2D Image registration using iterated local search metaheuristic

Oscar Cordón García[1], Sergio Damas Arroyo[2], Nicolás Pérez de la Blanca Capilla[1], Sara Fernández Vidal[3], and Eric Bardinet[3]

[1] Department of Computer Science and Artificial Intelligence, University of Granada, 18071 Granada, Spain. {ocordon,nicolas}@decsai.ugr.es
[2] Department of Software Engineering, University of Granada, 18071 Granada, Spain. sdamas@ugr.es
[3] INRIA, Epidaure Project, Sophia Antipolis, France. {ebard,sfernand}@sophia.inria.fr

Abstract. The ability to establish a mapping between the information of two different images and estimate the geometrical transform it is supposed it has been applied, are two open problems in computer vision. Indeed, it is a crucial task for a wide range of applications. In this work we try to take advantage of the information we can infer from the skeleton of an image. Then, we define a global optimization function holding both problems. We face such a complex optimization problem using a well known metaheuristic: iterated local search.

1 Introduction

Image registration is a fundamental task in image processing used to finding a correspondence (or transformation) among two or more pictures taken under different conditions: at different times, using different sensors, from different viewpoints, or a combination of them. Over the years, registration have been applied to a broad range of situations from remote sensing to medical images or artificial vision and different techniques have been independently studied resulting in a large body of research (in [5], a clear classification of different registration techniques and applications can be found).

In the last few years, a new family of search and optimization algorithms have arised based on extending basic heuristic methods by including them into an iterative framework augmenting its explorattion capabilities. These group of advanced approximate algorithms have received the name of *metaheuristics* and a deep sumary on the different ones existing is to be found in [3].

In recent literature, we can find different aproaches to the matching and registration problems ([15], [10]) from the metaheuristics point of view. In this work, we try to exploit the benefits of applying the *Iterated Local Search* (ILS) metaheuristic [13] to solve registration and our contributions are related to the fact of jointly solving matching and registration transform problems using skeleton derived information.

To do so, in section 2 we present the concept of skeleton and medial axis transform in the field of shape analysis in computer vision and the way we can use them to generate an object partition. In section 3 we give a brief overview of the concept of image registration and two important methods to understand our work. Next, section 4 shows the way we can apply ILS to the registration problem. In section 5,we expose different results we have achieved applying our method. Finally,in section 6 we review the work we've done and future improvements to be considered.

2 Shape analysis in computer vision: the medial axis transform

2.1 Basics

The input to a typical image processing and analysis system is a gray-scale or colour image of a scene containing the objects of interest. In order to understand the contents of the scene, it is necessary to recognize the objects located in it. The shape of the object is a binary image representing its extent. The shape can be thought of as a silhouette of the object. There are many imaging applications where image analysis can be reduced to the analysis of shapes (e.g. organs, cells, machine parts, characters).

Classification of shape analysis techniques can be done attending to different criteria and is beyond the scope of this paper. For an in depth discussion see [11] where a review of a variety of methods is presented. We will focus on a special shape description method: the skeleton and medial axis transform of an object.

Intuitively, the skeleton of an object is the set of points which are equidistant from at least two points of the object boundary (there is a part of the skeleton inside the object and another part outside, but we do not take into account the latter part). The first attempts to a formal definition of the skeleton were due to Blum [4] and Calabi [6]. Blum proposed the grassfire analogy: the skeleton consists of the points where different firefronts intersect, or quench points. He also proposed another definition, widely used later: if one considers the elevation surface (hypersurface in the 3D case) obtained from the distance transform of the object, which for each point inside the object gives the distance to the closest boundary point, the skeleton is the set of points for which there is a discontinuity of the distance map derivative. As the closest boundary point, P, to an object point, M, is known to be the orthogonal projection of M on the boundary, this definition is obviously equivalent to the intuitive one. Indeed, since a point M of the skeleton has at least two projections on the boundary, M is equidistant to almost two boundary points.

In order to formalize the notion of skeleton, in [6] the problem is analyzed from a topological point of view: the skeleton is defined as the set of the object maximal disk (maximum balls in the 3D case) centers. He also proved that the notions of quench points and maximal disk centers are equivalent.

Therefore, in the continuous case, the skeleton of an object is a set with no thickness, that is to say a set that only contains balls of radius zero. The

skeleton of a 2D object will look like a graph placed in the middle of the object and composed of pieces of courbes, called *skeleton branches*.

The medial axis is defined as a set containing the skeleton points and the distance vectors joining each skeletal point to its closest boundary point. Therefore, retrieving the object shape from its medial axis is straightforward. The medial axis is a more complete representation of the object than the skeleton. Indeed, objects with different shapes can yield the same skeleton (if, as was said before, we do not consider the part of the skeleton outside the object), and only the medial axis will differ and allow to distinguish between these objects.

Let us now introduce some notations. First, let X be a digital object. Its medial axis is denoted: $\{SK(X), \boldsymbol{\rho}_{SK}\}$, where $SK(X)$ is the skeleton of X and $\boldsymbol{\rho}$ is the distance map of X (inside X) defined by:

$$\mathbb{Z}^n \longrightarrow \mathbb{R}^+ \; ; \; M \longmapsto \boldsymbol{\rho}(M) = d(M, \overline{X}) = \inf_{P \in \overline{X}} d(M, P),$$

wth d being a metric, classically the euclidean distance. $\overline{X}$ denotes the complement of X, and finally $\boldsymbol{\rho}_{SK}$ is the restriction of ρ to $SK(X)$. The distance map for a digital object is usually computed by Euclidean Distance Transform (EDT) methods [7].

The sets of closest background points to a object point M, $\prod(M)$, is known to be the set of projections of M (on the background). It is defined by:

$$M \longmapsto \prod(M) = \{P \in \overline{X} \mid d(M, P) = \boldsymbol{\rho}(M)\}$$

As we said before, skeletal points are defined as centers of the object maximal disks, so if an object point M satisfies $|\prod(M)| \geq 2$, M is the center of a maximal ball, thus M is a skeletal point.

2.2 Shape characterization from the medial axis

Finally, let us look at skeletons from a structural point of view. As we stated in [9] the skeleton of an object is formed by pieces of curves (2D and 3D cases) and surfaces (3D case only) linked together by junctions. The pieces of curves and surfaces which do not contain any junctions are called pure curves and surfaces (therefore the connected components which remain when removing junction elements from a skeleton are pure curves and surfaces). In the following, we will refer to these pieces of skeleton by the expression *skeleton parts*. Finally by frontier points we denote the points which end skeleton parts and are not in contact with junction components (see Figure 1).

As the skeleton $SK(X)$ of an object X is a thin set, it allows us to classify the object topologically, thus allowing the study of the skeleton topology and therefore of the object topology. The topological classification of $SK(X)$, denoted $SK_c(X)$, attaches to each point of the skeleton one of the following labels:

- **Type F**: Frontier Point
- **Type C**: Pure Curve Point
- **Type J**: Junction Point
- **Type S**: Pure Surface Point (3D case only)

depending of which component does the current skeleton point belong to, respectively Frontier, Junction, Pure Curve and Pure Surface.

Let us now briefly recall the general idea of the n-D euclidean skeletonization twofold process presented in [12] that we have been considered in this work:

1. Local characterization of skeletal points by the introduction of two measures based on the Euclidean distance mapping.
2. Global topological reconstruction of the skeleton from two intermediate skeletons, one giving the adequate detail level, and the other giving the topology to be preserved.

As an example, Figure 1 illustrates the fact that branches of the skeleton of a 2D object are pure curves linked by junction components, and frontier components are the ends of the skeleton branches which are not in contact with a junction element.

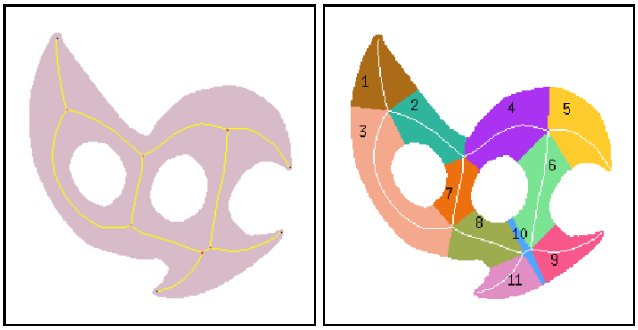

Fig. 1. On the left, topologically characterized skeleton of a 2D object. Skeleton branches (pure curves) are in yellow (or light gray), junction components in red (or black) and frontier components in black. On the right, object partition.

From the previous labeled skeleton parts we are able to obtain an object partition with region significance. Each of these regions will be related to a skeleton part. A well known skeleton based object partitioning method is skiz (skeleton by influence zones), which has been widely used. The skiz is obtained through the following steps: 1) compute the object skeleton and label the skeleton parts; 2) compute the distance map of the skeleton parts; 3) propagate the skeleton parts labels to the object points using the previous distance map, i.e., attach to each object point the label of its closest skeleton point.

The medial axis of an object is a very compact and informative representation of the object. To each skeletal point M we can attach the different attributes (the relative value of $\boldsymbol{\rho}(M)$, where $\boldsymbol{\rho}$ is the distance map of X; or its topological label: **Type F, J, C** or **S**).

From the **skeleton parts** labeling, we also infer a meaningful partition of the object into regions, each of these regions being associated to one of the

skeleton parts. So, now we have skeleton parts and associated object regions. To each skeleton part $P^i_{SK(X)}$ and object region R^i_X we can attach different attributes: i) $P^i_{SK(X)}$ size compared to skeleton size; ii) variation of the distance map along $P^i_{SK(X)}$; iii) R^i_X region size relative to the object size; iv) variation of the curvature sign along $P^i_{SK(X)}$

We have also characterized different skeleton points: **junctions** can be detected and labeled by a connected components extraction of type **J** pixels of $SK_c(X)$. To each skeleton junction $J^i_{SK(X)}$ we can attach different attributes: its order as defined by the number of *skeleton parts* which meet at junction $J^i_{SK(X)}$; the value of $\boldsymbol{\rho}$ at the junction $J^i_{SK(X)}$ compared to the maximal value of $\boldsymbol{\rho}$ in $SK(X)$.

Finally, we have identified **frontier points** in the skeleton. One possible attribute related to these points is the relative value of $\boldsymbol{\rho}$. The larger the value of $\boldsymbol{\rho}$, the smoother the curvature change at the boundary will be.

3 Image registration

3.1 Definition

Image registration can be defined as a mapping between two images (I_1 and I_2) both spatially and with respect to intensity:

$$I_2(x, y, z, t) = g(I_1(f(x, y, z, t))) \tag{1}$$

We can usually find situations where intensity difference is inherent to scene changes, and thus intensity transform estimation given by g is not necessary. In this contribution, we will consider f represents an isometric transform, i.e. rotation, translation and uniform scaling.

3.2 Image registration methods

Iterated Closest Point: The well known *(ICP)* method was proposed by Besl and McKay [2], and later on extended in different papers ([16], [8]):

- The point set P with N_p points $\boldsymbol{p_i}$ from the data shape and the model X —with N_x supporting geometric primitives: points, lines, or triangles— are given.
- The iteration is initialized by setting $P_0 = P$, the registration transform by $\boldsymbol{q_0} = [1, 0, 0, 0, 0, 0, 0]^t$, and $k = 0$. Next four steps are applied until convergence within a tolerance $\tau > 0$.
  1. Compute the matching between the data (scene) and model points by the closest assignament rule: $Y_k = C(P_k, X)$
  2. Compute the registration: $f_k(P_0, Y_k)$
  3. Apply the registration: $P_{k+1} = f_k(P_k)$
  4. Terminate iteration when the change in mean square error falls below τ

The use of the algorithm includes different advantages: i) it is independent of the shape representation (it allows the use of CAD data without any preprocessing, for instance); ii) no need of data smoothing when there are no outliers; iii) no need of data derivatives; iv) the method can be generalized to n-dimensional problems using the SVD algorithm; and v) it is possible to speed up the method.

However, the algorithm also presents some important drawbacks: i) it is very sensitive to outliers presence; ii) the initial states for global matching play a basic role for the success of the method when dealing with important deformations between model and scene points; iii) the estimation of the initial states mentioned above is not a trivial task, and iv) the cost of a local adjustment can be important if a low percentage of oclussion is present.

In view of the later, the algorithm performs bad when dealing with important transformations. As Zhang stated in [16]: "we assume the motion between the two frames is small or approximately known", hence this is a precondition of the algorithm to get reasonable results. Figure 2 shows several examples of that with one of the shapes considered in our experimental study.

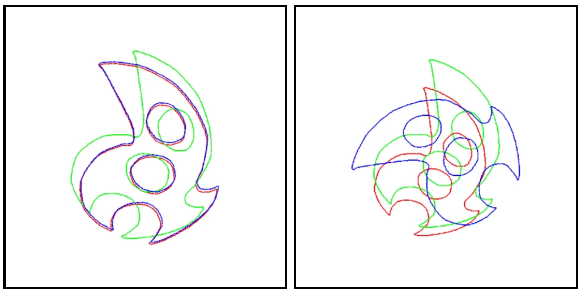

Fig. 2. ICP performance differs when dealing with small or big transforms. On the left: in blue, original shape; in green, original shape with a 20 rotation; in red, ICP estimation applied to the original object. On the right: same distribution of colours for a 90 rotation.

Robust point matching: Rangarajan et al. ([14]) present the *RPM* method based on finding the registration transform and the one-to-one correspondences between point features extracted from the images and rejecting non-homologies as outliers. The RPM algorithm minimizes the following objective function:

$$\min_{M,\theta,t,s} E(M,\theta,t,s) = \sum_{i=1}^{N_1}\sum_{j=1}^{N_2} M_{ij}\|X_i - t - sR(\theta)Y_j\|^2 + \frac{\gamma}{2}(\log s)^2 - \alpha\sum_{i=1}^{N_1}\sum_{j=1}^{N_2} M_{ij} \tag{2}$$

subject to

$$\sum_{i=1}^{N_1+1} M_{ij} = 1, \forall j \in \{1, ..., N_2\}, \sum_{j=1}^{N_2+1} M_{ij} = 1, \forall i \in \{1, ..., N_1\} \; and \; M_{ij} \in \{0, 1\}$$

Equation (2) describes an optimization problem from which the transformation parameters can be obtained by minimization. This problem contains two related optimization ones: the matching between the points of the two images and the registration transform estimation. The result is a two-stage algorithm which alternates between solving for the registration and the matching.

4 Iterated local search for the 2D registration problem

Instead of following previous approaches ([1]) based on searching for a good matching and then solving for the registration transform, in our work, we use the ILS metaheuristic for jointly solving our two-fold registration problem finding a good matching between both image points and getting the best isometric registration transform we supposed it has been applied. To do so, the basics of ILS are first described, and the local search (LS) algorithm considered and the different ILS components are later analyzed.

4.1 The iterated local search metaheuristic

ILS [13] belongs to the group of metaheuristics that extend classical LS methods by adding diversification capabilities to them. This way, ILS is based on wrapping a specific LS algorithm by generating multiple initial solutions to it as follows:

procedure *Iterated Local Search*
  s_0 = GenerateInitialSolution
  s^* = LocalSearch(s_0)
  repeat
    s' = Perturbation ($s^*, history$)
    $s^{*'}$ = LocalSearch(s')
    s^* = AcceptanceCriterion($s^*, s^{*'}, history$)
  until termination condition met
end

Hence, the algorithm starts by applying LS on an initial solution and iterates a procedure where a strong perturbation is applied on the current solution s^* (in order to move it away from its local neighborhood), and the solution so obtained is then considered as initial starting point for a new LS, from which another locally optimal solution $s^{*'}$ is obtained. Then, a decision is made between s^* and $s^{*'}$ to get the new current solution for the next iteration.

4.2 The local search algorithm considered

As said, this LS procedure allows us to obtain a complete solution to the 2D registration problem: a point matching between the data and the model shapes, and a registration transform to move the former into the latter. To do so, we only search in the matching space (only the point matching is encoded in the LS solution) and derive the registration by a least squares estimation as done in the ICP and RCP methods (see section 3.2).

The point matching between both images is represented as a permutation π of size $N = max(N_1, N_2)$, with N_1 and N_2 being the number of points in the data and model shapes, respectively. If $N_1 \geq N_2$, then $\pi(i)$ represents the model point associated to the data point i and viceversa. Notice that this representation has two main pros: i) it is based on a permutation, a very common structure in the field (used for example to solve the traveling salesman and the quadratic assignment problems), and ii) it allows us to deal with the case when both images have a different number of points, thus automatically discarding the outliers.

Moreover, the other novelty of our method is that the features of the shape are used to guide both the matching and the registration. This way, the objective function will include information regarding them both as follows:

$$\min_{M,\theta,t,s} E(M,\theta,t,s) = w_1 \cdot \sum_{i=1}^{N_1}\sum_{j=1}^{N_2} M_{ij}\|X_i - t - sR(\theta)Y_j\|^2 + w_2 \cdot f(M) \qquad (3)$$

where the first term stands for the registration error (M is the binary matrix storing the matching encoded in π and θ, t, s are the isometric transform parameters to be estimated (see expression 2)), the second one for the matching error, and w_1, w_2 are weighting coefficients defining the relative importance of each.

As regards the second term, there are different ways to define the f function evaluating the goodness of the matching stored in M as a big amount of information can be obtained from the medial axis ([9]). In this contribution, we have chosen the following:

$$f(M) = 0.75 \cdot pointtype \; + \; 0.25 \cdot (medaxis + length + izs)$$

where pointtype measures the error associated to the assignment of points of different types and the remaining three criteria refer to the variation of the distance map along different branches (see section 2.2).

Finally, the neighborhood operator is the usual 2-opt exchange, based on selecting two positions in π and exchanging their values. The LS considered is the first improvement variant, where the whole neighboorhod is generated to obtain the best neighbor and the algorithm iterates till the latter does not outperform the current solution.

4.3 The iterated local search components

GenerateInitialSolution: A random permutation is computed.

Perturbation: As a stronger change than the one performed by the 2-opt LS neighborhood operator is needed, we deal with the random exchange of the positions of the values included within a randomly selected sublist of size $\frac{N}{a}$, with $a \in \{2,3,4,5,6\}$ (of course, the lesser the value of a, the stronger the perturbation applied).

AcceptanceCriterion: We directly select the best of s^* and $s^{*'}$ as current solution, as usually done.

Termination condition: The algorithm stops when a fixed number of iterations is reached.

5 Experiments and analysis of results

We have applied the algorithm to two different shapes, over which different rotations, translations and scales have been developed to test the behavior of the technique. In the first row of figure 3 we show the original shapes and the corresponding partitions induced by medial axis transform, the information that will be used during the registration transform computation to guide the matching among points.

The ILS algorithm has been run during 50 iterations considering $a = 2$ for the perturbation operator. The weights in the objective functions have been defined as $w_1 = 1$ and $w_2 = 2 \cdot \frac{mse(R_{initial})}{f(M(\pi_{initial}))}$, with $R_{initial}$ and $f(M(\pi_{initial}))$ being respectively the registration and matching errors of the initial solution.

The results obtained are showed in rows 2 to 6 of figure 3. Second row corresponds to a simple 60^o rotation of the first original object; next column shows its partition; in the third column we store the representation of the mapping among different points of the different object (we have used several colours to distinguish the mappings of each kind of point). It is important to see in this column the way in which only points of same type match each other, even in the presence of noise in part of the object (third and fourth row) and different projection and junction points differ from the original and the transform shape. Last column reflects the effect of applying the estimated transform to the original shape and doing a superposition on the transformed object. Third row corresponds to a 180^o rotation with the presence of noise in part of the object. Next row adds a scale transform of 0.5 to the original object. In fifth and sixth rows, 30 and 90 with a 0.5 scale value transforms have been respectively applied to the top right shape.

6 Concluding remarks

In this contribution, we have formulated the image registration problem as a two fold one in order to jointly solve for the matching and the search for the isometric parameters of the registration transform. To do so, we take advantage

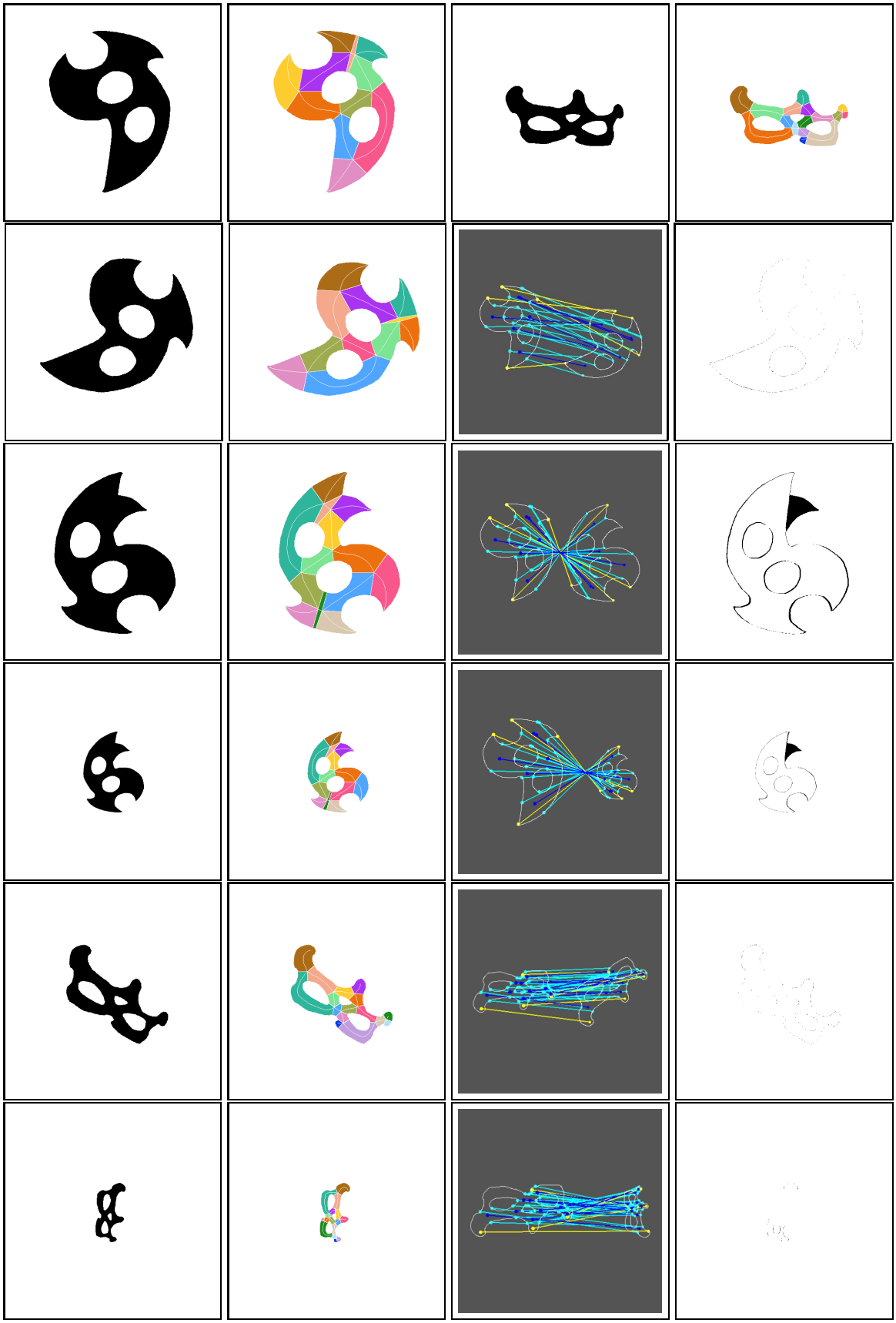

Fig. 3. First row: two original shapes and partition induced by the medial axis. Next rows (from left to right): isometric transform applied to the original shape, object partition, matching results: different point types with different colours (yellow→frontiers, blue→junctions, cyan→projections).Finally the superposition of the transformed image and the result of applying the estimated transform to the original shape.

of the information infered from the medial axis of the object and we use the ILS metaheuristic to face such a complex optimization task. We have presented results with important transformations applied to different model shapes.

References

1. E. Bardinet, S. Fernández-Vidal, S. Damas Arroyo, G. Malandain, and N. Pérez de la Blanca Capilla. Structural object matching. In *2nd International Symposium on Advanced Concepts for Intelligent Vision Systems (ACIVS 2000)*, page To appear, Baden-Baden, Germany, August 27-30 2000.
2. P.J. Besl and N.D. McKay. A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14:239–256, February 1992.
3. C. Blum and A. Roli. Metaheuristics in combinatorial optimization: overview and conceptual comparison. Technical Report 2001-13, IRIDIA, Univeristé Libre de Bruxelles, 2001.
4. H. Blum. A transformation for extracting new descriptors of shape. In W. Wathen-Dunn, editor, *Models for the Perception of Speech and Visual Form*, pages 362–380. M.I.T. Press, Cambridge, MA, 1967.
5. L.G. Brown. A survey of image registration techniques. *ACM Computing Surveys*, 24(4):325–376, December 1992.
6. L. Calabi. A Study of the Skeleton of Plane Figures. Technical Report 60429,sr-2, Parke Mathematical Laboratories, December 1965.
7. P.E. Danielsson. Euclidean distance mapping. *Computer Graphics and Image Processing*, 14:227–248, 1980.
8. J. Feldmar and N. Ayache. Locally affine registration of free-form surfaces. In *Computer Vision and Pattern Recognition (CVPR'94)*, pages 496–501, Seattle, USA, June 1994. IEEE.
9. S. Fernández-Vidal, E. Bardinet, S. Damas Arroyo, G. Malandain, and N. Pérez de la Blanca Capilla. Object representation and comparison inferred from its medial axis. In *15th International Conference on Pattern Recognition*, page To appear, Barcelona, Spain, September 3-8 2000.
10. K.P. Han, K.W. Song, E.Y. Chung, S.J. Cho, and Ha Y.H. Stereo matching using genetic algorithm with adaptive chromosomes. *Pattern Recognition*, 34:1729–1740, 2001.
11. S. Loncaric. A survey of shape analysis techniques. *Pattern Recognition*, 31(8):983–1001, 1998.
12. G. Malandain and Fernández-Vidal S. Euclidean Skeletons. *Image and Vision Computing*, 16(5):317–327, April 1998.
13. H. Ramalhinho, O. Martin, and T. Stützle. *Handbook of Metaheuristics*, chapter Iterated Local Search. F. Glover and G. Kochenberger (Eds.), 2002. To appear.
14. A. Rangarajan, H. Chui, E. Mjolsness, S. Pappu, L. Davachi, P. Goldman-Rakic, and J. Duncan. A robust point-matching algorithm for autoradiograph alignment. *Medical Image Analysis*, 1(4):379–398, 1997.
15. S.M. Yamany, M.N. Ahmed, and A.A. Farag. A new genetic-based technique for matching 3-d curves and surfaces. *Pattern Recognition*, 32:1817–1820, 1999.
16. Z. Zhang. Iterative point matching for registration of free-form curves and surfaces. *International Journal of Computer Vision*, 13(2):119–152, 1994.