9641	0.8174	0.8525	0.1632
	SaEvO	0.1050	0.8062	0.1911	0.1085	0.1667
	Santamaria09	0.1062	0.8354	0.2217	0.1086	0.2116
Tele	deFalco08	0.1240	0.7722	0.4785	0.5001	0.1520
	Silva05	0.1077	0.8950	0.5354	0.5142	0.2929
	Wachowiak04	0.5338	0.9362	0.8034	0.8277	0.1177
	SaEvO	0.3493	0.9440	0.4983	0.3584	0.2175
	Santamaria09	0.3498	0.9539	0.5271	0.3587	0.2467
Angel	deFalco08	0.3694	0.9599	0.7954	0.8420	0.1499
	Silva05	0.3527	0.9711	0.6790	0.7546	0.2640
	Wachowiak04	0.6317	0.9736	0.9182	0.9466	0.0730
	SaEvO	0.3990	0.9032	0.6120	0.6141	0.1224
	Santamaria09	0.3978	0.7524	0.6300	0.6559	0.1020
Buddha	deFalco08	0.6705	0.9335	0.8105	0.8132	0.0812
	Silva05	0.5075	0.9506	0.7146	0.7077	0.1126
	Wachowiak04	0.4200	0.9865	0.8586	0.9031	0.1363
	SaEvO	0.0712	0.7379	0.1800	0.0734	0.2052
	Santamaria09	0.3978	0.7524	0.6300	0.0738	0.1020
Bunny	deFalco08	0.6705	0.9335	0.8105	0.4990	0.0812
	Silva05	0.5075	0.9506	0.7146	0.5837	0.1126
	Wachowiak04	0.0790	0.9145	0.7555	0.8053	0.2028
	SaEvO	0.2505	0.7582	0.3787	0.2613	0.1894
	Santamaria09	0.2490	0.8056	0.4369	0.2636	0.2231
Lobster	deFalco08	0.3392	0.7917	0.6642	0.6872	0.1020
	Silva05	0.2665	0.9201	0.5727	0.5677	0.2089
	Wachowiak04	0.5030	0.9607	0.8049	0.8390	0.1082

The unit length is squared millimeters. The best minimum, mean, and median results are underlined.

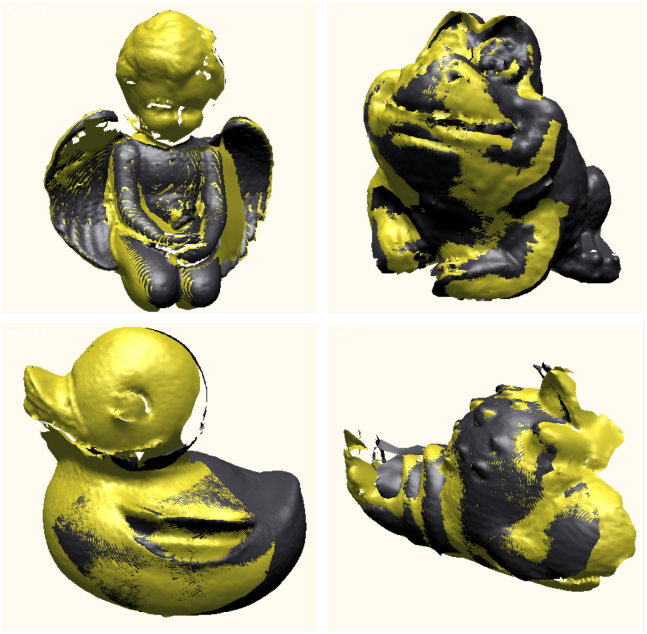

Fig. 9. Best prealignment RIR results (according to the minimum value in Table V) of the SaEvO algorithm when facing the *Angel*, *Frog*, *Duck*, and *Lobster* datasets of the 40° problem scenario.

TABLE VI
PREALIGNMENT RIR RESULTS FOR THE TWO PROBLEM SCENARIOS (45 AND 60°) USING THE SPECIFIC RANGE DATASETS

Dataset	Algorithm	Min.	Max.	Mean	Median	St. dev.
	SaEvO	0.2229	0.7806	0.3589	0.2320	0.1964
	Santamaria09	0.2260	0.8172	0.3620	0.2355	0.1974
Skull	deFalco08	0.2311	0.8465	0.6677	0.7432	0.1831
	Silva05	0.2395	0.8451	0.6691	0.7573	0.1859
	Wachowiak04	0.2874	0.9682	0.7658	0.8275	0.1832
	SaEvO	0.2446	0.3648	0.2599	0.2564	0.0198
	Santamaria09	0.2451	0.9841	0.2994	0.2571	0.1458
Sculpture	deFalco08	0.2530	0.9342	0.7081	0.8324	0.2317
	Silva05	0.2555	0.9842	0.5008	0.3103	0.2790
	Wachowiak04	0.5050	0.9874	0.9001	0.9379	0.1147
	SaEvO	0.0393	0.6409	0.1889	0.0460	0.1830
	Santamaria09	0.0443	0.7631	0.3076	0.3754	0.2301
Tooth	deFalco08	0.0454	0.7237	0.4936	0.5427	0.1922
	Silva05	0.0473	0.7511	0.3958	0.4339	0.2147
	Wachowiak04	0.0812	0.9383	0.7278	0.7702	0.1971

The unit length is squared millimeters. The best minimum, mean, and median results are underlined.

E. Specific Image Datasets Case Study

1) *Analysis of the Prealignment Results:* As in the previous section, Table VI shows the statistical results comparing the tested prealignment RIR algorithms facing the three specific range image datasets: *Skull*, *Sculpture*, and *Tooth*. From these results, we can see how SaEvO keeps on providing outstanding performance, achieving the most accurate and robust optimization results. In particular, its good behavior is highlighted in the *Tooth* dataset considering both mean and median values. Fig. 10 shows the 3-D models corresponding to the worst RIR runs of the two best methods, i.e., SaEvO and Santamaria09, tackling the most complex 60° of overlapping problem scenario using the *Sculpture* dataset.

2) *Analysis of the Refinement Results:* Finally, we completed the pair-wise RIR procedure by entirely tackling the two-stage RIR approach (see Fig. 3). As stated, we utilized the I-ICP algorithm in the refinement stage. We decided using the surface interpenetration measure (SIM) [19] for RIR evaluation purposes once each of the prealignment and the refinement stages obtained their outcomes. In contrary to the F function (used within the optimization component), high-quality RIR results show high SIM values. The SIM metric has shown more discriminant results when facing pair-wise RIR problems [19] than other classical IR evaluation metrics based on computing Euclidean distances as the well-known mean square error.

Fig. 11 shows the mean value of SIM computed from the 30 different runs for each of the three specific datasets. It is

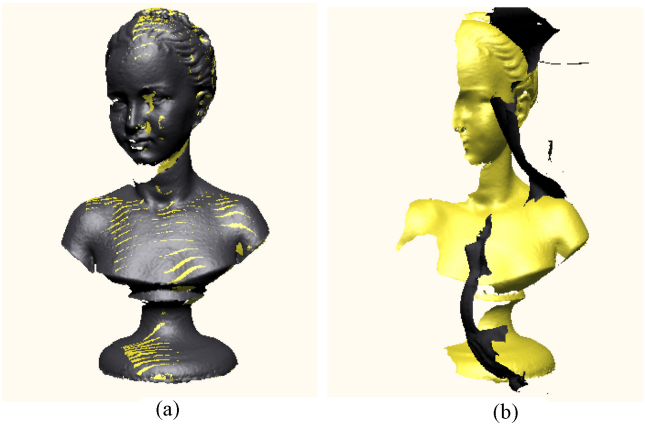

Fig. 10. Worst prealignment RIR results (according to the maximum value in Table VI) of the (a) SaEvO and (b) Santamaria09 algorithms, respectively, for the Sculpture specific dataset.

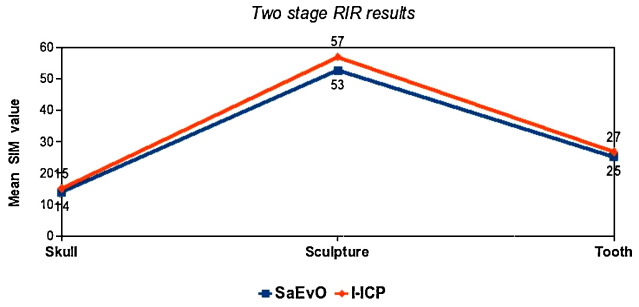

Fig. 11. Line graph showing the mean value of SIM for each of the two RIR stages based on prealignment and refinement using SaEvO and I-ICP, respectively.

remarkable how the refinement stage facing the *Skull* datasets does not show significant quantitative differences, compared to the outcomes of the initially assumed coarser prealignment stage based on SaEvO. That is, while SaEvO shows a mean value of the SIM equal to 14, the I-ICP algorithm only improves the latter result by a value of 15. Fig. 12 visually supports the latter statement for the best results obtained. This pleasant feature clearly demonstrates the outstanding behavior of SaEvO, even achieving precise RIR results close to the optimal ones obtained by adding I-ICP. Nevertheless, the more interpenetration or splotchy refinement results between surfaces in both the *Sculpture* and the *Tooth* datasets suggest that it is still necessary using this second stage when requiring highly accurate RIR outcomes.

V. Discussion

We proposed the first self-adaptive optimization approach based on MAs for tackling a well-known challenging computer vision task, i.e., the IR problem for range data. The conducted experimental results have proved that the proposed optimization framework is able to provide well-adapted control parameters in order to obtain high-quality RIR outcomes. Moreover, the reported results highlight the stable behavior of SaEvO regardless of the complexity of the problem faced, which is a precondition to be held for any non-self-adaptive optimization algorithm before benchmarking.

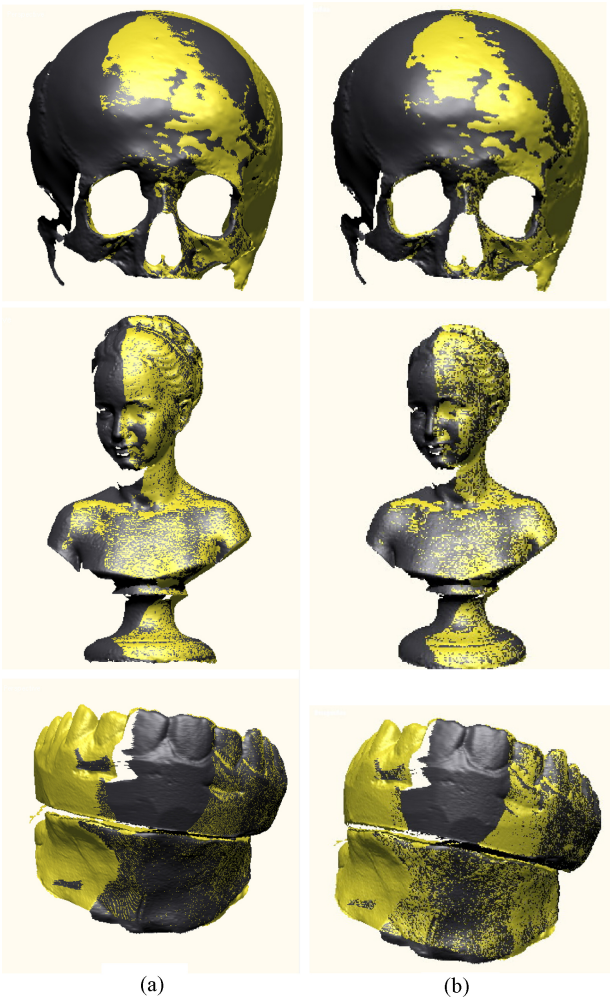

Fig. 12. Best RIR results (according to the SIM metric) of the (a) prealignment and (b) refinement stages using SaEvO and I-ICP.

What is more important, outstanding results regarding accuracy and robustness have been obtained when comparing the SaEvO approach with several state-of-the-art RIR methods also based on EAs. Specifically, we carried out a statistical test using the paired Mann–Whitney U test (also known as Wilcoxon ranksum test [40]) in order to state how significant the differences in performance are (based on the optimization scores, i.e., the F values) between the two best EA-based IR methods analyzed: SaEvO and Santamaria09. We noticed how the results obtained by the proposed algorithm significantly outperform those of the state-of-the-art algorithm Santamaria09 in all the tackled RIR problem scenarios (considering a 5% level of confidence): 20°, 40°, 45°, and 60° of overlapping.

As mentioned, we maintained the original values of control parameters of the tested state-of-the-art algorithms in order to demonstrate the importance of performing a careful tuning of these crucial variables when tackling a particular optimization problem. In particular, deFalco08 is an EA proposed for addressing IR problems of 2-D satellite images using the same evolutionary approach as in *stage1* of SaEvO, i.e., a DE algorithm. Despite the similarities between deFalco08 and SaEvO, the great differences regarding performance between

both IR methods are remarkable. Thus, it has been proved that using a self-adaptive approach of the control parameters is of key importance for achieving fully automatic optimization procedures.

As a consequence, the proposed approach can be considered as a novel parameter-free optimization tool that even provides more robust and accurate outcomes than all the methods in the state of the art. Likewise, together with the optimizer component of the IR process (see Fig. 2), its capabilities can also be utilized for simultaneously tuning the control parameters of IR components as the Similarity metric (i.e., the fitness function F). In our case, there is only a need to increase the dimensionality of the real-coded vectors evolved in SaEvO *stage2* in order to consider these new control parameters.

VI. Conclusion

IR is a very active research field today. A large number of publications related to IR show the high relevance of this topic to the computer vision and pattern recognition fields. In the last two decades, evolutionary approaches demonstrated their ability to tackle the IR problem due to their robust behavior as global optimization techniques. Contrary to traditional IR methods as the ICP algorithm, evolutionary IR methods did not need a good initial estimation of the alignment to avoid local optima in order to converge to near-optimal solutions. However, one of the main shortcomings of EC-based algorithms was that they should be carefully tuned in order to achieve the best performance for each tackled problem.

In this paper, we contributed the first self-adaptive optimization algorithm, SaEvO, based on MAs for facing IR problem instances of range data. Outstanding results were obtained when comparing SaEvO with several state-of-the-art IR algorithms, also based on EAs.

In future works, the SaEvO algorithm can be quickly extended for facing the RIR problem following the multiview approach (see Section II-A) [41]. In particular, we are working on extending the capabilities of the *MeshLab* [42] Open Source 3-D modeling package by using this new self-adaptive optimization framework. Moreover, novel designs based on the AIS metaheuristic can be considered to improve our current proposal [43], and other recent self-adaptive approaches using LS, such as in [44] and [45], will be of interest for comparing against the latter as well.

Appendix
SaEvO Pseudocode

Pseudocode 1 shows the first-level algorithmic description of SaEvO. In particular, lines 1–5, 6–36, and 37 concern *stage0*, *stage1*, and *stage2*, respectively, of the proposed framework, as shown in Fig. 4. Notice that the $U[\cdot,\cdot]$ and the $U\{\ldots\}$ symbols refer to both the float and the integer random values generated within the considered intervals using a uniform distribution. The remaining symbols introduced in Pseudocode 1 are subsequently explained along the next subsections, which are devoted to describing each of the two optimization stages, i.e., stages 1 and 2.

Pseudocode 1 First-level algorithmic description of SaEvO

Begin SaEvO

1 $t \leftarrow 0$;
2 *InitiatePopulations* (Γ_t, Ω_t, Υ_t)
3 *SortPopulation*$_{(F)}$ (Γ_t)
4 $x_{\text{best}} \leftarrow x_1$
5 *indexBestSolution* $\leftarrow 1$
6 **While** (*Not reached stop criterion*) **Do**
7 **For** $i \leftarrow 1$ **to** l **Do**
8 Randomly select $r_1 \neq r_2 \neq r_3$ ($r_j \in \{1,\ldots,l\}$)
9 $x_{\text{trial}} \leftarrow$ *SolutionGeneration* (i, Γ_t, Ω_t, r)
10 **If** ($F(x_{\text{trial}}) < F(x_i)$) **Then**
11 $aff^{o^i} \leftarrow \frac{F(x_i)-F(x_{\text{trial}})}{F(x_{\text{trial}})}$
12 $x_i \leftarrow x_{\text{trial}}$
13 **Else**
14 $aff^{o^i} \leftarrow 0$
15 **End-If**
16 **If** ($F(x_{\text{trial}}) < F(x_{\text{best}})$) **Then**
17 $x_{\text{best}} \leftarrow x_{\text{trial}}$
18 *indexBestSolution* $\leftarrow i$
19 **End-If**
20 **End-For**

21 **If** ($U[0,1] < \rho$) **Then**
22 **If** ($U[0,1] < 1-\rho$) **Then**
23 $i \leftarrow$ *indexBestSolution*
24 **Else**
25 $i \leftarrow U\{1,\ldots,l\}$
26 **End-If**
27 $x_{ls} \leftarrow$ *LocalSearch*(i,Γ_t, Ω_t)
28 **If** ($F(x_{ls}) < F(x_i)$) **Then**
29 $aff^{o^i} \leftarrow \frac{[F(x_i)\cdot aff^{o^i}+F(x_i)]-F(x_{ls})}{F(x_{ls})}$
30 $x_i \leftarrow x_{ls}$
31 **End-If**
32 **If** ($F(x_{ls}) < F(x_{\text{best}})$) **Then**
33 $x_{\text{best}} \leftarrow x_{ls}$
34 *indexBestSolution* $\leftarrow i$
35 **End-If**
36 **End-If**

37 *TuneControlParameters* (Ω_t, Υ_t)
38 $t \leftarrow t+1$
39 **End-While**
40 **Return** x_{best}

End SaEvO

A. Stage1: Searching for the Registration Solutions

As stated, the first stage of the proposed framework (see Fig. 4) is devoted to searching for RIR solutions. Our specific design is based on the optimization procedure of the DE algorithm [15], a well-known EA. It is a parallel direct search method that has proven to be a promising candidate to solve real-valued optimization problems. DE combines simple arithmetic operators with the classical crossover, mutation and selection operators of GAs within an easy-to-implement scheme. Recently, DE has been successfully applied in several IR problems [33]. In addition, another interesting advantage of DE against other EAs (e.g., GAs) is that it needs few control parameters to be tuned.

The DE algorithm starts building an initial population of solutions (line 2, Pseudocode 1), usually at random. The fundamental idea of DE is a new scheme for generating improved trial solutions, $\Gamma = \{x_1, x_2, \ldots, x_l\}$, by adding the weighted differenced vector between two population members

Pseudocode 2 The SolutionGeneration procedure

Begin SolutionGeneration

1 $j_{rand} \leftarrow U\{1, \ldots, D\}$
2 **For** $j \leftarrow 1$ **to** D **Do**
3 **If** $((U[0, 1] < \varrho_i^{CR})$ **Or** $(j == j_{rand}))$ **Then**
4 $x^j_{\text{trial}} \leftarrow x^j_{r_3} + \varrho_i^F \cdot (x^j_{r_1} - x^j_{r_2})$
5 **Else**
6 $x^j_{\text{trial}} \leftarrow x^j_i$
7 **End-If**
8 **End-For**
9 **Return** x_{trial}

End SolutionGeneration

or solutions of Γ to a third one (line 9, Pseudocode 1). There are a number of DE variants to be utilized, each one using a particular strategy for generation of trial solutions. Every generated trial solution will compete with its parent, and the former will replace the latter if a better score is obtained (lines 10–12, Pseudocode 1).

At each iteration t of SaEvO, every solution x_i of the population Γ is considered for possible replacement by a trial solution x_{trial}. As in GAs, the strategy for generation of trial solutions in DE is based on the application of both the differential (i.e., mutation) and the recombination (i.e., crossover) operators. Only one strategy is usually adopted when using DE, and the selection of the more suitable one will depend on the particular optimization problem being faced. Among all available strategies for solution generation, we used the $rand/1/bin$ scheme also adopted in [28] because of its good performance facing other optimization problems. Specifically, the *SolutionGeneration* procedure is accordingly detailed in Pseudocode 2.

As said, each generation strategy concerns both the differential and the recombination operations, and they can be implemented as follows.

Differential operation: First, for a given solution x_i, a differential solution vector z_i (noted as x_{trial} in lines 4 and 6, Pseudocode 2) is generated using the following equation:

$$z_i^d = x_{r_1}^d + \varrho_i^F \cdot (x_{r_2}^d - x_{r_3}^d) \quad \forall d \in \{1, \ldots, D = 7\} \tag{4}$$

where r_1, r_2, and r_3 are three randomly generated integers with uniform distribution and mutually different (line 8, Pseudocode 1), and the mutation factor $\varrho_i^F \in (0, 1]$ (also named F in the literature) that controls the amplification of the difference between two solutions.

Recombination operation: Subsequently, in order to increase the diversity of the new trial solution x_{trial}, recombination is applied by replacing (according to the recombination rate $\varrho_i^{CR} \in (0, 1]$, also named CR in the literature) certain randomly selected parameters (noted as $x^{j_{rand}}$ and $j_{rand} \in \{1, \ldots, D\}$) by the corresponding parameters of the previously generated differential solution z_i (noted as x_{trial} in Pseudocode 2).

In addition, our novel design of SaEvO extends the capabilities of former self-adaptive approaches by adopting an optimization scheme based on MAs, thus incorporating an LS component (noted $LocalSearch(\cdot)$ in line 27, Pseudocode 1) for improvement or refinement of problem solutions. In recent contributions, the synergy between global and local search procedures of MAs has proven to be a more robust and accurate approach for facing optimization problems such as RIR [19], [29]. As shown in lines 21–36 in Pseudocode 1, the application of the LS procedure at iteration t is carried out to a selected RIR solution by means of a probabilistic decision. As proposed in [46], we fixed an LS application probability $\rho = 0.0625$ as a suitable trade-off between intensification and diversification of the global and local search phases. Successful results were also reported in [29] when tackling RIR problems using this criterion. Lines 22–25 in Pseudocode 1 show how the parent solution x_i is selected for improvement by means of adopting the probabilistic rule $1 - \rho$, thus paying more attention to improve the best solution found so far. Otherwise, a different solution is randomly chosen from the evolved set of parent solutions of DE. Then, the LS output solution, x_{ls}, will be considered for replacement of either the parent x_i or the best x_{best} solutions in the case of the former will improve any of the latter two (lines 28–35, Pseudocode 1).

Our specific design of the LS procedure is based on VNS [16]. The basic idea of the VNS algorithm is a systematic change of the neighborhood of the current solution following a LS fashion (i.e., jumps to a new candidate solution if and only if there is an improvement). Similarly to DE, VNS is easy to implement and requires few control parameters. In particular, only the step size parameter is needed to be tuned in our design of VNS.

Pseudocode 3 details the description of the VNS-based LS procedure used in *stage1* of SaEvO (line 27, Pseudocode 1). As shown in lines 37–42, the local optimizer will iterate for either a fixed number of evaluations $\#evals = D \cdot (2 \cdot D)^{\frac{1}{2}}$ or until getting stuck at a local optimum (i.e., no improvement of the current solution x_{ls}). In case the LS procedure is successful (i.e., the new trial solution x_{temp} improves the current best x_{ls}) when $\#evals$ evaluations are reached, a new LS phase is carried out and the number of evaluations is accordingly increased (line 38, Pseudocode 3) in order to promote intensification to achieve accurate RIR results.

The core of the VNS-based LS procedure is sketched in lines 12–36. Essentially, the search procedure systematically changes to the next neighborhood structure, noted K ($K \in \{1, \ldots, K_{\max}\}$), when the current solution $x_{ls'}$ is not improved (line 22, Pseudocode 3); otherwise, the search will continue at the first neighborhood structure $K = 1$ in case of success. Specifically, using different neighborhood structure sizes means that different K real-valued parameters of the current solution $x_{ls'}$ will be randomly selected (line 13, Pseudocode 3). As shown in lines 17 and 19 in Pseudocode 3, each of the K solution parameters is systematically changed, namely *shaking* [16], based on using the specific value of the step size control parameter ϱ_i^{λ} related to the ith parent solution selected, x_i. Next, L different shaking operations are performed and L new surrounding solutions $x_{ls'}$ are generated from the current neighborhood structure, K. The best of these new solutions is accordingly selected. Notice how the extent of search space

Pseudocode 3 The VNS algorithm. Both $LB(x^i)$ and $UB(x^i)$ refer to the lower and upper bounds of the ith real-valued solution parameter ($i \in \{1, \ldots, D\}$), respectively

Begin LocalSearch

1 $K_{\max} \leftarrow D$
2 $Stop \leftarrow false$
3 $\#evals \leftarrow D \cdot (2 \cdot D)^{\frac{1}{2}}$
4 $x_{ls} \leftarrow x_{\text{temp}} \leftarrow x_i$
5 $L \leftarrow \frac{1}{\varrho_i^{\lambda}}$
6 **Do**
7 $K \leftarrow j \leftarrow 1$
8 **Do**
9 $h \leftarrow 1$
10 $improved \leftarrow false$
11 $x_{ls'} \leftarrow x_{ls}$
12 **While** ($h \leq L$ **And** $j \leq \#evals$ **And Not** $improved$) **Do**
13 Randomly select $m_1 \neq m_2 \ldots \neq m_K$ ($m_d \in \{1, \ldots, D\}$
14 **For** $k \leftarrow 1$ **to** K **Do**
15 $\omega^{m_k} \leftarrow \varrho_i^{\lambda} \cdot [UB(x^{m_k}) - LB(x^{m_k})]$
16 **If** ($U(0, 1] < 0.5$) **Then**
17 $x_{ls'}^{m_k} \leftarrow x_{ls}^{m_k} + U(\omega^{m_k} \cdot (h-1),\ \omega^{m_k} \cdot h]$
18 **Else**
19 $x_{ls'}^{m_k} \leftarrow x_{ls}^{m_k} - U(\omega^{m_k} \cdot (h-1),\ \omega^{m_k} \cdot h]$
20 **End-If**
21 **End-For**
22 **If** ($F(x_{ls'}) < F(x_{ls})$) **Then**
23 $improved \leftarrow true$
24 $x_{ls} \leftarrow x_{ls'}$
25 **Else**
26 $x_{ls'} \leftarrow x_{ls}$
27 **End-If**
28 $h \leftarrow h + 1$
29 $j \leftarrow j + 1$
30 **End-While**
31 **If** (**Not** $improved$) **Then**
32 $K \leftarrow Modulus(K + 1, K_{\max}) + 1$
33 **Else**
34 $K \leftarrow 1$
35 **End-If**
36 **While** ($j < \#evals$)
37 **If** ($F(x_{ls}) < F(x_{\text{temp}})$) **Then**
38 $\#evals \leftarrow \#evals + \frac{1}{3} \cdot \#evals$
39 $x_{\text{temp}} \leftarrow x_{ls}$
40 **Else**
41 $Stop \leftarrow true$
42 **End-If**
43 **While** (**Not** Stop)
44 **Return** x_{temp}

End LocalSearch

considered in each shaking operation is increasingly widened from $L = 1$ to $\frac{1}{\varrho_i^{\lambda}}$ different gaps. Then, the success of the LS procedure clearly depends on the latter control parameter, and it is necessary to provide a suitable value along the optimization process.

As stated in Pseudocode 2, each parent solution x_i of the population Γ is linked to a 3-D vector $\varrho_i \in \Omega$ that stores a candidate value for each of the three control parameters of both the global and the LS procedures, i.e., $\{\varrho^F, \varrho^{CR}\}$ and $\{\varrho^{\lambda}\}$, respectively. The next section is devoted to describing how such control parameters are adaptively tuned in *stage2* of the proposed optimization framework (see Fig. 4).

Pseudocode 4 The TuneControlParameters procedure

Begin TuneControlParameters

1 $SortPopulation_{(aff^e)}$ (Ω_t)
2 Build the new population of antibodies Υ_{t+1} by replacing the last γ percent of antibodies in Υ_t with the (best) first γ percent of control parameters in Ω_t
3 $Hypermutation$ (Ω_{t+1})
4 $\varphi_1 \leftarrow 1$
5 **For** $i \leftarrow 2$ **to** $2 \cdot l$ **Do**
6 $\varphi_i \leftarrow \varphi_{i-1} + i^{\beta}$
7 **End-For**
8 **For** $i \leftarrow 1$ **to** $2 \cdot l$ **Do**
9 $\varphi_i \leftarrow \frac{\varphi_i}{\varphi_{2 \cdot l}}$
10 **End-For**
11 **For** $i \leftarrow 1$ **to** l **Do**
12 **If** ($U[0, 1] \leq \varpi$) **Then**
13 $k \leftarrow 1$
14 **While** ($k < 2 \cdot l$ **And** $\varphi_k < U[0, 1]$) **Do**
15 $k \leftarrow k + 1$
16 **End-While**
17 $\varrho_i \in \Omega_{t+1} \leftarrow \varrho_k \in \Upsilon_{t+1}$
18 **End-If**
19 **End-For**
20 **Return**

End TuneControlParameters

B. Stage2: Searching for the Control Parameters

The clonal selection (CS) principle of immune systems (IS) has been used as inspiration for the development of a new class of EAs, named AIS [17]. AIS have been successfully applied in different pattern matching and optimization problems.

The CS algorithm is primarily focused on mimicking the CS principle that is composed of the following mechanisms: clonal selection, clonal expansion, and affinity maturation via somatic hypermutation. At each iteration, the IS develops a memory from the previous attack of antigens by means of the generation of more adapted antibodies with the aim of achieving a more effective immune response. This natural reactive mechanism is because of the latter ones having a higher affinity for the antigens each iteration.

Each 3-D member $\varrho_i = \langle \varrho^F, \varrho^{CR}, \varrho^{\lambda} \rangle$ ($i = \{1, \ldots, l\}$) of the initial populations of both the antigens $\Omega = \{\varrho_1, \ldots, \varrho_l\}$ and the antibodies $\Upsilon = \{\varrho_1, \ldots, \varrho_{k=2 \cdot l}\}$ is randomly generated (line 2, Pseudocode 1), using a Gaussian distribution $N(0, 1)$ and a uniform distribution $U[0, 1]$ for the parameters $\{\varrho^F, \varrho^{CR}\}$ within $(0, 1]$ and $[0.1, 0.25]$ for $\{\varrho^{\lambda}\}$, respectively. At each iteration t, the affinity score is computed in *stage1* (lines 11, 14, and 29 in Pseudocode 1) each time a new and improved trial solution is found; otherwise, the corresponding antigen $\varrho_i \in \Omega$ will be updated by a 0 value. Thus, higher values of affinity mean the CS principle implemented in *stage2* obtained a better immune response by means of matching more adapted control parameters, which will support *stage1* for achieving high-quality RIR solutions. Pseudocode 4 describes the AIS-based *stage2*.

At each iteration t of SaEvO, the population of antigens is increasingly sorted (line 1, Pseudocode 4) according to the affinity scores previously computed in *stage1*. The clonal mechanism (line 2, Pseudocode 4) will promote spreading of

a greater amount of antibodies for acting against those most ranked antigens, i.e., those control parameters ϱ_i achieving accurate RIR results. Specifically, clonal reproduction considers retaining the first best γ percent (usually, γ is fixed to 20) of antigens in Ω_t for selection to the subsequent population Ω_{t+1}. Next, the hypermutation procedure (line 3, Pseudocode 4) randomly generates the next population of antigens Ω_{t+1} using a Gaussian distribution $N(0, 1)$. The aim of the latter is injecting diversity in the optimization process of this stage. Finally, the clonal selection procedure (lines 11–19, Pseudocode 4) follows a probabilistic scheme: in case $U[0, 1] < \varpi$, the IS memory is used in order to improve the immune response and best ranked antibodies in Υ_{t+1} are more likely to be selected for inclusion in the next population of antigens Ω_{t+1}. Otherwise, a previously generated antigen $\varrho_i \in \Omega_{t+1}$ in the hypermutation step is considered instead. In Pseudocode 4, lines 4–10 establish the (increasing) importance of antibodies in Υ_{t+1} (previously arranged in lines 1 and 2, Pseudocode 4). Then, the *TuneControlParameters(·,·)* procedure will return more adapted control parameters (Ω_{t+1}) to be considered in the next iteration $t + 1$ of SaEvO.

A preliminary experiment was carried out, aimed at analyzing the robustness of SaEvO using different values of the two AIS parameters, ϖ and β. As expected, we obtained the same results as those reported in [28] and [47], in which an AIS was also applied for tackling two different real-world optimization problems. Both the AIS-based *stage2* and the whole proposed framework (SaEvO) achieved significant stable optimization outcomes with respect to the latter two control parameters for all the RIR problem instances considered in this paper. Among all the existing EAs, the latter interesting feature was our main motivation for using AIS in *stage2*. Specifically, we used the same values in [28]: $\varpi = 0.25$ and $\beta = 0.8$.

ACKNOWLEDGMENT

The authors would like to acknowledge the team of the Physical Anthropology Laboratory, University of Granada, Granada, Spain (headed by Dr. Botella and Dr. Alemán) for their support during the acquisition of the specific range datasets (*Skull*, *Sculpture*, and *Tooth*).

REFERENCES

[1] B. Zitová and J. Flusser, "Image registration methods: A survey," *Image Vision Comput.*, vol. 21, no. 11, pp. 977–1000, 2003.
[2] J. Salvi, C. Matabosch, D. Fofi, and J. Forest, "A review of recent range image registration methods with accuracy evaluation," *Image Vision Comput.*, vol. 25, no. 5, pp. 578–596, 2007.
[3] J. Santamaría, O. Cordón, and S. Damas, "A comparative study of state-of-the-art evolutionary image registration methods for 3-D modeling," *Comput. Vis. Image Underst.*, vol. 115, no. 9, pp. 1340–1354, Sep. 2011.
[4] R. J. Campbell and P. J. Flynn, "A survey of free-form object representation and recognition techniques," *Comput. Vis. Image Underst.*, vol. 81, no. 2, pp. 166–210, 2001.
[5] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 239–256, Feb. 1992.
[6] Y. Chen and G. Medioni, "Object modeling by registration of multiple range images," *Image Vision Comput.*, vol. 10, no. 3, pp. 145–155, 1992.
[7] S. Rusinkiewicz and M. Levoy, "Efficient variants of the ICP algorithm," in *Proc. 3rd Int. Conf. 3-D Digital Imaging Modeling*, 2001, pp. 145–152.
[8] F. Glover and G. A. Kochenberger, Eds., *Handbook of Metaheuristics*. Dordrecht, The Netherlands: Kluwer Academic, 2003.
[9] T. Bäck, D. B. Fogel, and Z. Michalewicz, *Handbook of Evolutionary Computation*. Bristol, U.K.: IOP Publishing/Oxford Univ. Press, 1997.
[10] S. Damas, O. Cordón, and J. Santamaría, "Medical image registration using evolutionary computation: A survey," *IEEE Comput. Intell. Mag.*, vol. 6, no. 4, pp. 26–42, Nov. 2011.
[11] A. E. Eiben and S. K. Smith, "Parameter tuning for configuring and analyzing evolutionary algorithms," *Swarm Evolut. Comput.*, vol. 1, no. 1, pp. 19–31, Mar. 2011.
[12] W. Vanzella, F. A. Pellegrino, and V. Torre, "Self-adaptive regularization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 6, pp. 804–809, Jun. 2004.
[13] P. Moscato, "On evolution, search, optimization, genetic algorithms and martial arts: Toward memetic algorithms," Caltech Concurrent Computation Program, Pasadena, CA, Tech. Rep. 826, 1989.
[14] N. Krasnogor and J. Smith, "A tutorial for competent memetic algorithms: Model, taxonomy and design issues," *IEEE Trans. Evol. Comput.*, vol. 9, no. 5, pp. 474–488, Oct. 2005.
[15] R. Storn, "Differential evolution: A simple and efficient heuristic for global optimization over continuous spaces," *J. Global Optim.*, vol. 11, no. 4, pp. 341–359, 1997.
[16] N. Mladenović and P. Hansen, "Variable neighborhood search," *Comput. Oper. Res.*, vol. 24, no. 11, pp. 1097–1100, 1997.
[17] L. de Castro and J. Timmins, *Artificial Immune Systems: A New Computational Intelligence Approach*. Berlin, Germany: Springer, 2002.
[18] E. Lomonosov, D. Chetverikov, and A. Ekart, "Preregistration of arbitrarily oriented 3-D surfaces using a genetic algorithm," *Pattern Recognit. Lett.*, vol. 27, no. 11, pp. 1201–1208, 2006.
[19] L. Silva, O. R. P. Bellon, and K. L. Boyer, "Precision range image registration using a robust surface interpenetration measure and enhanced genetic algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 5, pp. 762–776, May 2005.
[20] J. Fitzpatrick, J. Grefenstette, and D. Gucht, "Image registration by genetic search," in *Proc. IEEE Southeast Conf.*, Apr. 1984, pp. 460–464.
[21] F. Bernardini and H. Rushmeier, "The 3-D model acquisition pipeline," *Comput. Graph. Forum*, vol. 21, no. 2, pp. 149–172, 2002.
[22] M. Rodrigues, R. Fisher, and Y. Liu, "Special issue on registration and fusion of range images," *Comput. Vis. Image Underst.*, vol. 87, nos. 1–3, pp. 1–7, 2002.
[23] S. M. Yamany, M. N. Ahmed, and A. A. Farag, "A new geneticbased technique for matching 3-D curves and surfaces," *Pattern Recognit.*, vol. 32, no. 11, pp. 1817–1820, 1999.
[24] Y. S. Ong, M. Lim, N. Zhu, and K. Wong, "Classification of adaptive memetic algorithms: A comparative study," *IEEE Trans. Syst., Man, Cybern. B*, vol. 36, no. 1, pp. 141–152, Feb. 2006.
[25] A. K. Qin, V. L. Huang, and P. N. Suganthan, "Differential evolution algorithm with strategy adaptation for global numerical optimization," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 398–417, Apr. 2009.
[26] M. Pedersen, "Tuning and simplifying heuristical optimization," Ph.D. dissertation, School Eng. Sci., Univ. Southampton, Southampton, U.K., 2010.
[27] R. E. Mercer and J. R. Sampson, "Adaptive search using a reproductive metaplan," *Kybernetes*, vol. 7, no. 3, pp. 215–228, 1978.
[28] H. Chunping and Y. Xuefeng, "An immune self-adaptive differential evolution algorithm with application to estimate kinetic parameters for homogeneous mercury oxidation," *Chin. J. Chem. Eng.*, vol. 17, no. 2, pp. 232–240, 2009.
[29] J. Santamaría, O. Cordón, S. Damas, J. García-Torres, and A. Quirin, "Performance evaluation of memetic approaches in 3-D reconstruction of forensic objects," *Soft Comput.*, vol. 13, nos. 8–9, pp. 883–904, 2009.
[30] A. Ostermeier, A. Gawelczyk, and N. Hansen, "Step-size adaptation based on non-local use of selection information," in *Proc. Parallel Problem Solving From Nature (PPSN III)*, 1994, pp. 189–198.
[31] D. Arnold and A. MacLeod, "Step length adaptation on ridge functions," *Evol. Comput.*, vol. 16, no. 2, pp. 151–184, 2008.
[32] M. Laguna and R. Martí, *Scatter Search: Methodology and Implementations in C*. Boston, MA: Kluwer Academic, 2003.
[33] I. de Falco, A. Della Cioppa, D. Maisto, and E. Tarantino, "Differential evolution as a viable tool for satellite image registration," *Appl. Soft Comput.*, vol. 8, no. 4, pp. 1453–1462, 2008.
[34] M. P. Wachowiak, R. Smolikova, Y. Zheng, J. M. Zurada, and A. S. El-Maghraby, "An approach to multimodal biomedical image registration utilizing particle swarm optimization," *IEEE Trans. Evol. Comput.*, vol. 8, no. 3, pp. 289–301, Jun. 2004.

[35] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Netw.*, vol. 4. Nov. 1995, pp. 1942–1948.
[36] M. Clerc and J. Kennedy, "The particle swarm-explosion, stability, and convergence in a multidimensional complex space," *IEEE Trans. Evol. Comput.*, vol. 6, no. 1, pp. 58–73, Feb. 2002.
[37] Z. Zhang, "Iterative point matching for registration of freeform curves and surfaces," *Int. J. Comput. Vision*, vol. 13, no. 2, pp. 119–152, 1994.
[38] J. Brest and M. Maucec, "Self-adaptive differential evolution algorithm using population size reduction and three strategies," *Soft Comput.*, vol. 15, no. 11, pp. 2157–2174, 2011.
[39] Y. Liu, "Improving ICP with easy implementation for free form surface matching," *Pattern Recognit.*, vol. 37, no. 2, pp. 211–226, 2004.
[40] S. García, D. Molina, M. Lozano, and F. Herrera, "A study on the use of non-parametric tests for analyzing the evolutionary algorithms behavior: A case study on the CEC2005 special session on real parameter optimization," *J. Heuristics*, vol. 15, no. 6, pp. 617–644, 2009.
[41] L. Silva, O. R. P. Bellon, and K. L. Boyer, "Multiview range image registration using the surface interpenetration measure," *Image Vision Comput.*, vol. 25, no. 1, pp. 114–125, 2007.
[42] P. Cignoni, M. Corsini, and G. Ranzuglia, "MeshLab: An open-source 3-D mesh processing system," *ERCIM News*, vol. 73, pp. 45–46, Apr. 2008.
[43] K. Woldemariam and G. Yen, "Vaccine-enhanced artificial immune system for multimodal function optimization," *IEEE Trans. Syst., Man, Cybern. B*, vol. 40, no. 1, pp. 218–228, Feb. 2010.
[44] S. Zhao, P. Suganthan, and S. Das, "Self-adaptive differential evolution with multi-trajectory search for large-scale optimization," *Soft Comput.*, vol. 15, no. 11, pp. 2175–2185, 2011.
[45] J. Brest, A. Zamuda, B. Boskovic, S. Greiner, M. Maucec, and V. Zumer, "Self-adaptive differential evolution with SQP local search," in *Proc. 3rd Int. Conf. Bioinspired Optimization Methods Their Applicat.*, 2008, pp. 59–69.
[46] W. E. Hart, "Adaptive global optimization with local search," Ph.D. dissertation, Dept. Comput. Sci. Eng., Univ. California, San Diego, 1994.
[47] K. K. Delibasis, P. A. Asvestas, and G. K. Matsopoulos, "Automatic point correspondence using an artificial immune system optimization technique for medical image registration," *Comput. Med. Imaging Graph.*, vol. 35, no. 1, pp. 31–41, Jan. 2011.

José Santamaría (M'10) received the M.Sc. and Ph.D. degrees in computer science from the University of Granada, Granada, Spain.

He has been an Assistant Professor with the Department of Programming Languages and Software Engineering, University of Cádiz, Cádiz, Spain. He joined the Department of Computer Science, University of Jaén, Jaén, Spain, in 2008. He has published more than 50 papers in international journals, book chapters, and conferences.

Sergio Damas (M'09) received the M.Sc. and Ph.D. degrees in computer science from the University of Granada, Granada, Spain, and the Associate Professor Habilitation degree from the Spain's Ministry of Education.

From 2006 to 2011, he was an Associate Researcher with the European Centre for Soft Computing (ECSC), Mieres, Asturias, Spain. Since 2011, he has been a Principal Researcher with the Fuzzy-Evolutionary Applications Research Unit, ECSC. He became an Assistant Professor with the University of Granada in 1995. He has published more than 80 peer-reviewed scientific publications, including more than 20 science citation index-journal citation report (SCI-JCR)-indexed journal papers in prestigious journals.

Dr. Damas is the Founder and the Current Chair of the IEEE Task Force on Evolutionary Computer Vision and Image Processing. He is also a Member of the European Society for Fuzzy Logic and Technology, the Soft Computing in Image Processing Working Group, and the Soft Computing and Intelligent Information Systems Research Group.

Oscar Cordón (SM'11) is currently a Full Professor with the University of Granada, Granada, Spain. He was the Founder and Leader of the Virtual Learning Center, University of Granada, from 2001 to 2005. From 2006 to 2011, he was one of the founding researchers of the European Centre for Soft Computing, Mieres, Asturias, Spain, where he was a Principal Researcher. He has published around 280 peer-reviewed scientific publications, including a research book on genetic fuzzy systems and 66 JCR-SCI-indexed journal papers.

Prof. Cordón is an Associate Editor of ten international journals, and was recognized as the IEEE TRANSACTIONS ON FUZZY SYSTEMS OUTSTANDING Associate Editor in 2008. Since 2004, he has taken many different representative positions with Eusflat and the IEEE Computational Intelligence Society (CIS). He has been a member of the IEEE CIS AdCom from 2010 to 2012.

Agustín Escámez received the M.Sc. degree in applied physics from the University of Granada, Granada, Spain, and the Post-Bachelor Two-Year degree (DEA) in physical chemistry from the University of Murcia, Murcia, Spain, where he was with the Department of Physical Chemistry working on several projects regarding critical fluids using dynamic molecular simulation algorithms. He joined the Department of Programming Languages and Software Engineering, University of Granada, where he pursued his Ph.D. studies in real-time systems. He is currently pursuing the Ph.D. degree in computer science with the Department of Computer Science and Artificial Intelligence, University of Granada.

He has been an Assistant Professor with the Department of Computer Science and Artificial Intelligence, University of Granada, Ceuta, Spain.