Image Deformation Estimation via Multiobjective Optimization

TAKUMI NAKANE[1], **(Graduate Student Member, IEEE),**
HAORAN XIE[2], **(Member, IEEE), AND CHAO ZHANG**[1], **(Member, IEEE)**
[1]Department of Engineering, University of Fukui, Fukui 910-8507, Japan
[2]Japan Advanced Institute of Science and Technology, Nomi 923-1211, Japan

Corresponding author: Chao Zhang (zhang@u-fukui.ac.jp)

ABSTRACT The free-form deformation model can represent a wide range of non-rigid deformations by manipulating a control point lattice over the image. However, due to a large number of parameters, it is challenging to fit the free-form deformation model directly to the deformed image for deformation estimation because of the complexity of the fitness landscape. In this paper, we cast the registration task as a multi-objective optimization problem (MOP) according to the fact that regions affected by each control point overlap with each other. Specifically, by partitioning the template image into several regions and measuring the similarity of each region independently, multiple objectives are built and deformation estimation can thus be realized by solving the MOP with off-the-shelf multi-objective evolutionary algorithms (MOEAs). In addition, a coarse-to-fine strategy is realized by image pyramid combined with control point mesh subdivision. Specifically, the optimized candidate solutions of the current image level are inherited by the next level, which increases the ability to deal with large deformation. Also, a post-processing procedure is proposed to generate a single output utilizing the Pareto optimal solutions. Comparative experiments on both synthetic and real-world images show the effectiveness and usefulness of our deformation estimation method.

INDEX TERMS Genetic algorithm, image deformation estimation, image registration, multi-objective evolutionary algorithm.

I. INTRODUCTION

Estimation of the deformation parameters of a target (either objects or texture) is a fundamental technique mainly for computer vision applications such as registration [1]–[3] and tracking [4]–[6]. If the geometric deformation model is constrained to only rotation and translation then the deformation is rigid. Affine or projective transformations can express more complex deformation, while practical applications, such as medical image analysis [7], [8] and morphing [9], [10], often involve non-rigid deformation with more degrees of freedom. Furthermore, the deformation between two images can be global, local, and even space-invariant, which makes the problem more challenging because the movement vector of each pixel is required to be estimated independently while preserving the smoothness. This technique can estimate the deformation parameters by deforming a template image

such that the similarity between the deformed template and a target image is maximized. The procedure of similarity maximization can be cast as an optimization problem by treating deformation parameters as decision variables and the similarity as the objective function.

As to the geometric deformation model, free-form deformation (FFD) [11] models the deformation by manipulating control points arranged in a regular lattice over the target. Each pixel moves based on weights by basis functions and displacements of surrounding control points. B-spline basis functions are generally used to weigh displacements [12]. The larger the number of control points is, the more finely the deformation can be modeled. In FFD, the influence of a control point is limited to neighbor pixels, which brings benefits with respect to ability in modeling and computational cost [13]. Due to these characteristics, FFD is allowed to model highly free and subtle deformation. However, optimization with FFD's parameters is challenging since an optimizer needs to treat the displacements of all control points as

The associate editor coordinating the review of this manuscript and approving it for publication was Ziyan Wu.
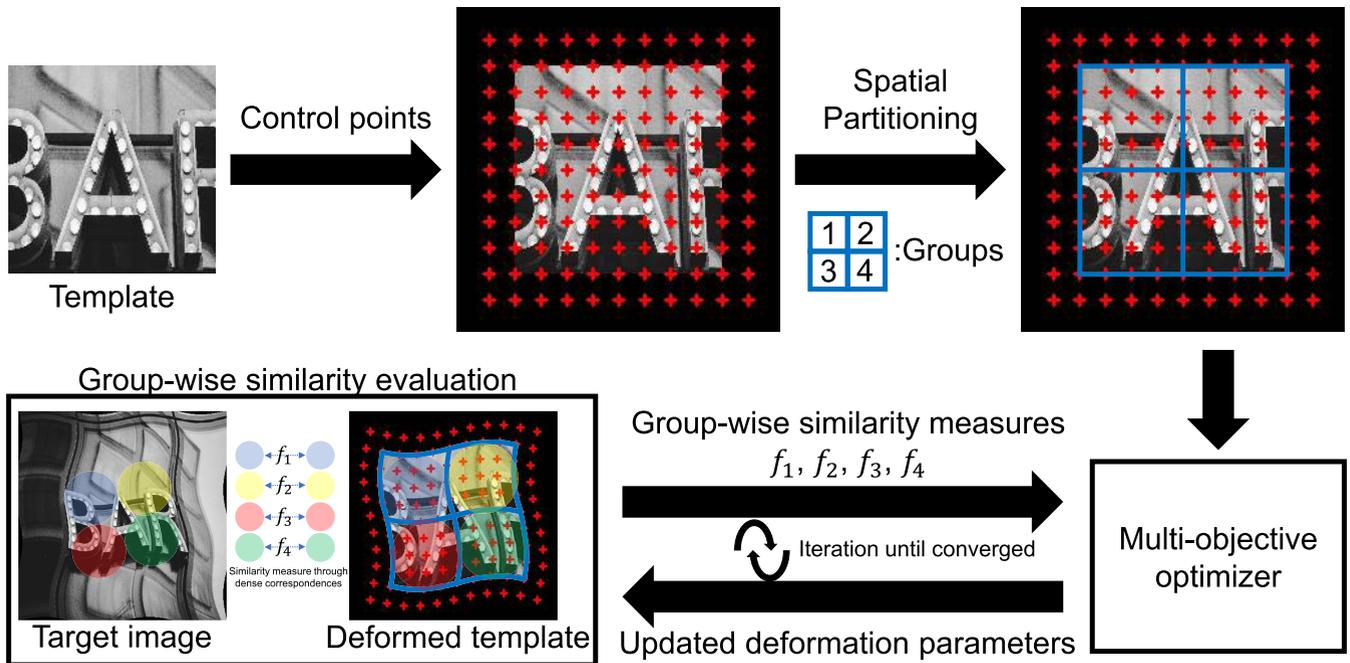
FIGURE 1. Problem setting and overview (group number is set to four here). Image deformation estimation can be regarded as an optimization problem where the goal is to search the deformation parameters which maximize the similarity between the deformed template image and the target image.

decision variables, and it is obvious that the expressive power of the deformation model is proportional to the number of parameters. Moreover, because each control point can affect multiple regions, an improvement of similarity in one region may negatively make similarity in other regions worse, i.e., there exist conflicts between regions.

To alleviate the above issues, we introduce a new idea to estimate the FFD parameters by casting the deformation estimation problem as a multi-objective optimization problem (MOP), which can be effectively solved by multi-objective evolutionary algorithms (MOEAs). The overview of our algorithm is shown in Fig. 1. A template is spatially divided into several groups and the similarity measure over each group is treated as a single-objective function and independently computed. Each group consists of patches, and the pixels in each patch are affected by the same control points. A MOP requires simultaneous optimization of two or more objectives that conflict with each other. In our problem setting, we aim to find Pareto optimal solutions given none of the groups can be improved without degrading some of the other groups, which can be solved by certain off-the-shelf MOEAs. In addition, we adopt a coarse-to-fine strategy using image pyramids to improve the estimation capability, especially for large deformations. Specifically, the optimization starts at the top of the pyramid (i.e., the lowest resolution image) and is executed at each level of the pyramid. The number of control points is gradually increased as the resolution increases, and the interpolation of control points is realized by mesh subdivision to allow fine-grained deformation. Also, a post-processing method is proposed to integrate Pareto optimal solutions into a single output as the decision-making procedure. For

each group in Fig. 1, the group-wise deformation parameters with the highest group-wise similarity are adopted. These group-wise deformation parameters are aggregated into a final solution. We perform comparative experiments using both synthetic and real-world data to show the effectiveness and usefulness of our method. In conclusion, our contributions are threefold.

- The deformation estimation problem is cast as a MOP by spatially dividing an image into multiple groups accompanied by independent similarity measures.
- The estimation capability is improved by a coarse-to-fine strategy, which is realized by building image pyramids and conducting mesh subdivisions at each level.
- A post-processing method is proposed to integrate Pareto optimal solutions into a final output.

The rest of this paper is organized as follows. We present related works of deformation estimation and MOEAs in Sec. II followed by a brief review of FFD model in Sec. III. The overview of three off-the-shelf genetic algorithms (GAs) are given in Sec. IV. In Sec. V, we describe the details of the spatial multi-objective problem and the coarse-to-fine optimization strategy. The experimental results are shown in Sec. VI. Conclusion is given in Sec. VII.

II. RELATED WORK
A. DEFORMATION ESTIMATION BETWEEN TWO IMAGES
Many methods have been proposed to deal with deformable surfaces, which can be roughly categorized as feature-based methods and pixel-based methods. The former category estimates deformation parameters by using feature

correspondences commonly extracted from two images. The latter category maximizes the similarity calculated using dense pixels directly. Also, hybrid methods have been studied to incorporate the advantages of both approaches [14]–[16].

Feature-based methods [5], [17]–[20] estimated deformation parameters based on the correspondences between feature points between the template image and the target image. The accuracy largely depends on the quality of the correspondences. Therefore, the elimination of outliers from the extracted feature set is an essential process. However, the large number of parameters in free-form deformation makes it difficult to apply standard methods such as RANSAC [19]. Other limitations are: 1) In the case of feature-less images, feature points are hard to be detected. Without inlier correspondences, the parameters cannot be appropriately estimated. Especially in the case of non-rigid transformations, more corresponding points are required [21]. 2) Local features such as SIFT [22] and ASIFT [23] are susceptible to complex transformation, which may largely degrade the confidence of correspondences when complex transformation occurs.

The purpose of pixel-based methods [1], [4], [21], [24]–[28] is to solve the minimization problem of the cost function consisting of a data term and some restrictions such as smoothness term calculated from pixel intensities. The data term is usually defined as the sum of the intensity differences between the pixels of the template image and the corresponding pixels in the deformed target image. Such methods are less dependent on image features compared to feature-based methods. In addition, the capability of dealing with self-occlusion is a notable point. Since only a few features typically exist near the self-occlusion boundary, the pixel-based methods are more reasonable in such cases [1], [15]. In [1], a penalty term called shrinker is incorporated into the cost function. The shrinker term acts to shrink the displacement in order to make self-occluded areas disappear. [15] employed a pixel-based approach to refine the deformation parameters given by a proposed feature-based method. When self-occlusion or strong deformations are involved, the hybrid method shows better results than only using the feature-based method. There also exist researches to maximize the similarity under the framework of evolutionary computation with a single-objective [27], [28]. There also exist methods achieving the minimization of the cost function by employing non-linear least squares solvers, such as the Gauss-Newton algorithm [1], [15], [21], the Levenberg-Marquardt algorithm [24], [26], and the learning-based methods [4]. To the best of our knowledge, exploiting evolutionary algorithms [29] or multi-objective optimization approaches [30], [31] to deal with deformable surfaces have been sparsely treated so far. Our previous work appearing in GECCO2019 addressed this problem by using a modified single-objective GA [32]. Different from the previous work, in this paper, we attempt to adopt evolutionary algorithms for solving this problem by casting it as a multi-objective optimization problem.

B. MULTIOBJECTIVE EVOLUTIONARY ALGORITHMS (MOEAS)

EAs are optimization algorithms inspired by Darwin's evolutionary theory, such as the GA, evolutionary strategy, and evolutionary programming. These algorithms share a common framework in which many candidate solutions are simultaneously dealt with and stochastic operations are iteratively applied. Because of the powerful exploration capability, EAs have been applied in a variety of computer vision tasks. Interested readers can also refer to the survey [33]. EAs are also effective tools for solving MOPs. The population-based search procedure provides the advantage of finding the Pareto optimal solutions in a single run. MOEAs use dominance relation to rank solutions in an objective space consisting of conflicting objectives. In particular, representative MOEAs, such as non-dominated sorting GA-II (NSGA-II) [34], strength Pareto EA2 (SPEA2) [35], and Pareto enveloped based selection algorithm-II (PEAS-II) [36], include a mechanism that preserves non-dominated solutions in every generation, called elitism, and hence these algorithms can outperform non-elitist MOEAs by preventing the loss of good solutions [37]. Since the goal of MOEAs is to provide solutions that are widely distributed on the Pareto front, MOEAs are also required to maintain the diversity of solutions. In NSGA-II, a crowding distance was proposed which is the sum of the distances between the two nearest solutions for each objective. SPEA2 used the inverse of the distance to the k-th nearest solution as the density. PEAS-II divided the objective space into several hyperboxes and counted the number of solutions within them. The density was assigned to each hyperbox as the number of solutions contained. On the other hand, MOEAs are less effective for problems with four or more objectives, i.e., many-objective optimization problems (MaOPs). The main reason is that as the number of objectives increases, the condition of dominance becomes more complex. More objectives lead to a greater proportion of non-dominated solutions, and hence the ability of convergence toward the Pareto front decreases [38]. There are several strategies to adapt MOEAs to MaOPs [39], such as dimensionality reduction [40], [41] and use of indicators [42], [43]. Among them, one representative strategy is via decomposition, e.g., MOEA based on decomposition (MOEA/D) [44], where a MaOP was decomposed into single-objective sub-problems using weighting vectors, and reference-point based many-objective NSGA-II (NSGA-III) [45], where an objective space was divided by reference vectors. There also exist various improved versions of MOEA/D [46], [47] and NSGA-III [48], [49]. Combination of both merits is also shown in [50].

III. DEFORMATION MODEL

Deformation estimation is achieved by registering the template image I to the target image I'. In order to deform I, we employ FFD combined with cubic B-splines using control point meshes. For I of $W \times H$ pixels, control points are
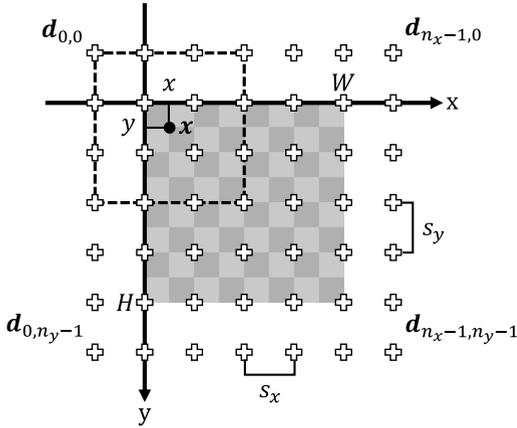
FIGURE 2. The control point lattice configuration of FFD. Deformation of the pixel coordinate x is determined by the 16 control points within/on the dashed rectangle.

FIGURE 3. (a) Deformed Fig. 2 by forward warping. (b) A closeup view around the center of (a). (c) Backward warping. (d) Shape comparison between forward warping (green dots are sampled points) and backward warping.

arranged on the $n_x \times n_y$ lattice with horizontal spacing $s_x = \lceil W/(n_x - 3) \rceil$ and vertical spacing $s_y = \lceil H/(n_y - 3) \rceil$ (i.e., the outmost control points are outside the region of I), as illustrated in Fig. 2. Each control point is assigned a displacement vector d representing the distance and direction from the initial position, and the movement of a certain coordinate $x = (x, y)^\top$ on I is determined by surrounding control points, which is defined as:

$$D(x) = \sum_{m=0}^{3} \sum_{n=0}^{3} B_m(u)B_n(v)d_{i+m,j+n}, \quad (1)$$

where $i = \lfloor x/s_x \rfloor, j = \lfloor y/s_y \rfloor, u = x/s_x - i, v = y/s_y - j$, and B_m, B_n are the cubic B-spline basis functions [51]. Then, the pixel coordinate x' on I' corresponding to x is given by the transformation function T:

$$x' = T(x) = x + D(x). \quad (2)$$
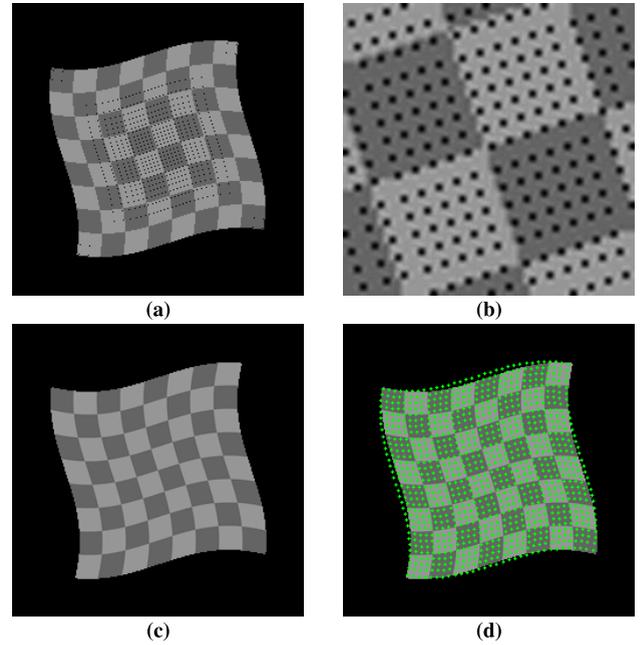
Since the transformation using Eq. 2 is a forward warping procedure and x' consists of real numbers, there is a problem that rounding operation is necessary to obtain pixel intensities. Such a process results in a large number of "holes" in the deformed template image, as illustrated in Fig. 3a and Fig 3b. An alternative is to employ backward warping that x corresponding to x' is computed using the inverse transformation T^{-1} and interpolation scheme can be used to obtain the pixel intensity. According to [52], T^{-1} can be defined using an approximation:

$$x = T^{-1}(x') \approx x' - D(x'). \quad (3)$$

The deformation obtained in Eq. 3 and its difference from Eq. 2 are illustrated in Fig. 3c and Fig. 3d, respectively. In conclusion, Eq. 1 is employed for modeling geometric deformation and Eq. 3 is employed as the image transformation for registration in this paper.

IV. REVIEW OF GENETIC ALGORITHM (GA)
To further explain how we use the multi-objective optimizer in Fig. 1 to solve the optimization problem, we provide a

short review on GAs in this section. GA is one of the leading algorithms in nature-inspired optimization methods. For a population P consisting of a number of candidate solutions, the GA gradually optimizes the P by iteratively applying genetic operators. One of the unique features of the GA is genotype representations for candidate solutions. Each candidate solution, called individual p, is encoded into an internal representation, such as a bit string or a real-valued vector, in order to apply genetic operators. The genotype representations can allow genetic operators to adapt to different problems flexibly.

The genetic operators mainly consist of parents selection, crossover and mutation. The procedure of the simple GA [53] is briefly listed as follows:

Step 1: Set $t = 0$ and generate n_P individuals in P_t randomly.

Step 2: Select individuals as parents from P_t with parents selection operator.

Step 3: Generate an offspring population Q_t from parents by crossover and mutation operators.

Step 4: Evaluate all individuals in Q_t.

Step 5: Select n_P individuals from Q_t as P_{t+1}.

Step 6: Set $t = t + 1$ and return to Step 2 until the termination criterion is satisfied.

Parent selection is typically implemented as probabilistic selection biased by evaluation values (i.e., individuals with better evaluation values are assigned higher probabilities.) The crossover operation generates offspring through the genetic recombination of multiple parents. The mutation operation randomly changes genes of offspring with low

probability. Step 5 is also known as survivors selection, that is, all individuals in P_t and Q_t compete in order to become members of P_{t+1} according to their evaluation values. The simple GA directly adopts Q_t as P_{t+1}. Besides, the elitism strategy, which preserves the best individual in the pool $R_t = P_t \cup Q_t$ is often adopted.

NSGA-II [34] is a representative GA-based algorithm for solving MOPs. The key idea of the NSGA-II is to introduce the selection criterion using two sorting approaches. The first one, called fast non-dominated sorting, iteratively extracts a non-dominated set F from the population and assigns a rank to each F according to the order in which they are extracted. Another one, called crowding distance sorting, determines the priority in F by the crowding distance which represents the density of neighboring individuals in the solution space. At last, n_P individuals are selected from S consisting of F_1 to F_{l-1}, where $|S| \leq n_P < |S| + |F_l|$. Insufficient individuals are taken from F_l according to the crowding distance. Fast non-dominated sorting promotes convergence to the Pareto front, while crowding distance sorting maintains diversity on the Pareto front. Moreover, elitism is ensured by using R in survivors selection.

NSGA-III [45] is a variant of NSGA-II which focuses on solving MaOPs. Instead of crowding distance sorting, reference lines connecting the origin with reference points evenly distributed on the evaluation value space are used to maintain the diversity of the population on the Pareto front. Each p is associated with the closest reference line in the perpendicular distance. After S is determined by fast non-dominated sorting in survivors selection, the number of individuals in S associated with each reference line, called niche count, is logged. NSGA-III then iteratively selects the individual in F_l which is associated with the reference line of the lowest niche count. Reference points can relieve the algorithm in adaptively maintaining population diversity. In addition, users can obtain only a part of the Pareto front as required by manually distributing reference points.

V. SPATIAL MULTIOBJECTIVE OPTIMIZATION

As described in Sec. III, the deformation of the template image is determined by the displacement of each control point. Therefore, the purpose of the optimization procedure is to calculate the displacements where the deformed template image matches the target in the target image most. The principal contribution of this work is to cast this task as a MOP by spatially partitioning the template into groups. Each group is assigned a single-objective function of the similarity measure. The overview of the optimization procedure is shown in Fig. 4. The procedure starts from building pyramids for both the template image and the target image, then the optimization is performed with the pyramids in a coarse-to-fine scheme. The key advantage of this framework is that the population optimized at each level can be inherited as the initial population of the next level. To ensure the consistency of parameter inheritance from low-resolution level to high-resolution level, a subdivision method considering control
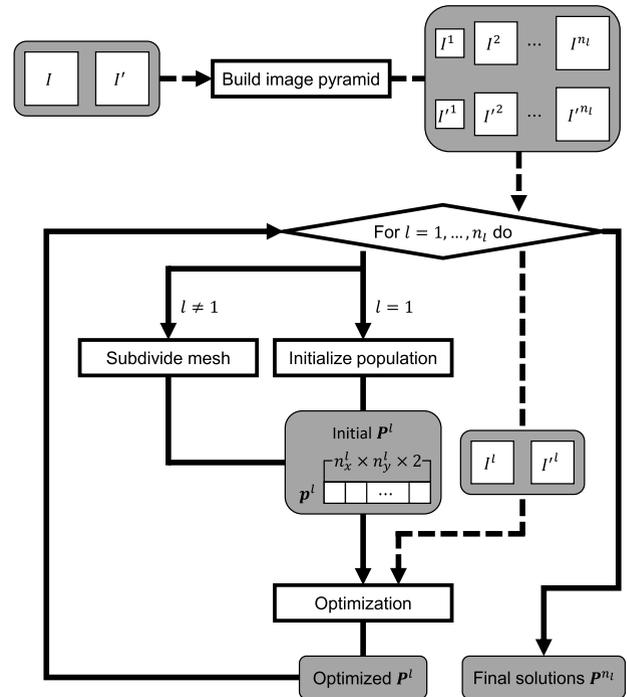

FIGURE 4. Overview of the optimization procedure.

point mesh is employed to achieve a natural interpolation of additional displacements, which allows the optimized population to be directly inherited.

Final candidate solutions are obtained by optimizing the population at the bottom of the pyramid (i.e., the image in the original resolution). The subsequent effort lies in how to determine a final solution as output from multiple solutions on the Pareto front. In addition to a direct selection approach, a post-processing approach using multiple candidate solutions is also proposed.

A. OPTIMIZING DEFORMATION PARAMETERS VIA MOEAS

Three GAs described in Sec. IV are employed and compared to optimize the displacements in the experiment. Since each control point can move within the plane, the total number of decision variables is $2n_x n_y$. To represent these variables as genes, we use real-valued coding because each displacement is a 2D vector of real values. For clarity, we denote the displacement of a single point by $\boldsymbol{d}_{i,j} = (d_{i,j,x}, d_{i,j,y})^\top$, and an individual $\boldsymbol{p}$ is represented by the vector concatenating the displacements of all the points as follows,

$$\boldsymbol{p} = (d_{0,0,y}, d_{0,0,x}, d_{1,0,y}, \ldots, d_{n_x-1,n_y-1,x})^\top. \quad (4)$$

$\boldsymbol{p}$ can directly represent a candidate control point mesh. The initial population is iteratively optimized by genetic operators. As to the evaluation of each $\boldsymbol{p}$, for simplicity, a single-objective function is firstly introduced, which is combined with the simple GA and compared to multi-objective GAs in the experiment. In the objective function, mean absolute difference with respect to intensities is used as the similarity

measure. As introduced in Sec. III, we use backward warping to find the correspondences for the calculation of objective function, hence sampling is performed on the target image. Let Ω and Ω' denote the entire region of the template image I and the target image I', respectively, and we can further define the sampling region $\omega' \subseteq \Omega'$ as:

$$\omega' = \{\boldsymbol{x}' \mid T^{-1}(\boldsymbol{x}') \in \Omega\}. \tag{5}$$

The single-objective function for the simple GA is given by:

$$f(\boldsymbol{p}) = \frac{1}{|\omega'|} \sum_{\boldsymbol{x}' \in \omega'} |I'(\boldsymbol{x}') - I(T^{-1}(\boldsymbol{x}'))|. \tag{6}$$

Based on Eq. 6, spatial multi-objective functions can be easily defined. We first define *patch* as a region consisting of pixels affected by the shared 4×4 control points, i.e., Ω consists of $(n_x - 3) \times (n_y - 3)$ patches. The group partitioning is achieved by dividing all the patches into n_f groups, where n_f is the number of objective functions. We denote the region of i-th group as $\omega_i \subseteq \Omega$. One objective function is assigned over each group, which can be written by modifying Eq. 5 and Eq. 6:

$$\omega_i' = \{\boldsymbol{x}' \mid T^{-1}(\boldsymbol{x}') \in \omega_i\}, \tag{7}$$

$$f_i(\boldsymbol{p}) = \frac{1}{|\omega_i'|} \sum_{\boldsymbol{x}' \in \omega_i'} |I'(\boldsymbol{x}') - I(T^{-1}(\boldsymbol{x}'))|, \tag{8}$$

where $i = 1, 2, \ldots, n_f$. Therefore, multi-objective GAs evaluate individuals based on the following vector function:

$$\boldsymbol{f}(\boldsymbol{p}) = (f_1(\boldsymbol{p}), f_2(\boldsymbol{p}), \ldots, f_{n_f}(\boldsymbol{p}))^\top. \tag{9}$$

Because each control point affects multiple patches, their similarity functions can "conflict" with each other, which is also considered in the case of groups consisting of multiple patches. This is an important motivation for adopting MOEAs because a single-objective optimizer can hardly solve such a conflicting problem efficiently [34]. The number of regions is adjustable as a hyper-parameter (increasing n_f makes optimization significantly more difficult). Pareto optimal solutions are supposed to obtain more appropriate solutions than optimizing a single-objective.

B. COARSE-TO-FINE STRATEGY

An iterative framework using image pyramids can be employed to alleviate the difficulty of deformation estimation with large displacements [1], [26]. Estimation starts from the lowest resolution image for a rough estimation, and more accurate estimations are achieved as the image resolution increases. Specifically, we adopt Gaussian pyramid which iteratively generates low resolution images through Gaussian smoothing. The assignment of pyramid level indices follows the order in which estimation is performed (i.e., the first and the n_l-th level images are with the lowest and the highest resolution, respectively). Both the image width and height are halved and hence the l-th level image has $1/4$ resolution of the $(l + 1)$-th level image.
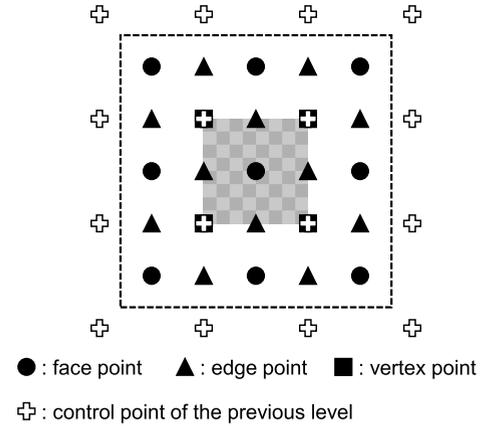
FIGURE 5. Interpolation of control points by the subdivision algorithm for the current level. The middle image is the previous level and the control points of the current level (black) are computed based on the previous level's ones (white). The outermost control points (outside the dashed rectangle) are discarded and will not be involved in deformation. Black points will be further displaced to reflect the image deformation.

With the increase of image resolution, it is also necessary to increase the resolution of the control point mesh to inherit deformation parameters between different levels. To this end, for a certain level, interpolating new control points without destroying the mesh configuration of the previous level is required. In this work, we adopt the Catmull-Clark subdivision [54] for the mesh subdivision. The purpose of subdivision is to update the $n_x^l \times n_y^l$ control point mesh with respect to each individual from the optimized $\boldsymbol{P}^l$ to $n_x^{l+1} \times n_y^{l+1}$ mesh, where $n_x^{l+1} = 2n_x^l - 3$ and $n_y^{l+1} = 2n_y^l - 3$ (i.e., s_x and s_y are fixed for all levels). The Catmull-Clark subdivision algorithm generates a subdivided mesh by inserting new control points and updating the existing control points. As illustrated in Fig. 5, the points on the subdivided mesh can be classified into three types: face points, edge points, and vertex points. A face point is inserted to a patch. Assuming that the vertices of a patch are $\boldsymbol{d}_1^l, \boldsymbol{d}_2^l, \boldsymbol{d}_3^l$, and $\boldsymbol{d}_4^l$, the face point $\boldsymbol{d}_F^{l+1}$ is computed as their centroid,

$$\boldsymbol{d}_F^{l+1} = \frac{1}{4} \sum_{i=1}^4 \boldsymbol{d}_i^l. \tag{10}$$

An edge point is inserted to the edge shared by two patches. Assuming that two face points are $\boldsymbol{d}_{F1}^{l+1}$ and $\boldsymbol{d}_{F2}^{l+1}$, and two endpoints of the edge are $\boldsymbol{d}_1^l$ and $\boldsymbol{d}_2^l$, the edge point $\boldsymbol{d}_E^{l+1}$ is computed as follow:

$$\boldsymbol{d}_E^{l+1} = \frac{\boldsymbol{d}_{F1}^{l+1} + \boldsymbol{d}_{F2}^{l+1} + \boldsymbol{d}_1^l + \boldsymbol{d}_2^l}{4}. \tag{11}$$

A vertex point is the updated point of a vertex $\boldsymbol{d}^l$ shared by the four patches. Denoting that the average of the four face points is $\bar{\boldsymbol{d}}_F$, and the average of the midpoints of the four edges which share $\boldsymbol{d}^l$ as one of the endpoints is $\bar{\boldsymbol{d}}_M$, the vertex point $\boldsymbol{d}_V^{l+1}$ is computed as follows:

$$\boldsymbol{d}_V^{l+1} = \frac{1}{4}\bar{\boldsymbol{d}}_F + \frac{1}{2}\bar{\boldsymbol{d}}_M + \frac{1}{4}\boldsymbol{d}^l. \tag{12}$$
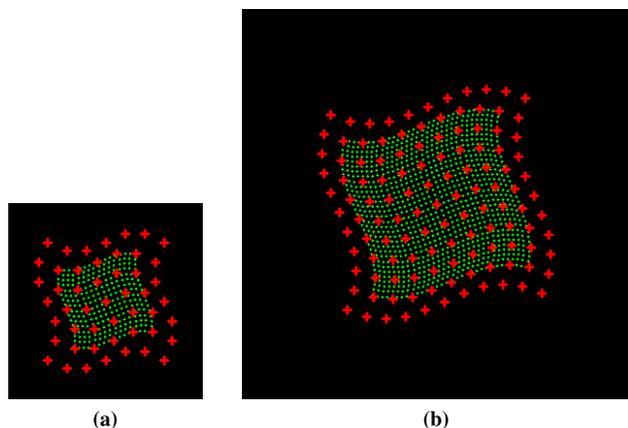
FIGURE 6. An example of a control point subdivision. Red markers represent the control points, and green dots represent the deformed surface with sampled points. (a) An example mesh with 7 × 7 lattice. (b) The subdivided mesh and enlarged surface with 11 × 11 lattice.
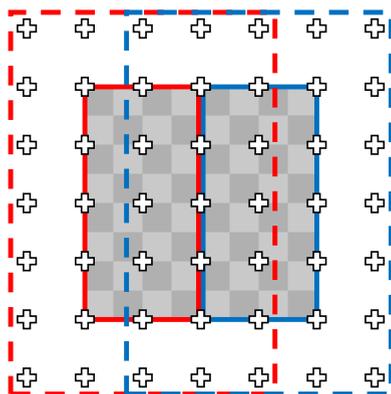
**FIGURE 7.** An example of control point aggregation under 7 × 7 lattice with two groups. The control points surrounded by one dashed rectangle are determined by the optimum solution which achieves the maximum similarity with respect to the group region surrounded by the solid rectangle of the same color. For shared control points (three columns in the center), the average of their displacements is computed.

Note that the edge points and vertex points outside the dashed rectangle are not calculated, as surrounding control points are needed for calculation. After interpolation of control points, all the displacements are doubled to fit the increase of resolution in the image pyramid. An example of the subdivision process is shown in Fig. 6. It can be observed that the subdivided mesh in Fig. 6b can maintain the shape of the previous mesh in Fig. 6a well. Therefore, this subdivision step is useful for providing a good initialization for optimizing the image of the next level in the pyramid.

C. DECISION OF THE FINAL OUTPUT

We introduce a post-processing procedure to decide the final single solution as the output from the optimized population. A natural idea is to define the best solution as the individual with the smallest sum of the objective function values (i.e., maximum similarity). However, in MOEAs, such an approach can lose most of the valuable information of the
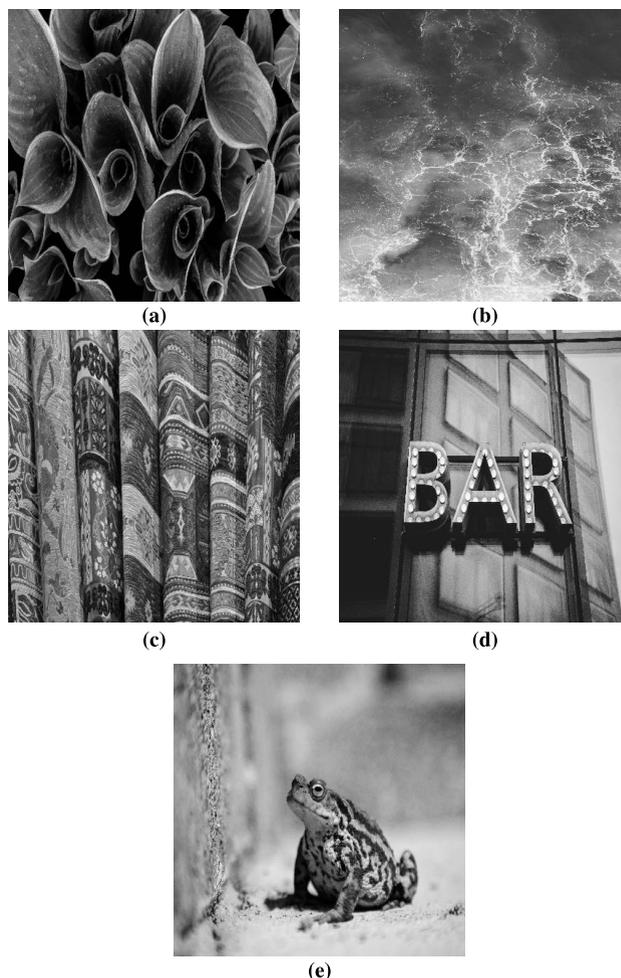
FIGURE 8. Images used for generating template and synthetically deformed target images: (a) plant, (b) sea, (c) rag, (d) alphabet, (e) toad.

Pareto optimal solutions. We propose a post-processed solution by exploiting these sub-optimal solutions, as illustrated in Fig. 7. For each group, the solution with the smallest value of the corresponding objective function is extracted, which provides the control points that only affect the corresponding group. The final output is created by aggregating control points provided from all the groups. For shared control points, the average of their displacements is computed.

VI. EXPERIMENTAL RESULTS

The effectiveness and usefulness of solving the deformation estimation problem with the multi-objective scheme are verified using both synthetic data (Sec. VI-A) and real-world images (Sec. VI-B). We compared the following four settings: the simple GA with a single-objective, NSGA-II and NSGA-III with two-objectives respectively, and NSGA-III with four-objectives. In the two-objective setting, patches are divided equally and vertically into two groups (e.g., for a 7 × 7 lattice including 4 × 4 patches, each group consists of 2 × 4 patches). Similarly, patches are divided vertically and horizontally into four groups in the four-objective setting

TABLE 1. Comparative results of the best solutions with respect to the *vertical wavy deformation*. Each cell consists of three values which represent the minimum, maximum, and average values based on five random trials from top to bottom. The best results in terms of RMSE and MEDE with respect to each row are emphasized in bold.

| Image	Lattice size	Decision variable range	RMSE				MEDE			
			GA	Two objective NSGA-II	Two objective NSGA-III	Four objective NSGA-III	GA	Two objective NSGA-II	Two objective NSGA-III	Four objective NSGA-III
Plant	7 × 7	[−5.0, 5.0]	2.28E+00	**2.01E+00**	2.16E+00	2.59E+00	1.22E-01	**1.01E-01**	1.02E-01	1.23E-01
			3.00E+00	2.76E+00	**2.53E+00**	3.36E+00	1.51E-01	**1.17E-01**	1.24E-01	1.83E-01
			2.64E+00	**2.30E+00**	2.34E+00	2.93E+00	1.35E-01	**1.10E-01**	1.13E-01	1.43E-01
		[−10.0, 10.0]	5.01E+00	**4.30E+00**	4.72E+00	5.05E+00	2.65E-01	**2.13E-01**	2.55E-01	2.76E-01
			6.06E+00	**5.55E+00**	6.15E+00	6.90E+00	3.27E-01	**3.02E-01**	3.02E-01	3.52E-01
			5.56E+00	**4.92E+00**	5.16E+00	5.93E+00	2.94E-01	**2.62E-01**	2.68E-01	3.11E-01
	11 × 11	[−5.0, 5.0]	3.26E+00	2.73E+00	**2.44E+00**	3.26E+00	1.62E-01	1.48E-01	**1.14E-01**	1.66E-01
			4.50E+00	4.23E+00	**3.78E+00**	4.16E+00	2.31E-01	2.09E-01	**1.95E-01**	2.14E-01
			3.85E+00	3.34E+00	**3.16E+00**	3.65E+00	1.96E-01	1.79E-01	**1.63E-01**	1.92E-01
		[−10.0, 10.0]	6.12E+00	**5.10E+00**	5.16E+00	6.53E+00	3.35E-01	**2.82E-01**	3.42E-01	3.66E-01
			9.34E+00	9.68E+00	**7.65E+00**	8.51E+00	6.91E-01	7.22E-01	**5.21E-01**	5.32E-01
			7.37E+00	6.89E+00	**6.84E+00**	7.43E+00	4.67E-01	4.53E-01	**4.41E-01**	4.64E-01
Sea	7 × 7	[−5.0, 5.0]	3.79E+00	3.85E+00	**3.49E+00**	4.13E+00	1.03E-01	1.07E-01	**9.17E-02**	1.41E-01
			4.19E+00	**4.16E+00**	4.36E+00	4.93E+00	1.28E-01	**1.20E-01**	1.24E-01	1.57E-01
			3.97E+00	4.04E+00	**3.94E+00**	4.50E+00	1.18E-01	1.15E-01	**1.13E-01**	1.48E-01
		[−10.0, 10.0]	8.08E+00	**7.68E+00**	8.05E+00	9.10E+00	**2.22E-01**	2.26E-01	2.46E-01	2.80E-01
			1.06E+01	**9.11E+00**	1.02E+01	1.04E+01	3.37E-01	**2.58E-01**	3.47E-01	3.73E-01
			9.10E+00	**8.46E+00**	9.11E+00	9.57E+00	2.66E-01	**2.41E-01**	3.00E-01	3.16E-01
	11 × 11	[−5.0, 5.0]	4.96E+00	4.19E+00	**3.84E+00**	4.71E+00	1.47E-01	**1.09E-01**	1.10E-01	1.47E-01
			7.10E+00	7.57E+00	**6.02E+00**	6.33E+00	**1.94E-01**	2.41E-01	2.01E-01	2.14E-01
			5.67E+00	5.37E+00	**4.78E+00**	5.35E+00	1.72E-01	1.59E-01	**1.45E-01**	1.64E-01
		[−10.0, 10.0]	8.90E+00	7.39E+00	**6.72E+00**	8.48E+00	3.85E-01	2.68E-01	**2.61E-01**	2.89E-01
			1.36E+01	**9.32E+00**	1.08E+01	1.15E+01	9.89E-01	**4.07E-01**	5.06E-01	6.42E-01
			1.14E+01	**8.12E+00**	9.03E+00	9.58E+00	6.49E-01	**3.12E-01**	3.66E-01	3.97E-01
Rag	7 × 7	[−5.0, 5.0]	3.52E+00	3.89E+00	**3.06E+00**	3.96E+00	9.64E-02	9.47E-02	**8.41E-02**	1.03E-01
			4.65E+00	4.93E+00	5.01E+00	5.18E+00	1.22E-01	**1.16E-01**	1.21E-01	1.32E-01
			4.24E+00	4.17E+00	**3.94E+00**	4.64E+00	1.10E-01	1.03E-01	**9.90E-02**	1.20E-01
		[−10.0, 10.0]	**8.12E+00**	8.18E+00	8.24E+00	8.78E+00	2.17E-01	**2.10E-01**	2.15E-01	2.27E-01
			1.01E+01	9.24E+00	**8.75E+00**	1.04E+01	2.69E-01	2.56E-01	**2.24E-01**	2.78E-01
			9.21E+00	8.73E+00	**8.54E+00**	9.59E+00	2.28E-01	**2.20E-01**	2.20E-01	2.61E-01
	11 × 11	[−5.0, 5.0]	4.97E+00	**4.08E+00**	4.88E+00	5.25E+00	1.35E-01	**1.03E-01**	1.25E-01	1.39E-01
			7.35E+00	**5.65E+00**	6.21E+00	6.13E+00	2.23E-01	**1.51E-01**	1.69E-01	1.54E-01
			6.01E+00	**4.93E+00**	5.41E+00	5.56E+00	1.68E-01	**1.23E-01**	1.43E-01	1.44E-01
		[−10.0, 10.0]	9.58E+00	8.43E+00	**7.21E+00**	1.03E+01	2.61E-01	2.31E-01	**2.03E-01**	2.81E-01
			2.15E+01	1.32E+01	1.87E+01	**1.17E+01**	8.39E-01	3.79E-01	5.93E-01	**3.54E-01**
			1.72E+01	**9.74E+00**	1.07E+01	1.10E+01	5.99E-01	**2.75E-01**	3.07E-01	3.19E-01
Alphabet	7 × 7	[−5.0, 5.0]	3.81E+00	**3.43E+00**	3.84E+00	4.55E+00	1.08E-01	**9.81E-02**	1.08E-01	1.19E-01
			4.69E+00	5.14E+00	**4.45E+00**	5.26E+00	1.41E-01	1.42E-01	**1.23E-01**	1.67E-01
			4.14E+00	4.15E+00	4.21E+00	4.90E+00	1.21E-01	1.23E-01	**1.14E-01**	1.44E-01
		[−10.0, 10.0]	7.46E+00	**7.44E+00**	7.67E+00	8.83E+00	**2.04E-01**	2.07E-01	2.36E-01	2.76E-01
			9.04E+00	9.55E+00	**8.67E+00**	1.10E+01	2.72E-01	**2.47E-01**	2.76E-01	3.45E-01
			8.44E+00	8.42E+00	**8.33E+00**	9.93E+00	2.36E-01	**2.29E-01**	2.52E-01	3.05E-01
	11 × 11	[−5.0, 5.0]	4.03E+00	3.98E+00	**3.96E+00**	5.17E+00	1.50E-01	1.36E-01	**1.34E-01**	1.95E-01
			5.93E+00	**5.17E+00**	5.21E+00	7.07E+00	2.42E-01	1.74E-01	**1.71E-01**	2.33E-01
			4.86E+00	4.76E+00	**4.35E+00**	6.07E+00	1.89E-01	1.55E-01	**1.43E-01**	2.16E-01
		[−10.0, 10.0]	1.14E+01	8.62E+00	**7.73E+00**	9.92E+00	3.99E-01	3.48E-01	**2.15E-01**	3.71E-01
			1.29E+01	**1.12E+01**	1.15E+01	1.28E+01	1.01E+00	4.63E-01	**4.32E-01**	4.92E-01
			1.19E+01	**9.50E+00**	9.86E+00	1.16E+01	5.77E-01	3.94E-01	**3.73E-01**	4.34E-01
Toad	7 × 7	[−5.0, 5.0]	1.56E+00	1.67E+00	**1.22E+00**	2.17E+00	**1.06E-01**	1.12E-01	1.23E-01	1.43E-01
			2.58E+00	2.40E+00	**2.32E+00**	2.35E+00	1.50E-01	**1.46E-01**	1.56E-01	1.61E-01
			2.03E+00	1.95E+00	**1.92E+00**	2.26E+00	1.31E-01	**1.25E-01**	1.35E-01	1.51E-01
		[−10.0, 10.0]	3.24E+00	2.76E+00	**2.62E+00**	3.53E+00	2.85E-01	2.73E-01	**2.71E-01**	2.85E-01
			4.97E+00	3.78E+00	**3.47E+00**	4.79E+00	3.92E-01	**3.36E-01**	3.41E-01	3.47E-01
			4.05E+00	3.27E+00	**3.12E+00**	4.25E+00	3.28E-01	**3.05E-01**	3.14E-01	3.18E-01
	11 × 11	[−5.0, 5.0]	2.19E+00	1.90E+00	**1.56E+00**	1.99E+00	2.31E-01	2.26E-01	2.20E-01	**2.02E-01**
			2.58E+00	**2.51E+00**	2.60E+00	3.45E+00	3.58E-01	2.95E-01	3.13E-01	**2.90E-01**
			2.39E+00	2.14E+00	**2.10E+00**	2.67E+00	3.21E-01	2.68E-01	**2.52E-01**	2.56E-01
		[−10.0, 10.0]	3.06E+00	2.97E+00	**2.81E+00**	3.44E+00	5.40E-01	**4.22E-01**	4.57E-01	4.35E-01
			4.53E+00	**3.57E+00**	4.91E+00	4.98E+00	8.54E-01	7.35E-01	7.92E-01	**5.01E-01**
			3.87E+00	**3.23E+00**	3.73E+00	4.08E+00	6.72E-01	5.18E-01	5.77E-01	**4.74E-01**

(e.g., each group consists of 2 × 2 patches for a 7 × 7 lattice). The number of pyramid levels n_l is fixed to three in all experiments. To reduce the computational cost, pixel sampling is performed for individual evaluation by scanning the target image at five pixel intervals. For the implementation of three GAs, the Platypus package[1] is used, which is an evolutionary computation framework in Python which includes many MOEAs. To ensure the correctness and fairness of the

[1] https://github.com/Project-Platypus/Platypus
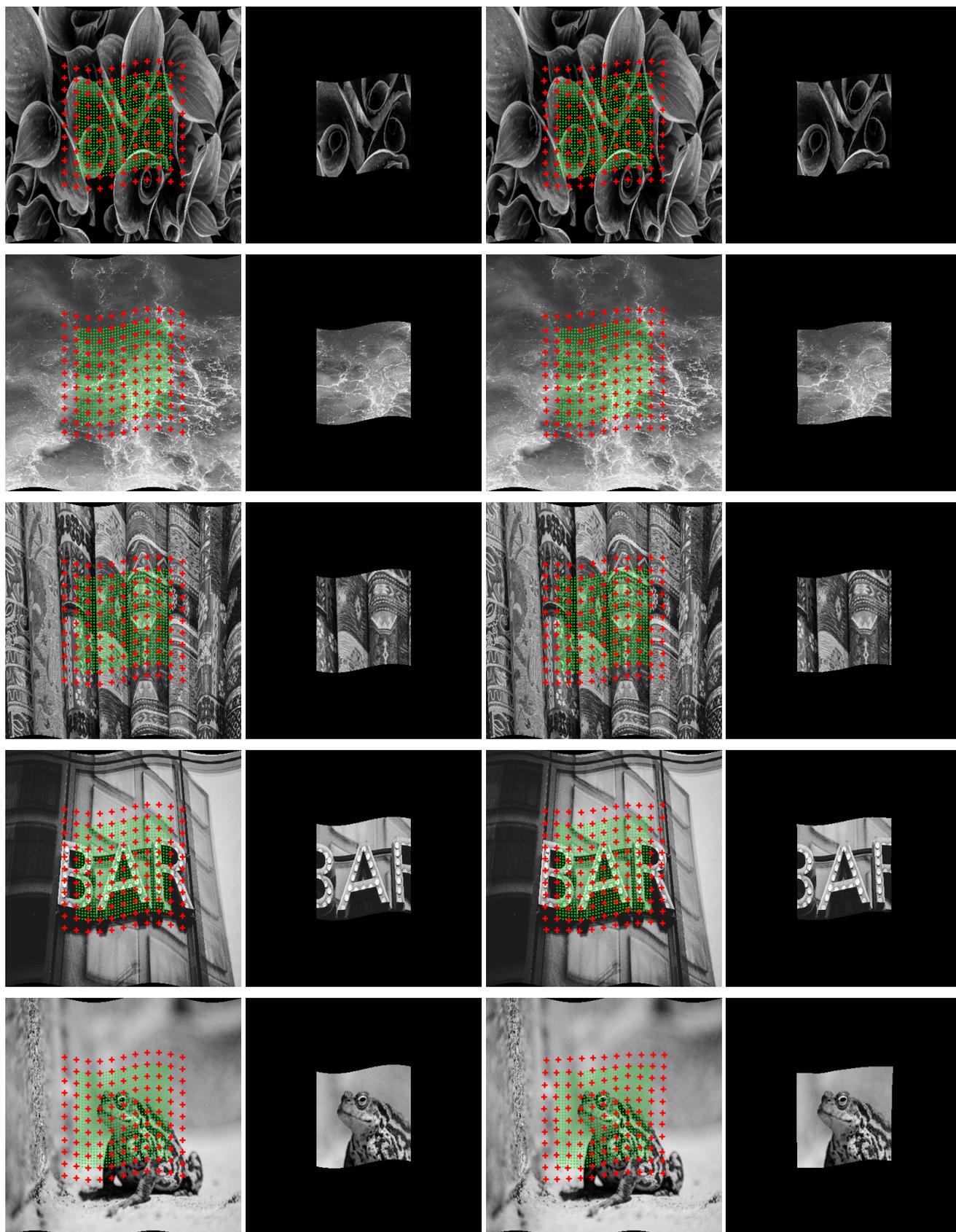
FIGURE 9. Examples of visual results achieved by multi-objective methods on *vertical wavy* images. The 1st~2nd columns are ground truth and 3rd~4th columns are results. Red markers represent the control points, and green dots represent the deformed surface generated by forward warping. Images in the 4th column show the deformed template images with estimated deformation.

(a) GA

(b) two-objective NSGA-II

(c) two-objective NSGA-III

(d) four-objective NSGA-III
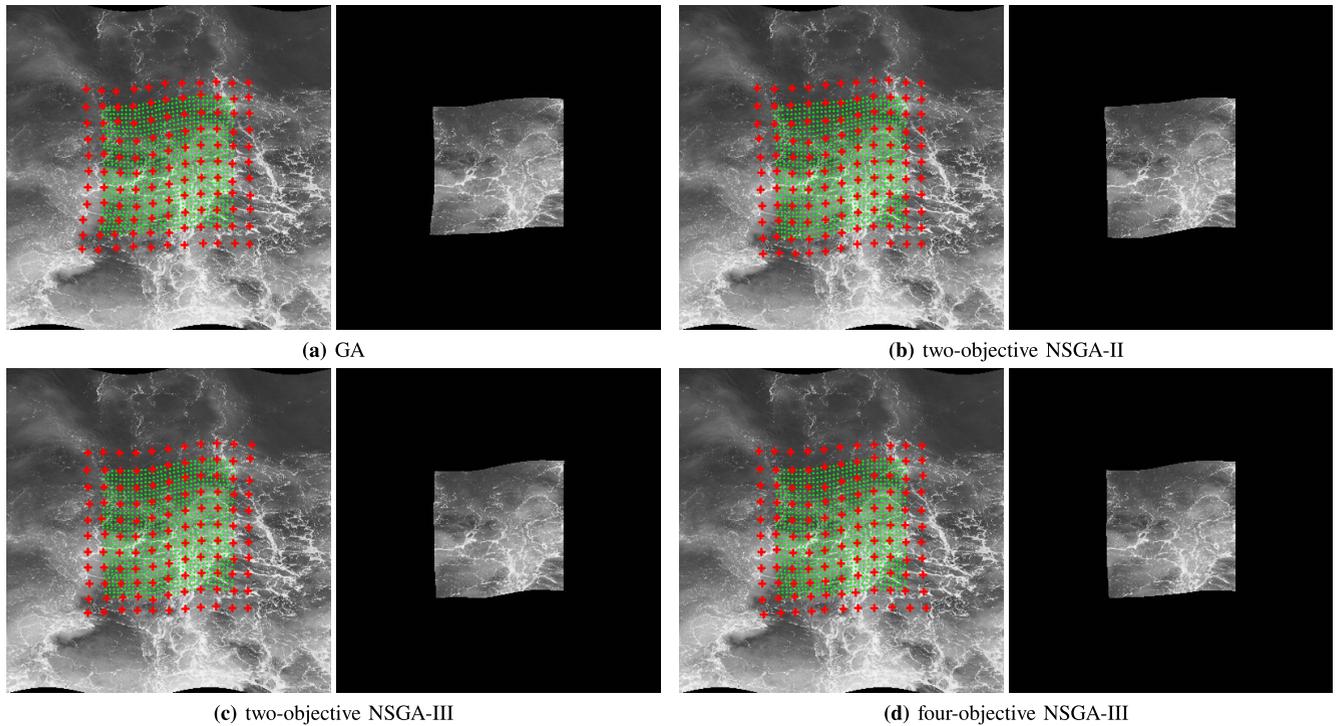
FIGURE 10. Comparison of the results showing maximum MEDE for the sea image with the 11 × 11 lattice and [−10.0, 10.0] range.

experiment, we only manually set several essential parameters and fix other parameters following the default setting throughout the experiment. Specifically, the number of evaluations is set to 10000 and the number of reference points of NSGA-III is set to 100 for the two-objective setting and 120 for the four-objective setting. For fair comparisons, all experiments are executed five times with different random seeds considering probabilistic operations. The initial population P_0^1 for all settings is kept the same with respect to each random seed.

A. COMPARISON WITH SYNTHETIC DATA

To only focus on verifying the estimation ability rather than robustness against noises, we prepare five images in 400 × 400 pixels for generating template images and deformed target images, as shown in Fig. 8. The corresponding template images are obtained by cropping 160 × 160 pixels regions from the center of each image. Eight types of deformations are used for generating a deformed target image, the parameters are:

- **Deformation (2 types)**: an image is deformed into a wavy shape by moving the control points according to a sine curve. In addition to vertical-only displacements, a combination of both vertical and horizontal displacements is used.
- **Lattice size (2 types)**: 7 × 7 and 11 × 11 lattices are used for the bottom image of the pyramid.
- **Ranges of decision variables (2 types)**: we use [−5.0, 5.0] and [−10.0, 10.0] as the range of each deci-

sion variable. Ranges are limited to make sure that control points do not overlap with each other spatially.

As a result, there are 40 (i.e., 5 images × 8 types of parameter settings) types of deformed target images in total. These images are generated by using backward warping, and hence the ground truth of displacements can be obtained. In this section, we evaluate each result based on not only the root mean square error (RMSE) but also the mean Euclidean distance error (MEDE). RMSE is calculated from all the pixels between the deformed template image and the sampling region ω'. MEDE is calculated based on all the ground truth displacements.

1) RESULTS OF VERTICAL WAVY IMAGES

Comprehensive numerical results of the best solutions with respect to the vertical wavy images are summarized in Table 1. For each combination of image setting and algorithm, the minimum, maximum, and average values based on five random trials are investigated. As can be observed by comparing the four algorithms in terms of RMSE and MEDE, it is clear that the two-objective algorithms can achieve better results in most cases. Examples of visual results are shown in Fig 9, from which we can observe that these methods can estimate deformation parameters correctly for all the test images. By contrast, GA with single-objective achieves the best result only once in terms of the average RMSE and zero times in terms of the average MEDE value. The accuracy of GA can degrade significantly, e.g., 11 × 11 lattice with [−10.0, 10.0] range as illustrated in Fig. 10a. These observations verify the

TABLE 2. Comparative results of the final solutions after post-processing with respect to the *vertical wavy deformation*. Each cell consists of three values which represent the minimum, maximum, and average values based on five random trials from top to bottom. The values smaller than the corresponding values in Table 1 with the same settings are shown in Italics. The minimum value in terms of RMSE and MEDE with respect to each row, including the results in Table 1, is emphasized in bold.

Image	Lattice size	Decision variable range	RMSE Two objective NSGA-II	RMSE Two objective NSGA-III	RMSE Four objective NSGA-III	MEDE Two objective NSGA-II	MEDE Two objective NSGA-III	MEDE Four objective NSGA-III
Plant	7 × 7	[−5.0, 5.0]	*1.98E+00* 3.04E+00 2.45E+00	2.40E+00 2.85E+00 2.54E+00	2.37E+00 3.40E+00 2.80E+00	**9.98E-02** 1.31E-01 1.15E-01	1.08E-01 1.38E-01 1.20E-01	*1.16E-01* *1.75E-01* *1.37E-01*
		[−10.0, 10.0]	4.50E+00 5.61E+00 5.12E+00	**4.29E+00** 6.71E+00 *5.15E+00*	4.98E+00 6.81E+00 5.70E+00	2.28E-01 **2.99E-01** 2.67E-01	2.23E-01 3.42E-01 2.70E-01	2.59E-01 3.45E-01 3.01E-01
	11 × 11	[−5.0, 5.0]	*2.55E+00* 4.27E+00 3.23E+00	**2.25E+00** *3.51E+00* *3.03E+00*	3.07E+00 3.79E+00 3.38E+00	1.37E-01 2.16E-01 1.76E-01	**1.04E-01** **1.85E-01** **1.57E-01**	1.58E-01 1.96E-01 1.74E-01
		[−10.0, 10.0]	5.21E+00 9.51E+00 6.73E+00	5.27E+00 *7.49E+00* *6.61E+00*	6.04E+00 8.34E+00 6.87E+00	2.86E-01 7.39E-01 4.53E-01	3.48E-01 5.27E-01 4.29E-01	*3.25E-01* **5.03E-01** **4.26E-01**
Sea	7 × 7	[−5.0, 5.0]	3.36E+00 **3.78E+00** **3.61E+00**	**3.19E+00** 4.18E+00 3.71E+00	4.11E+00 5.00E+00 4.53E+00	9.53E-02 1.23E-01 1.11E-01	1.00E-01 *1.21E-01* **1.10E-01**	1.26E-01 1.54E-01 1.41E-01
		[−10.0, 10.0]	7.73E+00 9.29E+00 **8.32E+00**	8.15E+00 9.69E+00 8.94E+00	8.45E+00 1.11E+01 9.45E+00	2.34E-01 2.67E-01 2.46E-01	2.43E-01 3.40E-01 3.04E-01	2.55E-01 3.75E-01 3.09E-01
	11 × 11	[−5.0, 5.0]	4.00E+00 7.43E+00 5.21E+00	**3.69E+00** 5.68E+00 4.58E+00	4.46E+00 **5.59E+00** 4.92E+00	1.09E-01 2.34E-01 1.56E-01	**1.03E-01** 1.86E-01 **1.39E-01**	*1.27E-01* **1.85E-01** 1.43E-01
		[−10.0, 10.0]	6.72E+00 9.33E+00 **7.83E+00**	**6.38E+00** 1.07E+01 8.90E+00	7.84E+00 1.11E+01 9.12E+00	2.45E-01 **4.05E-01** **3.00E-01**	**2.42E-01** 5.03E-01 3.54E-01	2.55E-01 5.94E-01 3.60E-01
Rag	7 × 7	[−5.0, 5.0]	3.90E+00 5.02E+00 4.25E+00	3.17E+00 *4.68E+00* **3.77E+00**	4.08E+00 5.18E+00 4.67E+00	9.39E-02 1.29E-01 1.08E-01	8.83E-02 *1.16E-01* **9.79E-02**	*1.03E-01* 1.44E-01 1.21E-01
		[−10.0, 10.0]	**7.59E+00** 9.32E+00 **8.44E+00**	8.27E+00 9.16E+00 8.65E+00	8.78E+00 9.85E+00 9.45E+00	**2.08E-01** 2.58E-01 2.22E-01	2.17E-01 2.38E-01 2.27E-01	2.33E-01 *2.57E-01* 2.49E-01
	11 × 11	[−5.0, 5.0]	**4.04E+00** **5.47E+00** **4.70E+00**	4.86E+00 5.95E+00 5.33E+00	4.72E+00 5.82E+00 5.23E+00	**1.02E-01** *1.43E-01* **1.18E-01**	1.21E-01 1.62E-01 1.38E-01	1.24E-01 **1.41E-01** 1.31E-01
		[−10.0, 10.0]	*8.14E+00* 1.35E+01 **9.66E+00**	**6.86E+00** 1.86E+01 1.03E+01	9.25E+00 **1.14E+01** 1.02E+01	2.33E-01 3.93E-01 **2.73E-01**	**1.95E-01** 5.86E-01 2.99E-01	2.52E-01 **3.42E-01** 2.89E-01
Alphabet	7 × 7	[−5.0, 5.0]	3.58E+00 4.86E+00 **4.11E+00**	3.57E+00 4.49E+00 4.14E+00	4.30E+00 5.02E+00 4.72E+00	1.08E-01 *1.34E-01* 1.21E-01	*1.07E-01* 1.29E-01 1.20E-01	*1.18E-01* *1.55E-01* *1.40E-01*
		[−10.0, 10.0]	**7.28E+00** 9.51E+00 **8.10E+00**	7.85E+00 **8.62E+00** 8.22E+00	8.84E+00 1.02E+01 9.58E+00	**1.93E-01** 2.73E-01 2.31E-01	2.35E-01 2.90E-01 2.54E-01	2.57E-01 3.24E-01 2.91E-01
	11 × 11	[−5.0, 5.0]	4.00E+00 5.04E+00 4.64E+00	**3.86E+00** *4.95E+00* **4.26E+00**	5.01E+00 6.51E+00 5.73E+00	*1.34E-01* 1.81E-01 1.52E-01	**1.31E-01** **1.61E-01** **1.39E-01**	1.79E-01 2.04E-01 1.93E-01
		[−10.0, 10.0]	8.45E+00 **1.10E+01** 9.56E+00	**7.47E+00** 1.13E+01 9.68E+00	9.00E+00 1.20E+01 1.10E+01	3.51E-01 4.50E-01 3.88E-01	**2.14E-01** **4.22E-01** **3.66E-01**	3.46E-01 4.66E-01 4.02E-01
Toad	7 × 7	[−5.0, 5.0]	*1.55E+00* 2.52E+00 1.93E+00	**1.21E+00** 2.39E+00 **1.91E+00**	1.91E+00 2.58E+00 2.24E+00	**1.01E-01** **1.38E-01** **1.21E-01**	1.11E-01 1.49E-01 1.31E-01	1.25E-01 1.55E-01 1.39E-01
		[−10.0, 10.0]	2.99E+00 4.29E+00 3.68E+00	**2.44E+00** 3.62E+00 **3.12E+00**	3.78E+00 5.14E+00 4.64E+00	2.56E-01 **3.35E-01** **2.91E-01**	2.79E-01 3.52E-01 3.14E-01	**2.54E-01** 3.52E-01 3.03E-01
	11 × 11	[−5.0, 5.0]	*1.86E+00* **2.50E+00** 2.12E+00	**1.51E+00** 2.64E+00 2.13E+00	1.90E+00 3.40E+00 2.60E+00	2.33E-01 2.88E-01 2.65E-01	2.08E-01 3.04E-01 2.45E-01	**1.76E-01** **2.80E-01** **2.36E-01**
		[−10.0, 10.0]	2.92E+00 3.67E+00 **3.20E+00**	**2.79E+00** 4.72E+00 3.64E+00	3.18E+00 4.81E+00 3.94E+00	4.26E+00 7.59E-01 5.27E-01	4.71E-01 7.87E-01 5.70E-01	**4.05E-01** **4.80E-01** **4.45E-01**

effectiveness of the multi-objective approaches. In addition, we can observe that four-objective NSGA-III gets poorer results than two-objective algorithms. Focusing on average values of both evaluation criteria, four-objective NSGA-III outperforms others for zero times regarding the RMSE and only once regarding the MEDE.

The final solutions after post-processing on Pareto optimal solutions obtained by multi-objective algorithms are
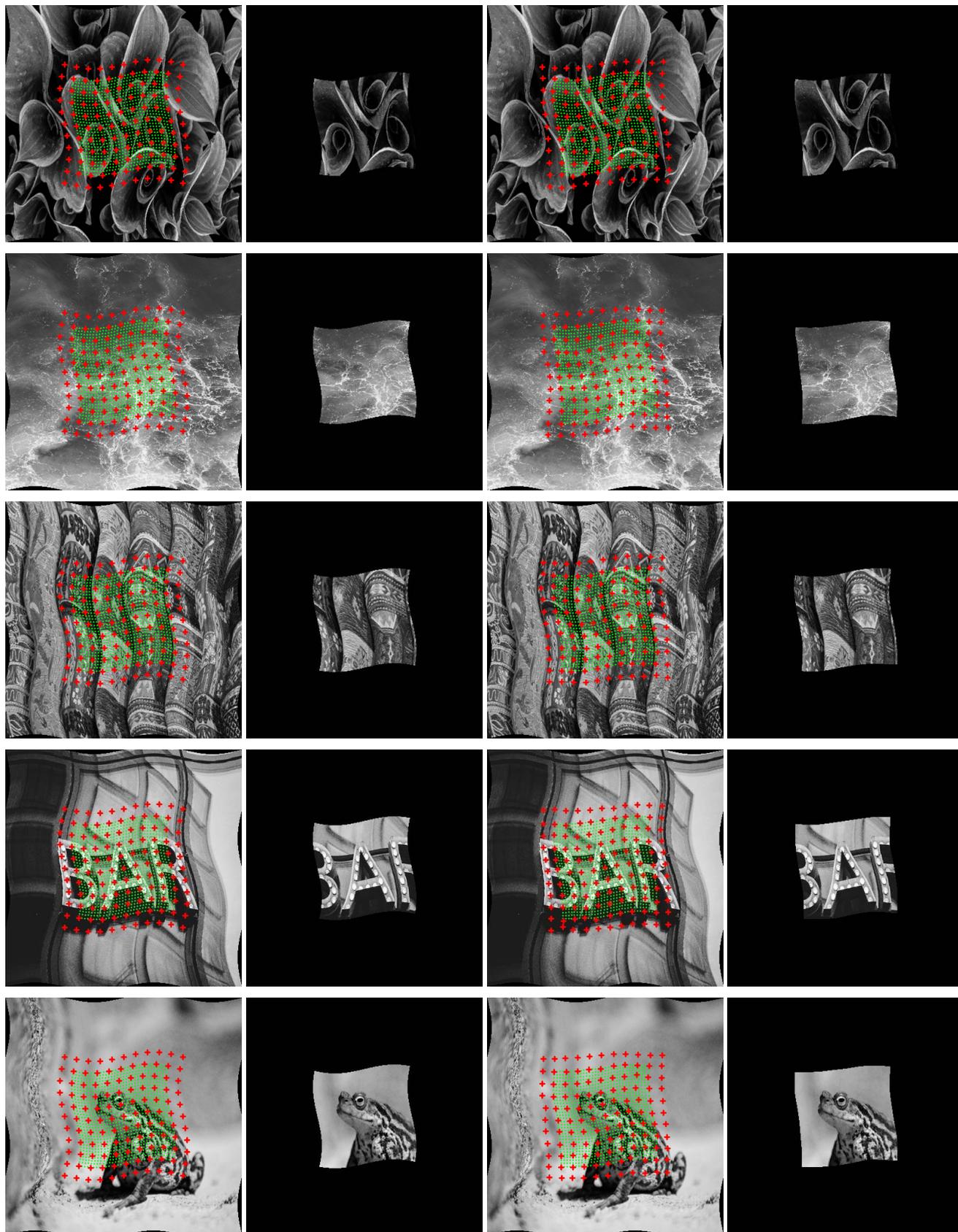
TABLE 3. Comparative results of the best solutions with respect to the *vertical and horizontal wavy deformation*. Each cell consists of three values which represent the minimum, maximum, and average values based on five random trials from top to bottom. The best results in terms of RMSE and MEDE with respect to each row are emphasized in bold.

Image	Lattice size	Decision variable range	RMSE				MEDE			
			GA	Two objective NSGA-II	Two objective NSGA-III	Four objective NSGA-III	GA	Two objective NSGA-II	Two objective NSGA-III	Four objective NSGA-III
Plant	7 × 7	[−5.0, 5.0]	5.13E+00	**4.38E+00**	4.49E+00	4.95E+00	2.29E-01	**1.89E-01**	2.06E-01	2.12E-01
			5.58E+00	6.89E+00	5.65E+00	7.17E+00	2.64E-01	3.45E-01	**2.55E-01**	3.27E-01
			5.39E+00	5.45E+00	**5.01E+00**	5.80E+00	2.44E-01	2.44E-01	**2.30E-01**	2.66E-01
		[−10.0, 10.0]	**9.42E+00**	9.56E+00	9.51E+00	1.11E+01	4.94E-01	5.27E-01	**4.86E-01**	6.19E-01
			1.05E+01	1.08E+01	1.16E+01	1.25E+01	6.06E-01	**5.94E-01**	6.96E-01	6.90E-01
			1.01E+01	1.01E+01	1.05E+01	1.16E+01	**5.53E-01**	5.66E-01	5.71E-01	6.57E-01
	11 × 11	[−5.0, 5.0]	8.40E+00	6.10E+00	6.28E+00	**5.13E+00**	3.32E-01	3.57E-01		**2.55E-01**
			9.95E+00	8.99E+00	8.92E+00	**8.79E+00**	5.76E-01	5.43E-01	**5.18E-01**	5.27E-01
			8.95E+00	7.82E+00	7.50E+00	**7.37E+00**	5.41E-01	4.48E-01	4.36E-01	**4.14E-01**
		[−10.0, 10.0]	1.02E+01	**8.32E+00**	9.84E+00	9.94E+00	6.07E-01	**5.18E-01**	6.63E-01	6.22E-01
			2.28E+01	1.22E+01	1.34E+01	**1.17E+01**	3.00E+00	7.95E-01	9.62E-01	**7.07E-01**
			1.36E+01	**9.97E+00**	1.14E+01	1.05E+01	1.20E+00	**6.44E-01**	7.45E-01	6.80E-01
Sea	7 × 7	[−5.0, 5.0]	**5.32E+00**	5.41E+00	5.65E+00	7.00E+00	**1.60E-01**	1.76E-01	1.62E-01	2.06E-01
			6.29E+00	6.78E+00	6.40E+00	8.50E+00	**1.98E-01**	2.21E-01	2.15E-01	3.12E-01
			5.78E+00	6.15E+00	6.07E+00	7.54E+00	**1.72E-01**	1.96E-01	1.97E-01	2.55E-01
		[−10.0, 10.0]	**1.09E+01**	1.14E+01	1.23E+01	1.26E+01	4.94E-01	**4.90E-01**	5.35E-01	5.90E-01
			1.36E+01	**1.32E+01**	1.46E+01	1.45E+01	6.47E-01	**5.77E-01**	9.34E-01	8.52E-01
			1.25E+01	**1.21E+01**	1.34E+01	1.32E+01	5.78E-01	**5.28E-01**	6.60E-01	6.60E-01
	11 × 11	[−5.0, 5.0]	6.38E+00	6.85E+00	**5.99E+00**	7.31E+00	2.73E-01	2.81E-01	**2.43E-01**	2.70E-01
			9.62E+00	**9.61E+00**	9.90E+00	1.00E+01	4.97E-01	4.65E-01	5.17E-01	**4.62E-01**
			8.07E+00	8.41E+00	8.82E+00	8.47E+00	3.73E-01	3.90E-01	4.33E-01	**3.43E-01**
		[−10.0, 10.0]	**1.17E+01**	1.22E+01	1.29E+01	1.31E+01	1.02E+00	**6.86E-01**	9.20E-01	8.90E-01
			1.74E+01	1.68E+01	**1.52E+01**	1.59E+01	**1.56E+00**	1.94E+00	1.57E+00	1.78E+00
			1.48E+01	1.48E+01	**1.37E+01**	1.45E+01	1.31E+00	1.39E+00	**1.14E+00**	1.29E+00
Rag	7 × 7	[−5.0, 5.0]	**6.54E+00**	6.68E+00	7.07E+00	7.23E+00	1.74E-01	**1.59E-01**	1.70E-01	2.02E-01
			1.09E+01	**8.72E+00**	1.04E+01	1.08E+01	3.16E-01	**2.18E-01**	3.08E-01	3.01E-01
			8.38E+00	**7.42E+00**	8.27E+00	9.08E+00	2.26E-01	**1.88E-01**	2.15E-01	2.48E-01
		[−10.0, 10.0]	1.30E+01	1.31E+01	**1.29E+01**	1.61E+01	**3.62E-01**	3.68E-01	3.84E-01	4.70E-01
			1.57E+01	1.64E+01	**1.53E+01**	1.85E+01	5.19E-01	5.22E-01	**4.93E-01**	6.77E-01
			1.43E+01	1.45E+01	1.45E+01	1.76E+01	4.37E-01	**4.36E-01**	4.41E-01	5.68E-01
	11 × 11	[−5.0, 5.0]	1.33E+01	**1.15E+01**	1.29E+01	1.37E+01	4.61E-01	**3.73E-01**	4.54E-01	5.07E-01
			1.74E+01	**1.50E+01**	1.58E+01	1.63E+01	7.63E-01	**5.45E-01**	5.89E-01	6.04E-01
			1.53E+01	**1.40E+01**	1.47E+01	1.49E+01	5.90E-01	**4.87E-01**	5.27E-01	5.43E-01
		[−10.0, 10.0]	1.66E+01	1.71E+01	**1.55E+01**	1.89E+01	6.91E-01	6.42E-01	**5.06E-01**	6.77E-01
			2.66E+01	3.00E+01	2.86E+01	**2.27E+01**	2.02E+00	2.19E+00	2.22E+00	**9.76E-01**
			2.29E+01	2.18E+01	2.22E+01	**2.02E+01**	1.35E+00	1.11E+00	1.23E+00	**7.90E-01**
Alphabet	7 × 7	[−5.0, 5.0]	5.15E+00	5.20E+00	**4.78E+00**	6.62E+00	2.03E-01	2.13E-01	**1.95E-01**	2.67E-01
			6.23E+00	**5.70E+00**	5.90E+00	8.25E+00	2.48E-01	**2.33E-01**	2.45E-01	3.30E-01
			5.51E+00	5.50E+00	**5.35E+00**	7.10E+00	**2.21E-01**	2.21E-01	2.24E-01	2.88E-01
		[−10.0, 10.0]	**9.87E+00**	1.09E+01	1.06E+01	1.25E+01	**3.92E-01**	4.24E-01	4.19E-01	5.24E-01
			1.18E+01	1.20E+01	1.29E+01	1.39E+01	5.08E-01	**4.85E-01**	5.58E-01	6.40E-01
			1.09E+01	1.13E+01	1.17E+01	1.32E+01	**4.65E-01**	4.67E-01	4.96E-01	5.72E-01
	11 × 11	[−5.0, 5.0]	6.93E+00	5.76E+00	**4.34E+00**	6.92E+00	3.21E-01	2.49E-01	**1.93E-01**	3.15E-01
			8.88E+00	**6.93E+00**	1.08E+01	8.43E+00	3.82E-01	**3.18E-01**	4.62E-01	3.96E-01
			7.52E+00	**6.59E+00**	6.82E+00	7.65E+00	3.49E-01	**2.95E-01**	3.00E-01	3.60E-01
		[−10.0, 10.0]	1.84E+01	1.19E+01	1.08E+01	**1.06E+01**	1.02E+00	5.98E-01	**5.37E-01**	5.77E-01
			2.59E+01	2.26E+01	2.37E+01	**1.48E+01**	1.70E+00	1.44E+00	1.62E+00	**7.67E-01**
			2.20E+01	1.68E+01	1.63E+01	**1.26E+01**	1.29E+00	9.69E-01	9.61E-01	**6.60E-01**
Toad	7 × 7	[−5.0, 5.0]	3.09E+00	3.05E+00	**2.91E+00**	3.13E+00	2.21E-01	2.00E-01	**1.82E-01**	2.29E-01
			3.88E+00	3.79E+00	**3.42E+00**	4.33E+00	3.52E-01	4.26E-01	**2.57E-01**	3.86E-01
			3.62E+00	3.40E+00	**3.05E+00**	3.83E+00	2.72E-01	2.53E-01	**2.41E-01**	3.11E-01
		[−10.0, 10.0]	6.72E+00	5.80E+00	**4.63E+00**	7.48E+00	5.08E-01	**4.70E-01**	5.02E-01	5.93E-01
			7.95E+00	**7.14E+00**	8.59E+00	9.89E+00	7.05E-01	6.39E-01	**5.83E-01**	7.51E-01
			7.48E+00	6.48E+00	**6.16E+00**	8.43E+00	6.13E-01	5.50E-01	**5.46E-01**	6.60E-01
	11 × 11	[−5.0, 5.0]	3.30E+00	2.67E+00	**2.51E+00**	3.36E+00	4.48E-01	**3.18E-01**	3.70E-01	4.17E-01
			3.91E+00	**3.64E+00**	3.73E+00	4.61E+00	7.05E-01	6.74E-01	7.19E-01	**5.13E-01**
			3.54E+00	**3.08E+00**	3.21E+00	3.95E+00	6.02E-01	5.14E-01	4.99E-01	**4.61E-01**
		[−10.0, 10.0]	7.57E+00	6.26E+00	**5.95E+00**	8.72E+00	1.35E+00	8.29E-01	8.65E-01	**8.24E-01**
			9.98E+00	8.55E+00	**7.39E+00**	1.07E+01	1.92E+00	1.39E+00	1.33E+00	**1.03E+00**
			8.70E+00	7.62E+00	**6.56E+00**	9.58E+00	1.57E+00	1.20E+00	1.13E+00	**9.22E-01**

compared in Table 2. As can be observed, the post-processing successfully improves the estimation accuracy in many cases comparing to Table 1. In particular, four-objective NSGA-III benefits most from the post-processing. Focusing on the number of improved results on the average value, four-objective NSGA-III performs the best 17 times on RMSE and 19 times on MEDE, while two-objective NSGA-II performs the best for 15 and 14 times, respectively.

2) RESULTS OF VERTICAL AND HORIZONTAL WAVY IMAGES

The results of the best solutions with respect to the vertical and horizontal wavy images are shown in Table 3. Despite

FIGURE 11. Examples of visual results achieved by multi-objective methods on *horizontal and vertical wavy* images. The 1st~2nd columns are ground truth and 3rd~4th columns are results. Red markers represent the control points, and green dots represent the deformed surface generated by forward warping. Images in the 4th column show the deformed template images with estimated deformation.
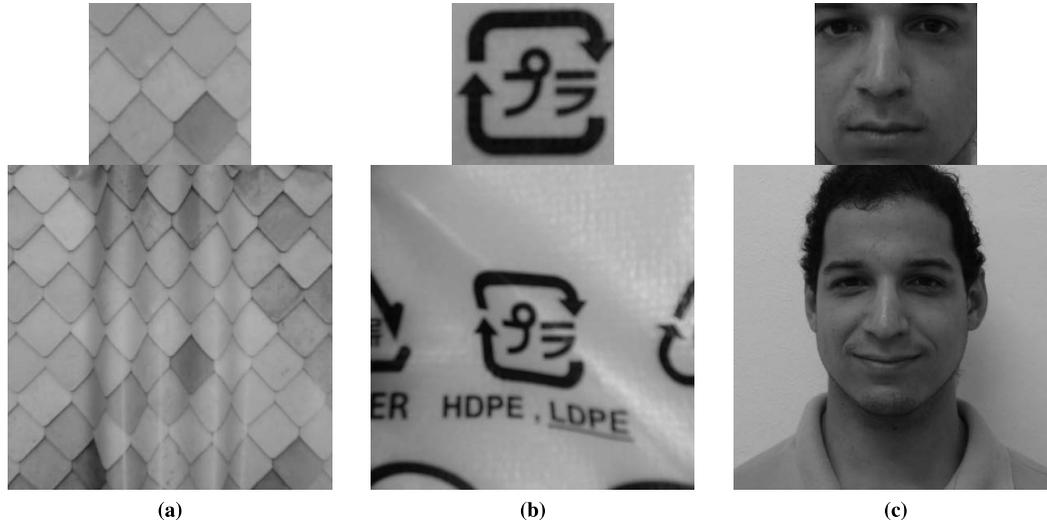
FIGURE 12. Real-world template images (top row) and target images (bottom row). (a) texture, (b) sign, and (c) face.

TABLE 4. The RMSE results of best solutions and post-processed solutions with respect to the *real-world images*. Each cell consists of three values which represent the minimum, maximum, and average values based on five random trials from top to bottom. The values of post-processed solutions which are smaller than the values of the corresponding best solutions are shown in Italics. The best results in terms of RMSE with respect to each row are emphasized in bold.

Image	Lattice size	Decision variable range	GA	Best solution			Post-processed solution		
				Two objective NSGA-II	Two objective NSGA-III	Four objective NSGA-III	Two objective NSGA-II	Two objective NSGA-III	Four objective NSGA-III
Texture	7 × 7	[−20.0, 20.0]	1.23E+01	1.21E+01	1.24E+01	**1.18E+01**	1.23E+01	*1.23E+01*	1.19E+01
			1.29E+01	**1.25E+01**	1.28E+01	1.25E+01	1.25E+01	1.28E+01	*1.25E+01*
			1.25E+01	1.23E+01	1.25E+01	**1.21E+01**	1.24E+01	1.26E+01	1.22E+01
Sign	7 × 7	[−10.0, 10.0]	2.04E+01	1.95E+01	1.95E+01	2.04E+01	1.98E+01	*1.95E+01*	*2.02E+01*
			2.13E+01	**2.03E+01**	2.06E+01	2.07E+01	2.04E+01	2.10E+01	*2.06E+01*
			2.09E+01	2.00E+01	**2.00E+01**	2.05E+01	2.01E+01	2.00E+01	*2.05E+01*
Face	7 × 7	[−20.0, 20.0]	7.55E+00	7.42E+00	**7.37E+00**	7.50E+00	7.47E+00	7.42E+00	7.50E+00
			8.01E+00	7.71E+00	**7.61E+00**	7.87E+00	7.88E+00	7.82E+00	8.08E+00
			7.72E+00	7.58E+00	**7.48E+00**	7.69E+00	7.65E+00	7.56E+00	7.77E+00

more complex deformations, the multi-objective approaches can still achieve good estimation. Several qualitative results are shown in Fig. 11. Comparing with Table 1, we can observe that the best results are irregularly distributed in terms of both evaluation criteria. Nevertheless, two-objective NSGA-II and NSGA-III are still better choices overall. In the case of 7 × 7 lattice, GA can achieve the best result when the search space is small (e.g., the sea image). Four-objective NSGA-III shows good results in the case of 11 × 11 lattice setting, which achieves the best average MEDE for six times out of ten times. Although the number of groups is a hyper-parameter to be handled carefully as mentioned in Sec. VI-A1, our results show the trend that larger number of groups is more effective when dealing with complex and subtle deformations.

B. COMPARISON ON REAL-WORLD IMAGES

The usefulness of the proposed method under real-world scenarios is verified in this section. We use three different pairs of template and target images as shown in Fig. 12.

- Texture (Fig. 12a): the images capture a part of the undeformed/deformed texture printed on a piece of paper. The target image has two vertical bumps. We set the lattice as 7 × 7 and the decision variable range as [−20.0, 20.0].
- Sign (Fig. 12b): the images capture a sign with undeformed/deformed text printed on a piece of wrapping paper. The lattice and decision variable range are set to 7 × 7 lattice and [−10.0, 10.0], respectively.
- Face (Fig. 12c): the template image and the target image show a frontal face with serious expression and smile, respectively. The goal is to obtain deformation parameters that express smiling. We use the 7 × 7 lattice and the [−20.0, 20.0] decision variable range.

The sizes of the template images and target images are set the same as Sec. VI-A. Fig. 12a and Fig. 12b are captured by a web camera and Fig. 12c is extracted from the FEI face

(a) texture

(b) sign

(c) face
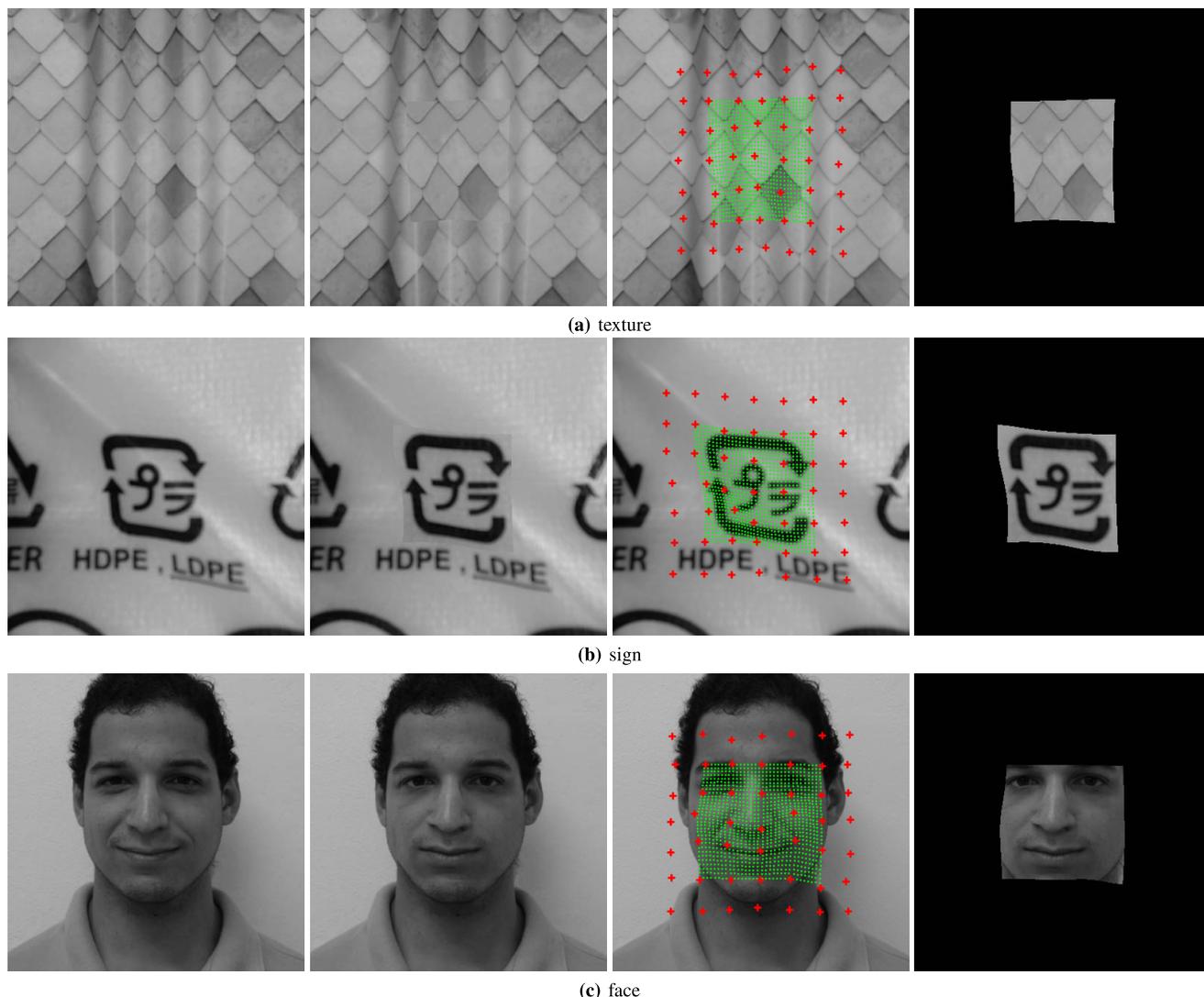
FIGURE 13. Target images (1st column) and estimation results (2nd~4th columns). The 4th column shows the template deformed with estimated deformation. Composite images in the 2nd column are created from the images in the 1st column and 4th column.

database[2] [55] and cropped. Because the ground truth of the real deformations is unknown, we evaluate results only using RMSE.

The results of best solutions and post-processed solutions are shown in Table 4. These results show that multi-objective approaches can outperform the single-objective approach in real-world situations. However, post-processing fails to improve estimation accuracy in a number of cases (e.g., the face image). The estimation results for each image are shown in Fig. 13. It can be observed that the deformed texture image contains some highlight areas and the eye area of the face image is also shadowed. Hence, the deformation results for these corresponding areas show worse accuracy than the other areas. The groups including these areas cannot contribute to the post-processing. As a limitation, the proposed method

---

[2]https://fei.edu.br/~cet/facedatabase.html

suffers from illumination changes due to the characteristics of the similarity measure.

VII. CONCLUSION

In this paper, we proposed a novel deformation estimation method using MOEAs to tackle the conflicts based on the fact that each control point of the deformation model affects a local region rather than a single pixel. Our method casts deformation estimation as a MOP by dividing a template image into several groups consisting of patches with group-wise similarity defined as group-wise objective functions, which can be solved by off-the-shelf MOEAs. To handle large deformations, optimization is run hierarchically following a coarse-to-fine strategy powered by image pyramid and control point mesh subdivision. Besides, a post-processing procedure is proposed to integrate Pareto optimal solutions into a single output, which can improve the estimation

accuracy. The observations from experimental results can be summarized threefold. First, our partitioning approach with two-objective algorithms can obtain deformation parameters more accurately than GA with a single objective. Second, although the four-objective algorithm performs not as well as expected due to a large number of objectives, it shows to be effective in dealing with complex and subtle deformations. Third, the post-processing procedure can improve estimation accuracy in many cases. We can observe the usefulness of the proposed method with real-world images.

The main limitation of our method is that high computational resources are required. As future work, we would like to address this issue by further tuning the hyper-parameters that can reduce the computational cost without degrading the performance. We are also interested in the referenced point distribution of the NSGA-III. A user-supplied setting may be able to focus solutions on regions that are desirable for the post-processing procedure.

REFERENCES

[1] V. Gay-Bellile, A. Bartoli, and P. Sayd, "Direct estimation of nonrigid registrations with image-based self-occlusion reasoning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 1, pp. 87–104, Jan. 2010, doi: 10.1109/TPAMI.2008.265.

[2] X. Cao, J. Yang, J. Zhang, Q. Wang, P.-T. Yap, and D. Shen, "Deformable image registration using a cue-aware deep regression network," *IEEE Trans. Biomed. Eng.*, vol. 65, no. 9, pp. 1900–1911, Sep. 2018, doi: 10.1109/TBME.2018.2822826.

[3] G. Balakrishnan, A. Zhao, M. R. Sabuncu, J. Guttag, and A. V. Dalca, "VoxelMorph: A learning framework for deformable medical image registration," *IEEE Trans. Med. Imag.*, vol. 38, no. 8, pp. 1788–1800, Aug. 2019, doi: 10.1109/TMI.2019.2897538.

[4] D. J. Tan, S. Holzer, N. Navab, and S. Ilic, "Deformable template tracking in 1 ms," in *Proc. Brit. Mach. Vis. Conf.*, 2014, pp. 1–11, doi: 10.5244/C.28.27.

[5] T. Wang, H. Ling, C. Lang, S. Feng, and X. Hou, "Deformable surface tracking by graph matching," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 901–910, doi: 10.1109/ICCV.2019.00099.

[6] Y. Wang, D. McConachie, and D. Berenson, "Tracking partially-occluded deformable objects while enforcing geometric constraints," 2020, *arXiv:2011.00627*.

[7] D. Rueckert, L. I. Sonoda, C. Hayes, D. L. G. Hill, M. O. Leach, and D. J. Hawkes, "Nonrigid registration using free-form deformations: Application to breast MR images," *IEEE Trans. Med. Imag.*, vol. 18, no. 8, pp. 712–721, Aug. 1999, doi: 10.1109/42.796284.

[8] F. P. M. Oliveira and J. M. R. S. Tavares, "Medical image registration: A review," *Comput. Methods Biomech. Biomed. Eng.*, vol. 17, no. 2, pp. 73–93, 2014, doi: 10.1080/10255842.2012.670855.

[9] M. Alexa, "Recent advances in mesh morphing," *Comput. Graph. Forum*, vol. 21, pp. 173–198, Jun. 2002, doi: 10.1111/1467-8659.00575.

[10] U. Scherhag, C. Rathgeb, J. Merkle, R. Breithaupt, and C. Busch, "Face recognition systems under morphing attacks: A survey," *IEEE Access*, vol. 7, pp. 23012–23026, 2019, doi: 10.1109/ACCESS.2019.2899367.

[11] T. W. Sederberg and S. R. Parry, "Free-form deformation of solid geometric models," *ACM SIGGRAPH Comput. Graph.*, vol. 20, no. 4, pp. 151–160, 1986, doi: 10.1145/15886.15903.

[12] W. M. Hsu, J. F. Hughes, and H. Kaufman, "Direct manipulation of free-form deformations," *ACM SIGGRAPH Comput. Graph.*, vol. 26, no. 2, pp. 177–184, Jul. 1992, doi: 10.1145/142920.134036.

[13] W. R. Crum, T. Hartkens, and D. L. G. Hill, "Non-rigid image registration: Theory and practice," *Brit. J. Radiol.*, vol. 77, no. 2, pp. S140–S153, Dec. 2004, doi: 10.1259/bjr/25329214.

[14] J. Zhu, M. R. Lyu, and T. S. Huang, "A fast 2D shape recovery approach by fusing features and appearance," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 7, pp. 1210–1224, Jul. 2009, doi: 10.1109/TPAMI.2008.151.

[15] D. Pizarro and A. Bartoli, "Feature-based deformable surface detection with self-occlusion reasoning," *Int. J. Comput. Vis.*, vol. 97, no. 1, pp. 54–70, Mar. 2012, doi: 10.1007/s11263-011-0452-0.

[16] Y. Wu, Y. Ijiri, and M.-H. Yang, "Multiple non-rigid surface detection and registration," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 1992–1999, doi: 10.1109/ICCV.2013.249.

[17] J. Pilet, V. Lepetit, and P. Fua, "Fast non-rigid surface detection, registration and realistic augmentation," *Int. J. Comput. Vis.*, vol. 76, no. 2, pp. 109–122, Feb. 2008, doi: 10.1007/s11263-006-0017-9.

[18] M. Salzmann and P. Fua, "Linear local models for monocular reconstruction of deformable surfaces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 931–944, May 2011, doi: 10.1109/TPAMI.2010.158.

[19] Q.-H. Tran, T.-J. Chin, G. Carneiro, M. S. Brown, and D. Suter, "In defence of RANSAC for outlier rejection in deformable registration," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 274–287, doi: 10.1007/978-3-642-33765-9_20.

[20] D. T. Ngo, J. Östlund, and P. Fua, "Template-based monocular 3D shape recovery using Laplacian meshes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 1, pp. 172–187, Jan. 2016, doi: 10.1109/TPAMI.2015.2435739.

[21] A. Bartoli and A. Zisserman, "Direct estimation of non-rigid registration," in *Proc. Brit. Mach. Vis. Conf.*, 2004, pp. 92.1–92.10, doi: 10.5244/C.18.92.

[22] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, pp. 91–110, Dec. 2004, doi: 10.1023/B:Visi.0000029664.99615.94.

[23] J. M. Morel and G. Yu, "ASIFT: A new framework for fully affine invariant image comparison," *SIAM J. Imag. Sci.*, vol. 2, no. 2, pp. 438–469, 2009, doi: 10.1137/080732730.

[24] V. Gay-Bellile, M. Perriollat, A. Bartoli, and P. Sayd, "Image registration by combining thin-plate splines with a 3D morphable model," in *Proc. Int. Conf. Image Process.*, Oct. 2006, pp. 1069–1072, doi: 10.1109/ICIP.2006.312740.

[25] E. Malis, "An efficient unified approach to direct visual tracking of rigid and deformable surfaces," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2007, pp. 2729–2734, doi: 10.1109/IROS.2007.4398990.

[26] A. Hilsmann, D. C. Schneider, and P. Eisert, "Realistic cloth augmentation in single view video under occlusions," *Comput. Graph.*, vol. 34, no. 5, pp. 567–574, Oct. 2010, doi: 10.1016/j.cag.2010.05.015.

[27] C. Zhang and T. Akashi, "Fast affine template matching over Galois field," in *Proc. BMVC*, 2015, pp. 121.1–121.11.

[28] C. Zhang and T. Akashi, "Robust projective template matching," *IEICE Trans. Inf. Syst.*, vol. 99, no. 9, pp. 2341–2350, 2016, doi: 10.1587/transinf.2016EDP7038.

[29] S. Klein, M. Staring, and J. P. W. Pluim, "Evaluation of optimization methods for nonrigid medical image registration using mutual information and B-splines," *IEEE Trans. Image Process.*, vol. 16, no. 12, pp. 2879–2890, Dec. 2007, doi: 10.1109/TIP.2007.909412.

[30] T. Alderliesten, J.-J. Sonke, and P. A. Bosman, "Multi-objective optimization for deformable image registration: Proof of concept," *Proc. SPIE*, vol. 8314, Feb. 2012, pp. 594–600, doi: 10.1117/12.911268.

[31] K. Pirpinia, P. A. Bosman, J.-J. Sonke, M. van Herk, and T. Alderliesten, "Evolutionary multi-objective meta-optimization of deformation and tissue removal parameters improves the performance of deformable image registration of pre- and post-surgery images," *Proc. SPIE*, vol. 10949, Mar. 2019, pp. 838–848, doi: 10.1117/12.2512760.

[32] T. Nakane, T. Akashi, X. Lu, and C. Zhang, "A probabilistic bitwise genetic algorithm for B-spline based image deformation estimation," in *Proc. Genetic Evol. Comput. Conf. Companion*, Jul. 2019, pp. 300–301.

[33] T. Nakane, N. Bold, H. Sun, X. Lu, T. Akashi, and C. Zhang, "Application of evolutionary and swarm optimization in computer vision: A literature survey," *IPSJ Trans. Comput. Vis. Appl.*, vol. 12, no. 1, pp. 1–34, Dec. 2020, doi: 10.1186/s41074-020-00065-9.

[34] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2000, doi: 10.1109/4235.996017.

[35] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength Pareto evolutionary algorithm for multiobjective optimization," in *Evolutionary Methods for Design, Optimization and Control With Applications to Industrial Problems*. Barcelona, Spain: International Center for Numerical Methods in Engineering, 2002, pp. 95–100.

[36] D. W. Corne, N. R. Jerram, J. D. Knowles, and M. J. Oates, "PESA-II: Region-based selection in evolutionary multiobjective optimization," in *Proc. 3rd Annu. Conf. Genetic Evol. Comput.*, 2001, pp. 283–290, doi: 10.5555/2955239.2955289.

[37] V. L. Vachhani, V. K. Dabhi, and H. B. Prajapati, "Survey of multi objective evolutionary algorithms," in *Proc. Int. Conf. Circuits, Power Comput. Technol.*, Mar. 2015, pp. 1–9, doi: 10.1109/ICCPCT.2015.7159422.

[38] M. Garza-Fabre, G. T. Pulido, and C. A. C. Coello, "Ranking methods for many-objective optimization," in *Proc. Mex. Int. Conf. Artif. Intell.*, 2009, pp. 633–645, doi: 10.1007/978-3-642-05258-3_56.

[39] B. Li, J. Li, K. Tang, and X. Yao, "Many-objective evolutionary algorithms: A survey," *ACM Comput. Surv.*, vol. 48, no. 1, pp. 1–35, Sep. 2015, doi: 10.1145/2792984.

[40] D. K. Saxena, J. A. Duro, A. Tiwari, K. Debm, and Q. Zhang, "Objective reduction in many-objective optimization: Linear and nonlinear algorithms," *IEEE Trans. Evol. Comput.*, vol. 17, no. 1, pp. 77–99, Feb. 2013, doi: 10.1109/TEVC.2012.2185847.

[41] H. Wang and X. Yao, "Objective reduction based on nonlinear correlation information entropy," *Soft Comput.*, vol. 20, no. 6, pp. 2393–2407, 2016, doi: 10.1007/s00500-015-1648-y.

[42] J. Bader and E. Zitzler, "HypE: An algorithm for fast hypervolume-based many-objective optimization," *Evol. Comput.*, vol. 19, no. 1, pp. 45–76, Mar. 2011, doi: 10.1162/EVCO_a_00009.

[43] B. Li, K. Tang, J. Li, and X. Yao, "Stochastic ranking algorithm for many-objective optimization based on multiple indicators," *IEEE Trans. Evol. Comput.*, vol. 20, no. 6, pp. 924–938, Dec. 2016, doi: 10.1109/TEVC.2016.2549267.

[44] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, Dec. 2007, doi: 10.1109/TEVC.2007.892759.

[45] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach. Part I: Solving problems with box constraints," *IEEE Trans. Evol. Comput.*, vol. 18, no. 4, pp. 577–601, Aug. 2013, doi: 10.1109/TEVC.2013.2281535.

[46] H. Xu, W. Zeng, D. Zhang, and X. Zeng, "MOEA/HD: A multiobjective evolutionary algorithm based on hierarchical decomposition," *IEEE Trans. Cybern.*, vol. 49, no. 2, pp. 517–526, Feb. 2019, doi: 10.1109/TCYB.2017.2779450.

[47] Y. Zhang, G.-G. Wang, K. Li, W.-C. Yeh, M. Jian, and J. Dong, "Enhancing MOEA/D with information feedback models for large-scale many-objective optimization," *Inf. Sci.*, vol. 522, pp. 1–16, Jun. 2020, doi: 10.1016/j.ins.2020.02.066.

[48] Y. Yuan, H. Xu, B. Wang, and X. Yao, "A new dominance relation-based evolutionary algorithm for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 20, no. 1, pp. 16–37, Feb. 2016, doi: 10.1109/TEVC.2015.2420112.

[49] Z. Cui, J. Zhang, Y. Chang, X. Cai, and W. Zhang, "Improved NSGA-III with selection-and-elimination operator," *Swarm Evol. Comput.*, vol. 49, pp. 23–33, Sep. 2019, doi: 10.1016/j.swevo.2019.05.011.

[50] K. Li, K. Deb, Q. Zhang, and S. Kwong, "An evolutionary many-objective optimization algorithm based on dominance and decomposition," *IEEE Trans. Evol. Comput.*, vol. 19, no. 5, pp. 694–716, Oct. 2015, doi: 10.1109/TEVC.2014.2373386.

[51] S. Lee, G. Wolberg, and S. Y. Shin, "Scattered data interpolation with multilevel B-splines," *IEEE Trans. Vis. Comput. Graph.*, vol. 3, no. 3, pp. 228–244, Jul.-Sep. 1997, doi: 10.1109/2945.620490.

[52] L. A. Schwarz, "Non-rigid registration using free-form deformations," Ph.D. dissertation, Fakultät für Informatik, Technische Universität München, Munich, Germany, 2007.

[53] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI, USA: Univ. Michigan Press, 1975.

[54] E. Catmull and J. Clark, "Recursively generated B-spline surfaces on arbitrary topological meshes," *Comput.-Aided Des.*, vol. 10, no. 6, pp. 350–355, 1978, doi: 10.1016/0010-4485(78)90110-0.

[55] C. E. Thomaz and G. A. Giraldi, "A new ranking method for principal components analysis and its application to face image analysis," *Image Vis. Comput.*, vol. 28, no. 6, pp. 902–913, Jun. 2010, doi: 10.1016/j.imavis.2009.11.005.

TAKUMI NAKANE (Graduate Student Member, IEEE) received the B.E. and M.E. degrees from the University of Fukui, Japan, in 2017 and 2020, respectively, where he is currently pursuing the Ph.D. degree in advanced interdisciplinary science and technology with the Graduate School of Engineering. His research interests include evolutionary computing and vision-based optimization problems. More information can be found at: https://sites.google.com/view/t-nakane.

HAORAN XIE (Member, IEEE) received the Ph.D. degree from the Japan Advanced Institute of Science and Technology (JAIST), in 2015. After that, he worked with the Computer Science Department, The University of Tokyo, from 2015 to 2018, and was an Assistant Professor with the JAIST, from 2018 to 2020. Since 2020, he has been a Senior Lecturer at the JAIST. His main research interests include interactive computer graphics and user interface. More information can be found at: http://x-interface.com.

CHAO ZHANG (Member, IEEE) received the Ph.D. degree from Iwate University, Japan, in March 2017. He is currently a Senior Lecturer at the Department of Engineering, University of Fukui, Japan. His research interests include computer vision, computer graphics, and evolutionary computing, mainly focused on applying optimization methods to solve visual computing problems. More information can be found at: http://www.labzhang.com/.

• • •