Towards Automatic Skeleton Extraction with Skeleton Grafting

Cong Yang, Bipin Indurkhya, John See, and Marcin Grzegorzek

Abstract—This paper introduces a novel approach to generate visually promising skeletons automatically without any manual tuning. In practice, it is challenging to extract promising skeletons directly using existing approaches. This is because they either cannot fully preserve shape features, or require manual intervention, such as boundary smoothing and skeleton pruning, to justify the eye-level view assumption. We propose an approach here that generates backbone and dense skeletons by shape input, and then extends the backbone branches via skeleton grafting from the dense skeleton to ensure a well-integrated output. Based on our evaluation, the generated skeletons best depict the shapes at levels that are similar to human perception. To evaluate and fully express the properties of the extracted skeletons, we introduce two potential functions within the high-order matching protocol to improve the accuracy of skeleton-based matching. These two functions fuse the similarities between skeleton graphs and geometrical relations characterized by multiple skeleton endpoints. Experiments on three high-order matching protocols show that the proposed potential functions can effectively reduce the number of incorrect matches.

Index Terms—Skeleton Extraction, Skeleton Matching, Shape Matching, Skeleton Grafting, High-order Matching.

1 Introduction

Skeleton is an important descriptor for shape recognition and animation as it offers a low-dimensional and intuitive shape representation. Shape similarity based on skeleton matching usually performs better than contour or other shape descriptors in the presence of partial occlusion and articulation of parts [1], [2]. Moreover, skeletons have the potential to provide a compact, but meaningful, shape representation, suitable for both neural modelling and computational applications [3]. Following the convention in [1], [4], [5], an intuitive explanation of skeleton extraction is that if we keep collecting the centre points (or close to the centre points) of maximally inscribed circles (also called *disc*) that touch the shape contour in at least two places, these centre points are known as *skeleton points*. Some skeleton components are formally defined as follows:

- Endpoint: a skeleton point that has only one adjacent point.
- Junction point: a skeleton point that has three or more adjacent points.
- Connection point: a skeleton point that is neither an endpoint nor a junction point.
- Skeleton branch: a sequence of connected points within two directly connected skeleton points.

Though skeleton extraction has been extensively studied in the past decades [6], it is still hard to apply existing approaches in practice, particularly using massive shape representation for retrieval scenarios. As shown in Table 1, this is because most methods either cannot guarantee the topological and geometrical features in the representation, or require manual tuning of different parameters. For instance, a parameter k is required in the Discrete Curve Evolution (DCE) [5] method to calibrate the power of pruning. As shown in Figure 1, skeletons extracted under different k have different levels of completeness. To generate proper skeletons, k is normally selected manually [5], [28] for each shape by justifying the human eye-level view assumption. Recently, research has progressed towards training non-linear models such as U-Net [25] for skeleton extraction, especially with incorporating deep learning techniques. However, the generalization and the robustness of these models are not guaranteed, particularly on non-rigid objects [26], [27]. Therefore, fully automatic skeleton extraction, being as close as possible to real-world applications, is necessary for promoting research in this field.

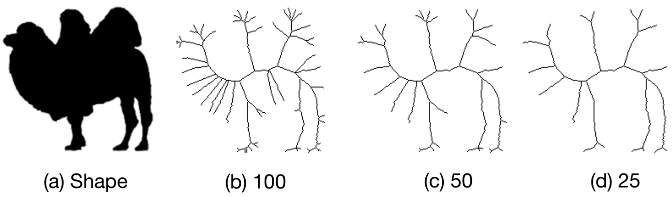

(a) Shape (b) 100 (c) 50 (d) 25

Fig. 1. Skeletons of camel shape under different k in DCE [5].

To advance skeleton extraction research, this paper introduces a novel approach that can be applied automatically without any manual tuning. As shown in Figure 2, our method is inspired by tree grafting [29], which is a horticultural technique whereby branches of a tree are joined so that they continue to grow together. The upper and the lower parts of the combined branch are called the scion and rootstock, respectively. Accordingly, we first estimate an initial skeleton, called a *backbone skeleton* (similar to root-

- *Cong Yang is with the Institute for Vision and Graphics, University of Siegen, Siegen, Germany; Bipin Indurkhya is with the Computer Science and Cognitive Science Departments, Jagiellonian University, Cracow, Poland; Jonh See is with the Faculty of Computing and Informatics, Multimedia University, Selangor, Malaysia; Marcin Grzegorzek is with the Institute of Medical Informatics, University of Lübeck, Lübeck, Germany. E-mail: cong.yang@uni-siegen.de/*

Manuscript received XX XX, 20XX; revised XX XX, 20XX.

TABLE 1
Comparison between the proposed method and other widely used skeleton extraction methods.

Method	Parameters Need Manual Tuning	Geometrical	Topological	Supervised
Voronoi-Skeleton [7]	Approximation degree ΔR	Yes	Yes	No
Connectivity Criterion [8]	Contour point distance ρ	Yes	Yes	No
Canonical Skeletons [9]	Reconstruction penalty ω	Yes	Yes	No
DCE [5]	Stop parameter k	Yes	Yes	No
Iterative Shrinking [10]	Characteristic value α	Yes	Yes	No
BPR [11]	Filter threshold t	Yes	Yes	No
DECS [12]	Complexity $th_{perc-max-ball}$ and $th_{ridge-high}$	Yes	Yes	No
Hierarchical1 [4]	Skeleton level T	Yes	Yes	No
Hierarchical2 [13]	Smoothing $n_{smooth-max}$	Yes	Yes	No
AOF [14]	Complexity th_{aof} and th_{rdg}	Yes	Yes	No
Propagated Skeleton (PS) [15]	Skeleton complexity ρ and α	Yes	Yes	No
One Step (OS) [16]	Boundary precision ε	Yes	Yes	No
Similarity Domains Network (SDN) [17]	Number of bins m	No	No	Yes
Distance Transform [18]	No	Yes	No	No
Thinning [19], Bayesian [20], HE [21], LK [22]	No	No	Yes	No
GMM [23], FHN [24]	No	No	Yes	Yes
U-Net [25], SkeletonNet [26], PSPU-SkelNet [27]	No	Yes	Yes	Yes
Proposed	**No**	**Yes**	**Yes**	**No**

stock), which best 'describes' the overall shape. Though the backbone skeleton preserves the topological shape features, and is robust to small deformations, the main skeleton branches are inherently shortened. As a result, there could be no skeleton branches correlated to some significant shape parts. Because of this, we apply a second step to extend the short branches by performing skeleton grafting from a dense skeleton (similar to the supplier of scions). The generated skeleton after this step would be the one that best 'fits' the shape. The rationale behind this is to generate a skeleton from coarse-to-fine, which is similar to human perception [30]. In Table 1, we note that in our proposed method, manual tuning is not needed during the skeleton extraction process, while both geometrical and topological shape features are preserved.

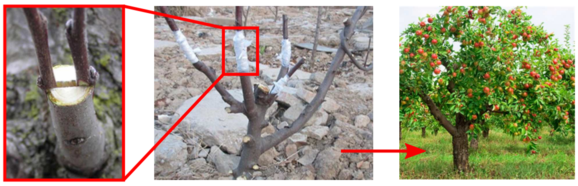


Fig. 2. The proposed method is inspired by tree grafting [29].

Theoretically, the proposed skeleton grafting method is inherently more stable than the skeleton growing approaches [11]. This is because the grafted skeleton branches are from dense skeletons, containing rich sub-region features, which are independent from the backbone skeletons. On the other hand, the skeleton growing methods rely heavily on the endpoints (also called root points in some literature) of the backbone skeleton. If the endpoints are far from the shape boundary, the grown skeleton branches become unstable as there is a high probability that the grown paths are sub-optimal. Finally, most growing methods require manual intervention for skeleton pruning.

When it comes to skeleton matching, it is important to incorporate similarities between skeleton endpoints into the overall similarity. However, most existing matching methods are based on one-to-one endpoint matching, which does not consider geometric relations characterised by multiple endpoints. As a result, matching is not robust as it could be easily disturbed by endpoints from spurious skeleton branches. For example, Figure 3 illustrates a matching between two elephant graphs using the classic Hungarian algorithm [1]. The wrong assignments are caused by a spurious branch at the tail of the left elephant (the green point).

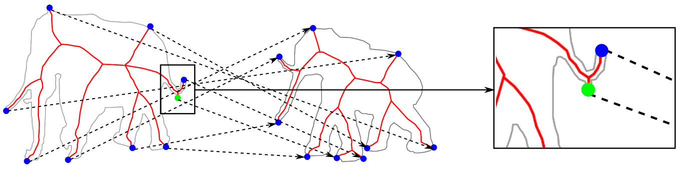


Fig. 3. Mismatch caused by spurious branches.

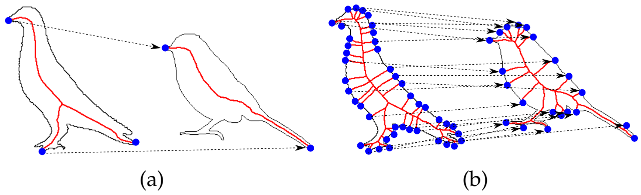


Fig. 4. Skeleton matching with different number of branches using [1].

It should be noted that we cannot improve the skeleton matching performance by simply increasing or reducing the number of skeleton branches [4]. The main reason is that a skeleton endpoint cannot properly find its correspondence if there are too many or too few candidate endpoints during the matching process (e.g. Figure 4). Moreover, if we increase the number of skeleton branches, the overfitted endpoints reduce the matching performance and increase the overall matching cost [30].

In order to evaluate and fully express the properties of extracted skeletons, we propose to optimize skeleton matching by considering additional geometric relations among multiple endpoints. A match between the skeleton graphs is then modelled as a high-order matching problem [31]. Due to theoretical advances and empirical successes, high-order matching has been given attention and many methods have been proposed ([32], [33], [34] and the references therein). However, it is unclear which potential functions are suitable for our skeleton graph matching tasks. Moreover, it is interesting to explore the performances of different hypergraph matching algorithms for the scenario of skeleton-based shape matching. In this paper, we propose the singleton and third-order potential functions for the 2D skeleton matching problem.

The contributions of this article include (1) the introduction of an automatic skeleton extraction approach. The generated skeletons are visually promising, and accurately 'describe' and 'fit' the original shape of the object. (2) Singleton and third-order potential functions are proposed under a high-order matching protocol to improve the skeleton matching performance. Experiments on eight datasets demonstrate the efficiency and robustness of our proposed skeleton extraction method. Moreover, a set of evaluations under different high-order matching protocols show the efficiency of our proposed potential functions.

2 Automatic Skeleton Extraction

C. Firestone *et al.* studied shape skeletons in human vision [28] from a psychological point of view. They observed that shape skeletons in human perception not only depend on whether shapes are represented completely, but also succinctly. More specifically, a promising shape skeleton should have at least three features: (1) The skeleton should be regionally or globally symmetric if the shape is regionally or globally symmetric. (2) There should be at least one skeleton branch in each intuitive shape region. (3) There should be at most one skeleton branch in each shape region with relatively minor boundary perturbations. Figure 5 shows the full pipeline of the proposed skeleton extraction approach, which extracts the backbone $\widehat{S}$ and dense skeleton S_{dense} (both described in Sections 2.1 and 2.2 respectively) from the input shape D for the purpose of skeleton grafting. The backbone best 'describes' the shape as it preserves the topological information about the shape. The dense skeleton preserves the full shape details. Then, in Section 2.3, we extend the backbone branches via skeleton grafting to ensure that the output skeleton S is more integrated and fitting to the shape. Overall, we generate a skeleton from coarse to fine, which is similar to human perception [30].

2.1 Backbone

As shown in Figure 5, a *backbone skeleton* $\widehat{S}$ refers to the major skeleton of a shape (also known as a topology-preserving skeleton [5]). It preserves the topology of the original shape, and the skeleton branches are normally shortened to reduce its sensitivity to small variations and noise at the shape boundaries. Based on preliminary experiments in [28], a promising backbone should be "intuitively generated" under the simple assumption that the major shape parts should be touched. Thus, we generate skeleton backbones using Bayesian estimation [20], as it has been shown to be accurate in modelling human contour perception [35] and free of user-specified parameters. With D denoting the original shape of $\widehat{S}$, Eq. 1 illustrates the basic idea of Bayesian estimation [20]:

$$p(\widehat{S}|D) = \frac{p(D|\widehat{S})p(\widehat{S})}{\sum_i p(D|\widehat{S}_i)p(\widehat{S}_i)} \tag{1}$$

where the dense skeleton $\widehat{S}$ is generated under a probability density function $p(\widehat{S})$. Its shape D, in turn, is generated from $\widehat{S}$ under a conditional probability density function playing the role of a likelihood function $p(D|\widehat{S})$. In order to model this, Bayesian estimation is applied to identify a shape's most likely "generative skeleton" under simple assumptions about the probability distribution of skeletons $\widehat{S}_i$ (providing a Bayesian prior), and a stochastic model of how shapes are generated from skeletons (providing a Bayesian likelihood function), where $\sum_i$ is the sum over all possible skeletons $\widehat{S}_i$. As the denominator in Eq. 1 is constant for a given shape, it can be maximized by maximizing the numerator, i.e., the product of the prior and the likelihood.

In practice, as introduced in [20], a prior probability density $p(\widehat{S})$ is computed by accumulating the prior density (calculated by angular measurements in [36]) of its axial segments, which are hierarchically organized into a root skeleton path, branches, sub-branches, etc. The general impression is that relatively straight axes will produce high probability, whereas the probability decreases with larger turning angles, i.e., larger magnitude of curvature in the underlying curve. For the likelihood model, the generated shape is the shape boundary formed by the rib endpoints. As shown in Figure 6(a), ribs sprout from both sides of each axis in directions that are perpendicular to the axis plus a random directional error. Building on that, the likelihood $p(D|\widehat{S})$ is calculated by accumulating the likelihood of shape boundary points. In particular, the likelihood of each point is the product of its correlated rib features such as length, directional error and length error. This prior and likelihood are then combined by Bayes' rule to identify the generative skeleton that is most likely to produce the shape. The estimated skeleton, also known as the *maximum a posteriori* (MAP) skeleton, is the skeletal interpretation that, under the generative assumptions underlying the prior and the likelihood functions, best "explains" the shape D. In other words, the branches of $\widehat{S}$ are correlated to the natural parts of D.

However, there are still two problems with the backbone skeleton $\widehat{S}$: (1) The branches are slightly shortened due to the skeleton simplification in Bayesian prior. (2) The skeleton is not guaranteed to be medial (marked in Figure 6(a)) due to the variance in rib lengths in the Bayesian likelihood. Although the first problem can be solved by the skeleton grafting method, the second problem badly affects the grafting process as the dense skeletons are often found at the shape centre. For this, we introduce a point shifting method to maximize the mediality (or centrality) of the backbone skeleton. Let E and C' denote an endpoint (blue) and a skeleton point (orange), respectively, on the skeleton branch in Figure 6(b). C is the skeleton point (red) after shifting from C'. Starting from E, such shifting is applied in three steps:

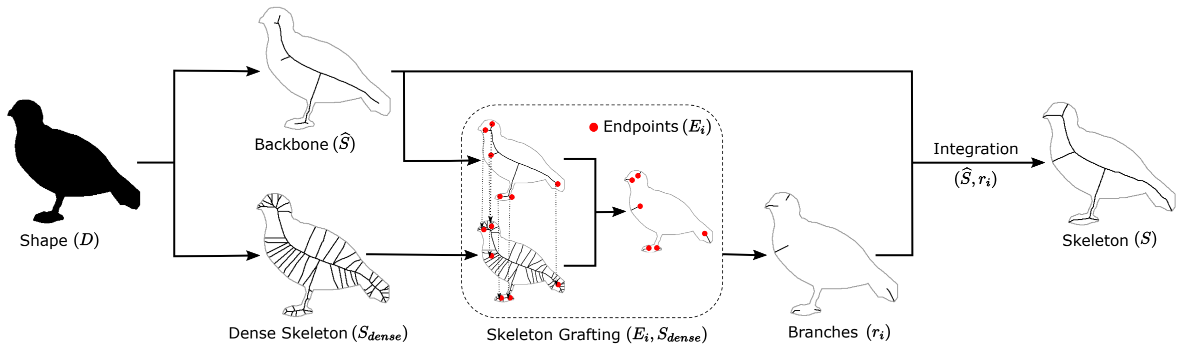


Fig. 5. Pipeline of the proposed skeleton extraction method.

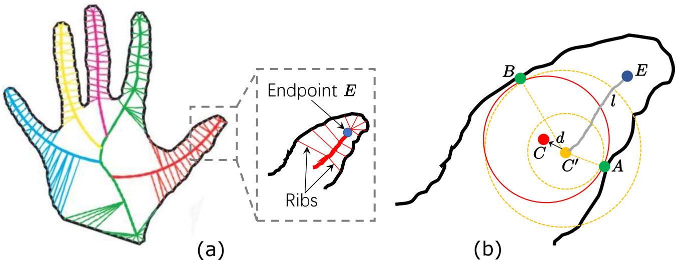


Fig. 6. Backbone skeleton generation. (a) Likelihood model [20] and (b) point shifting to maximize the mediality of backbones.

1) Using C' as the centre point, two maximally inscribed discs (orange circles with dashed line) are drawn touching the shape boundary at A and B, respectively.
2) A maximally inscribed disc (the red circle) is drawn touching the shape boundary at both A and B. Its centre point (red) is denoted by C.
3) Shift C' in the direction of $\overrightarrow{C'C}$ for a distance d:

$$d = \begin{cases} 0 & \text{if } d' \leqslant 2 \\ \dfrac{d'(L-l)}{L} & \text{otherwise} \end{cases} \qquad (2)$$

where d' denotes the distance in pixels between C' and C. Theoretically, $d' \leqslant 2$ can ensure the connectivity between the backbone and the grafted branches as skeletons are one pixel wide. As a result, dense skeletons can be grafted and docked with backbone endpoints by padding with a maximum of two pixels. L is the full branch length computed by counting the number of connected pixels in the branch, and l is the length from E to C' along the skeleton branch. In practice, a branch is represented by a list of pixel locations from junction point to endpoint. With this, we can easily compute L and l using the lengths of their respective lists.

As the boundary points are divided based on their correlation and relative positions within each branch, A and B are selected based on the shortest distance between C' and the shape boundary points within both sides of the branch. The shifting of C' is performed with gradually reducing power along skeleton paths to ensure smoothness of the branch. Finally, the shifting action within the branch is terminated once d becomes 0. The rationale behind Eq. 2 is that we do not shift all skeleton points, i.e. branches between junction points, thus ensuring that dense skeletons can be smoothly grafted and docked with the backbone endpoints.

Note that Eq. 2 can be simply updated to the full skeleton shifting once we set $d = d'$. However, our preliminary experiments show that backbone skeletons, before and after full shifting, have only a mean accuracy difference of around 0.02% for skeleton matching. This is also theoretically confirmed in [20]: the mediality of the original backbone skeletons tend to be maximized in conjunction with other properties such as skeletal simplicity and low variance in the rib lengths. Therefore, we propose Eq. 2 to reduce the complexity of the shifting process.

2.2 Dense Skeleton

As shown in Figure 5, a *dense skeleton* S_{dense} constitutes a dictionary of all big and small branches since it preserves the original shape features with a considerable number of redundant branches. Our target is to ensure each backbone endpoint can properly find its corresponding branch from S_{dense} for grafting. Therefore, the number of skeleton branches in S_{dense} should be at least $2 \sim 3$ times more than the one in $\widehat{S}$. Theoretically, most unsupervised skeletonisation approaches in Table 1 can be employed for generating S_{dense}. In this paper, we extend the classic skeletonisation method in [8], [37] for fast dense skeleton generation. The general pipeline is presented in Figure 7. Inputting a binary shape, first the Euclidean Distance Transform method is employed for shape preprocessing. After that, the maximum value points are recursively added and examined following the ridges in $DT(D)$ until the final S'_{dense} is extracted.

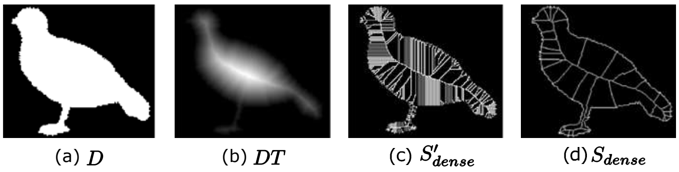


Fig. 7. Dense skeleton generation pipeline.

To reduce the computational complexity of S_{dense} extraction, it is useful to control the density of skeleton branches in S'_{dense}. As the skeletonisation process generates skeletons from sparse to dense, it is easy to set up a fixed but general threshold k' to control the progress. However, S_{dense} should not be too sparse for grafting as some grafting errors may arise if the endpoints from $\widehat{S}$ cannot properly find their corresponding branches from S_{dense}. Figure 8 illustrates the extraction speed (in seconds) of S_{dense} as well as the correlated grafting errors by varying the value of k' in our preliminary experiment [30]. We can clearly see that $k' = 100$ is sufficient to generate a dense skeleton S_{dense} by balancing the grafting error and the complexity. This configuration is validated through a set of experiments in Section 4 as there is no grafting error in eight databases.

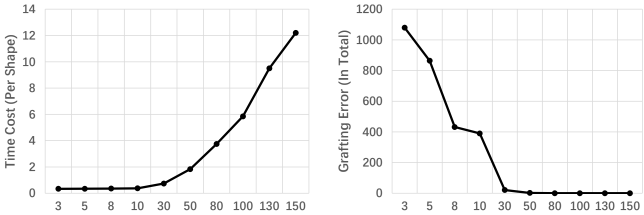
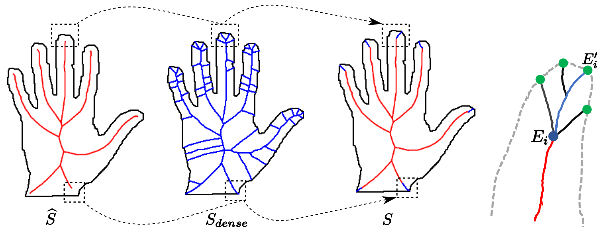


Fig. 8. S_{dense} extraction speed (left) and the correlated grafting errors (right) by varying the value of k' (x axis) in our preliminary experiment [30]. $k' = 100$ is general enough for the experiments in Section 4.

It is important to mention that it may seem that DCE can also be used to generate both $\widehat{S}$ and S_{dense}, that is by choosing a sufficiently high value of k to ensure that we obtain enough branches for different shapes, and choosing a low value of k to produce only a bare skeleton. However, we cannot employ DCE to generate the backbone as $k \geq 3$ by definition and DCE with $k = 3$ cannot properly preserve the topology of complex shapes. This is because there are maximally three branches extracted to form a dense skeleton if $k = 3$, and therefore some complex shape regions may be missing correlated branches.

2.3 Grafting



(a) Grafting (b) Branch

Fig. 9. The proposed grafting method. The blue line $r(E_i, E_i')$ in (b) is one of the grafted branches for $\widehat{S}$ and S_{dense} integration in (a).

To fully use the properties of backbone $\widehat{S}$ and dense skeleton S_{dense}, we apply a skeleton grafting process to "stretch" each endpoint E to its correlated boundary corner E'. The general idea is illustrated in Figure 9. If E in $\widehat{S}$ is not on the shape boundary, we connect it with its correlated boundary corner E' using a grafted skeleton path $r(E, E')$ from S_{dense}. This idea is based on the fact that each E in $\widehat{S}$ is strictly shifted to the shape center and is therefore close to or even overlapped with S_{dense}. We observe that $r(E, E')$ are mostly located within two pixels from E. Therefore, we search $r(E, E')$ from neighbouring four pixels of E. If there is still no $r(E, E')$ returned, a grafting error is signaled.

Let us assume that there are M' endpoints in $\widehat{S}$, and E_i is one of them, $i = 1, 2, ..., M'$. Starting from E_i, we can find its correlated boundary corners $E'_{i,j}$ along the direction of skeleton growing in S_{dense}, $j = 1, 2, ..., N'$. In other words, the direction from the junction point to E_i that is sharing the same branch. This idea can effectively reduce the computational complexity when searching for $E'_{i,j}$ from E_i as the shape boundary points in the opposite direction can be discarded. In practise, a skeleton is represented by a graph with branches from junction points to endpoints. Therefore, a branch is encoded along the direction of skeleton growing by default. Let r_i denote the grafted path $r(E_i, E_i')$, which is generated by:

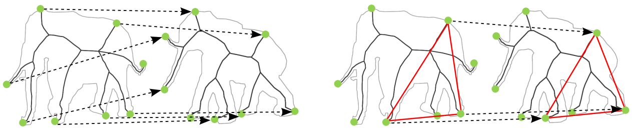
$$r_i = \max(pd(r_{i,j})) \tag{3}$$

where $pd(r_{i,j})$ is the path length of $r(E_i, E'_{i,j})$. Similar to L in Eq. 2, $r(E_i, E'_{i,j})$ is computed by counting the pixel number as skeletons are one pixel wide. Building on $\widehat{S}$ and r_i, the final integrated skeleton S is generated by

$$S = \widehat{S} \cap r_i \quad . \tag{4}$$

3 High-order Skeleton Matching

The main purpose of skeleton matching is to find optimal correspondences between skeleton graphs so that the similarity values between pairs of skeletons can be calculated. Based on our previous work in [31], we formulate skeleton matching within the high-order matching protocol with different order of potentials. Given two skeletons S_1 and S_2, let E_1 and E_2 denote all endpoints in S_1 and S_2 respectively. Also, let $e_{1,i}$ and $e_{2,j}$ be single endpoints in E_1 and E_2, respectively, for $i = 1, 2, \cdots, M$, $j = 1, 2, \cdots, N$, $M \geq N$. As shown in Figure 10(a), singleton potential is essentially an assignment problem, where each endpoint of a skeleton is matched with an endpoint in another skeleton. For a high-order matching (Figure 10(b)), we consider the cost of matching three correspondences. More specifically, a triple of endpoints in a skeleton is matched with a triple in another one. High-order matching protocol is employed to improve the establishment of correspondences between skeleton endpoints, thereby improving the accuracy of skeleton matching. The rationale behind this is that even if two endpoints are mismatched in the singleton potential using the Hungarian algorithm, they could be re-adjusted by a higher-order potential.



(a) Singleton potential (b) Third-order potential

Fig. 10. The proposed singleton and third-order potentials.

3.1 The Singleton Potential

Let $\theta_{e_{1,i},e_{2,j}}$ be defined as the cost of singleton potential for the correspondence between two endpoints $e_{1,i}$ and $e_{2,j}$. We employ the Path Similarity-based Skeleton Matching (PSS) [1] method for singleton potential calculation. Taking properties of geodesic paths [38] between $e_{1,i}$ and $e_{2,j}$, PSS method can find the optimal endpoint correspondences for calculating the global skeleton dissimilarity.

However, there are three major limitations of this method. Firstly, this method cannot properly handle flipped images. This is because each dissimilarity cost $c(e_{1,i}, e_{2,j})$ between endpoints is calculated by the Optimal Subsequence Bijection function (OSB-function), which is not flexible enough to search for backward correspondences [38]. Therefore, this method is not able to estimate a reliable matching on flipped images. Secondly, it is not always possible to assign a correct match for each skeleton endpoint. The main reason is that each endpoint has to be assigned to one matching endpoint based on the Hungarian algorithm, even if the two points do not correspond correctly. Thirdly, since each endpoint has to be assigned to a unique matching endpoint, spurious skeleton branches have a significant negative impact on the matching result.

To overcome the first limitation, we apply the OSB-function twice: once on the original image, and once on the horizontally flipped version of the original image. From the resulting lists of two matches, the one with lower dissimilarity cost $c(e_{1,i}, e_{2,j})$ will be chosen as the real match. For the second and the third limitations, the proposed third-order potential can properly alleviate them, thereby improving the matching accuracy. Rather than the traditional one-to-one endpoint matching, the third-order potential performs matching by extracting the correspondences of multiple nodes. In such a case, even if $e_{1,i}$ and $e_{2,j}$ are mismatched in the singleton potential using Hungarian algorithm, they could be reassigned by the third-order potential. We use $c_1, c_2, \cdots, c_N$ to denote the cost of the matched endpoints, with the global dissimilarity $c(S_1, S_2)$ calculated as their summation. Finally, the singleton potential of a correspondence $(e_{1,i}, e_{2,j})$ is determined as:

$$\theta_{e_{1,i},e_{2,j}} = 1 - c_j \quad . \tag{5}$$

where $j = 1, 2, \cdots, N$.

3.2 The Third-Order Potential

The third-order potential is defined using the top two largest internal angles formed by a triplet of skeleton endpoints. Our method is different from the potential function in [31], which forms the third-order potential using three full angles ordered in a clockwise fashion. Theoretically, our method is more efficient as we do not need to compare all the angles within a triangle. This is based on the fact that if two corresponding angles of two triangles are congruent, the triangles must be similar. Moreover, the potential function in [31] is not invariant to rotation, as the order of endpoints within a triangle could be changed if the shape is rotated. In contrast, our proposed method is rotation-invariant as we rank endpoints based on the size of their correlated internal angles. This assumption is validated by the experiment in Section 4.3.2. As the number of possible matching triplets between two skeletons is very large, it is cost-effective to reduce the complexity of each match using our method.

Specifically, let $(e_{1,i}, e_{1,u}, e_{1,z}) \in E_1$ and $(e_{2,i}, e_{2,u}, e_{2,z}) \in E_2$ be two triplets, then the cost of third-order potential $\theta_{e_{1,i},e_{1,u},e_{2,i},e_{2,u}}$ for each possible matching triplet $(e_{1,i}, e_{1,u}, e_{1,z}) \rightarrow (e_{2,i}, e_{2,u}, e_{2,z})$ is defined with a truncated Gaussian kernel:

$$\theta_{e_{1,i},e_{1,u},e_{2,i},e_{2,u}} = \begin{cases} \exp(-\gamma \| f_{i_1,u_1} - f_{i_2,u_2} \|^2) & \text{if } \| f_{i_1,u_1} - f_{i_2,u_2} \| \leqslant \vartheta \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where f_{i_1,u_1} (or f_{i_2,u_2}) is a two-dimensional vector which describes sine values of the top two biggest internal angles of triangles formed by endpoints $(e_{1,i}, e_{1,u})$ (or $(e_{2,i}, e_{2,u})$). The truncated Gaussian kernel is used to scatter and reduce the matching times by filtering out those redundant pairs that are not likely to be matched. Also, γ is a scaling factor, and we empirically set it to 2 in our experiment. With Eq. 6, for each triplet in E_1, we find the matching triplets in E_2 within the neighbourhood of size ϑ.

4 Experiment

In this section, we first evaluate the usability of our skeleton generation approach. Then the efficiency of our proposed high-order skeleton matching algorithm is validated. We also conduct skeleton-based shape retrieval experiments to evaluate the overall performance using our proposed extraction and matching methods. For comparison, experiments are run on eight datasets: Perception105 [30], Kimia216 [39], Kimia99 [39], Tari56 [40], Tetrapod [4], MPEG7 [41], Animal2000 [41] and Pixel SkelNetOn [42]. As most employed datasets are actively used and can be easily checked in the cited literatures, here we only briefly introduce Pixel SkelNetOn dataset because we need to apply a post-processing step to convert the shape point clouds into binary shapes in our scenario. As shown in Figure 11, it consists of 1,725 shapes in the format of shape point clouds. As this dataset is designed for training and testing deep learning models, its images are pre-formatted to size 256×256, and partitioned into 1,218 training, 241 validation and 255 test images [42]. For a fair comparison, we employ the same test images in our experiments.

Fig. 11. Sample shapes from SkelNetOn dataset.

4.1 Skeleton Extraction

We evaluated our proposed method in four aspects: (1) closeness to human perception, (2) preservation of the major features of the original shapes, (3) validation for matching, and (4) comparison with deep learning-based approaches. The first three are presented in this section, and the fourth one is discussed in Section 4.2

4.1.1 Closeness to Human Perception

To evaluate whether the extracted skeletons are close to human perception, we conducted experiments based on Perception105 dataset and questionnaires with 35 participants (students). Specifically, for each shape in the questionnaire, three candidate skeletons were generated using the popular DCE method ($k = 12$) [5], the recently published PS method

($\rho = 0.7, \alpha = 2.1$) [15] and our proposed method from Section 2 above. For the DCE and PS methods, we employed the parameters from their original papers. We also visualized both skeleton and shape boundary in the questionnaire to clarify the original shape structure. For each shape, only a number was attached for subsequent analysis, while its class name was hidden from the participants to reduce the possible influence of context information. During the voting process, the participants were allowed to choose only one skeleton for each question. Finally, the skeleton with the highest votes was selected as the final skeleton chosen based on human perception. If two skeletons received the same number of votes, we repeated the voting process (for that skeleton set) until a clear winner was obtained.

TABLE 2
Comparison between DCE [5], PS [15] and our method on the Perception105 dataset. Sum and standard deviation (STD) of endpoints are presented on the left and right half of the last row, respectively.

	Most Voted Skeletons			Mean Endpoint Number		
	DCE	PS	Proposed	DCE	PS	Proposed
glass	3	0	12	8	8	6
camel	7	0	8	10	12	9
elephant	6	0	9	9	10	7
bird	5	0	10	7	8	5
heart	5	0	10	8	7	5
dolphin	4	1	10	8	7	5
folk	4	0	11	7	7	5
Sum:	34	1	70	STD: 1.0690	1.9024	1.5274

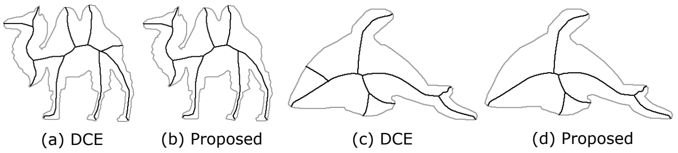


Fig. 12. Failure cases of the proposed grafting method.

Table 2 illustrates the results of a human perception test conducted on the Perception105 dataset to compare between the three methods. On the left half of the table, we compare the number of times the skeletons generated by the methods were most voted on, distinguished by class. As skeleton branches with PS method are often inconsistent (some connected to shape boundaries and some are not), it has the poorest perception among the three. For DCE and the proposed method, it can be observed that their performances are closer to each other on complex shapes such as *camel*. This is understandable as the pruning power in the DCE method is set to $k = 12$, which is close to the optimized parameter reported in [5]. Hence, some failure cases are observable in the proposed method. For instance, almost half of the voted *camel* skeletons are generated by the DCE method, rather than the proposed one. Such failure cases demonstrate the limitations of the proposed method whereby the grafting procedure is highly dependent on the quality of dense skeletons. Figure 12 presents two examples: volunteers preferred Figures 12 (a) and (c) extracted by the DCE method because the branches of the *camel* tail and the *dolphin* head are missing in the grafted skeletons (also dense skeletons) of Figure 12 (b) and (d), respectively.

However, the DCE fared poorly in certain classes such as *glass* and *folk*. Considering all classes, our method is closer to human perception. To see why is this so, the standard deviation of endpoints in each class are shown on the right half of the table. We can see that the skeleton endpoints that are produced by our method are more flexible (STD: 1.5274) and adapted in different classes. Endpoints with the DCE method, however, are more rigid (STD: 1.0690) as they are determined by a fixed pruning power.

4.1.2 Representing Original Shapes

We employed two metrics that have been used in the past for comparing the representability of extracted skeletons on MPEG7 dataset. Specifically, (1) Hausdorff distance $\mathscr{H}(D, D')$ [13] between the original shape D and the shape D' reconstructed by the skeleton S, and (2) Ratio $\mathscr{A}(D, D')$ between the change in the shape area $|\Delta(D) - \Delta(D')|$ and the original shape area $\Delta(D)$. For comparison, DECS [12], PS [15] and DCE [5] methods were employed using the proposed parameters in their original papers. We also manually modified the pruning power of the DCE method (DCE-Manual) on each individual shape to explore the influence of manual tuning; this method is the most likely to find very good skeletons but is extremely tedious. The comparison results are shown in Table 3.

TABLE 3
Comparison of skeleton representability between DECS [12], PS [15], DCE [5] and our proposed method on MPEG7 dataset. The Hausdorff distance $\mathscr{H}(D, D')$ in pixels, the area difference $\mathscr{A}(D, D')$ in percentages. DCE-M is short for DCE-Manual method.

	DECS	PS	DCE	DCE-M	**Proposed**
Mean $\mathscr{H}(D, D')$	8.1	2.0	1.4	1.0	**1.0**
Median $\mathscr{H}(D, D')$	4.6	2.1	1.6	1.2	**1.1**
Mean $\mathscr{A}(D, D')$	2.6	4.1	2.3	1.3	**1.3**
Median $\mathscr{A}(D, D')$	2.1	3.2	1.8	**1.1**	1.2

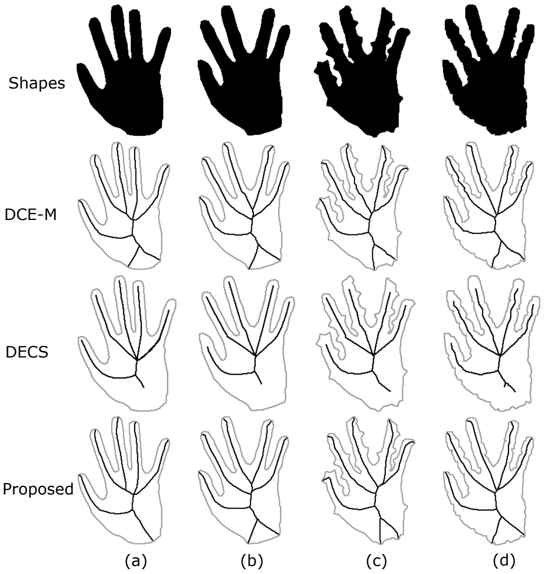


Fig. 13. Extracted skeletons with DCE-M (ground truth), DECS [12] and the proposed method.

As skeleton branches produced by the DECS and the PS are simpler and shorter than the DCE and our proposed method, $\mathscr{H}(D, D')$ and $\mathscr{A}(D, D')$ in the last three columns are smaller than those in the first two. In other words, shapes reconstructed by the DCE and our proposed method are more similar to the original shape.

In Table 3, we see that both $\mathscr{H}(D, D')$ and $\mathscr{A}(D, D')$ using our proposed method perform better than the DCE. This is because the fixed pruning power of DCE is not generic enough to cover all classes in the MPEG7 dataset; in particular, some complex shapes are not catered for. This is verified by the result for DCE-Manual, which achieved similar performance to our proposed approach since the skeleton branches generated by both the methods are fully adapted to the respective shape's complexity. However, DCE-Manual took 5.5 hours and 10 volunteers to tune the parameters. In contrast, our proposed method took less than 2 hours to generate all the skeletons automatically on a laptop computer with Intel Core i7 2.2 GHz CPU, 8.00 GB memory and 64-bit Windows 8.1 OS. This demonstrates the obvious advantages and efficiency of our proposed method.

To better demonstrate the robustness of the proposed method, we employed two hand shapes (Figure 13 (a) and (b)) from Kimia99 and extract their skeletons using DECS and the proposed method. In columns (c) and (d), we also manually added articulation and noise to the original shape in (b), and proceeded to extract their skeletons. We observe that the skeletons extracted by the proposed method appear more robust and closer to the ground truth (DCE-Manual), even when articulation and noise are added.

4.1.3 Ability for Matching

To evaluate the usability of skeletons for shape matching, we compared our method against three commonly used approaches and one recently published skeleton extraction method using the same matching method in a skeleton-based shape retrieval scenario. Specifically, DCE [5], DCE-Manual, BPR [11], OS [16] and our proposed method were evaluated based on the PSS [1] matching algorithm. The PSS method was selected because their authors have made the code publicly available and it is also actively used in many other research works. The evaluation is built on a retrieval framework introduced in [4]. The experiments were carried out on four datasets (Kimia216, Kimia99, Tari56 and Tetrapod) and the results are shown in Table 4 and 5. Results in these tables can be interpreted as follows: The numbers indicate the number of shapes in the i-th position that belong to the same class as their query shapes. Thus, a higher number at lower positions indicates better matching.

It can be observed that the BPR method performs similar to the DCE in four datasets, while the DCE-Manual and our proposed method achieve the best result. This is expected as each skeleton generated by the DCE-Manual and the proposed method is optimized to adapt to the complexity of the original shape, the former manually and the latter automatically. We also find that OS has the lowest performance across all four datasets. This is because their skeleton branches are mostly shortened, so parts of the topological and the geometrical features cannot be preserved for skeleton matching. Such observations show the advantage of our proposed skeleton grafting process. To further verify these observations, we applied an additional operation to extend the skeleton branches generated by OS [16] using the proposed skeleton grafting method in Section 2.3. Results in Table 4 and 5 (row OS-G) show that the overall performance of OS can be further improved with the inclusion of our grafting procedure.

TABLE 4
Experimental comparison of various skeleton extraction methods on Tari56 dataset. BPR_Paper refer to the results reported in the original paper [11] while BPR_Implementation are results based on our re-implementation. OS-G refer to the updated skeletons after performing additional grafting step on OS [16].

Tari56	1st	2nd	3rd	4th
DCE	56	49	44	40
DCE-Manual	56	51	48	43
BPR_Paper Report	55	**55**	**55**	**53**
BPR_Implementation	56	49	45	40
OS	53	49	41	37
OS-G	56	50	46	42
Proposed	**56**	50	47	43

TABLE 5
Experimental comparison of our skeleton extraction method with four methods on Kimia216, Kimia99 and Tetrapod datasets. DCE-M is short for DCE-Manual method. OS-G refers to the updated skeletons after performing additional grafting step on OS [16].

Kimia216	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th
DCE	216	210	209	204	196	197	176	173	164	152
DCE-M	216	**213**	211	**211**	208	200	191	**183**	170	**161**
BPR	216	211	209	203	193	193	178	172	166	153
OS	213	196	187	178	170	161	153	139	120	120
OS-G	215	212	209	200	193	194	181	173	165	160
Proposed	**216**	212	**212**	209	**210**	**203**	**191**	182	**173**	160
Kimia99	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th
DCE	99	97	97	97	**96**	92	93	81	71	68
DCE-M	99	99	99	99	**96**	**97**	95	93	89	**73**
BPR	99	97	97	96	**96**	93	93	82	71	68
OS	99	94	89	89	87	80	76	71	65	61
OS-G	99	99	98	97	95	92	90	87	84	70
Proposed	**99**	**99**	**99**	**99**	95	96	**95**	**94**	**90**	72
Tetrapod	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th
DCE	120	109	101	98	81	78	68	66	65	59
DCE-M	120	**115**	104	**101**	93	87	**79**	**75**	**70**	62
BPR	120	110	101	98	82	79	67	67	66	60
OS	117	112	93	90	80	73	63	60	56	50
OS-G	120	113	103	94	91	83	71	68	63	60
Proposed	**120**	114	**107**	100	**93**	**88**	76	74	69	**62**

4.2 Comparison with Supervised Approaches

We explored the deep learning methods in shape analysis and observed that those methods are mostly used in three tasks: (1) modelling binary shape images for shape classification [43]; (2) performing data augmentation for training another machine learning method [44]; and (3) extracting shape skeletons [26]. Motivated by this, we compared our proposed grafting approach with the recent deep learning-based skeleton extraction methods such as Pix2pix [42], U-Net [25], FHN [24], GMM [23] and SkeletonNet [26] (in

Table 6) based on SkelNetOn dataset. These approaches were mainly introduced in the 2019 SkelNetOn Challenge in Geometric Shape Understanding Workshop [42]. For a fair comparison, we employed the same configuration as for the benchmark and evaluation methods. In particular, we employed the testing data (255 images) and the F1-score (the harmonic average of the precision and recall values from the skeleton and the background pixels) for evaluation. The only difference is that our approach did not require any training process.

Overall, the results confirm again the strength and generalization of our proposed skeleton extraction approach, which marginally surpassed the performance of SkeletonNet. However, the improvement in performance is not significant on this dataset. We observe that the ground truth skeletons do not fully preserve topological shape features. An intuitive explanation is that some ground truth skeleton branches are shortened and their endpoints are not connected to the shape boundary (see the ground truth of U-Net in Figure 14). As a result, some skeleton points extracted from the proposed method are misclassified as negative pixels, even when parts of their correlated branches do match with the ground truth.

TABLE 6
Experimental comparison of our proposed method against supervised methods: Pix2pix [42], U-Net [25], FHN [24], GMM [23] and SkeletonNet [26] on SkelNetOn dataset.

Methods	Pix2pix	U-Net	FHN	GMM	SkeletonNet	**Proposed**
F1 Score	0.6244	0.7846	0.6477	0.6456	0.7877	**0.7891**

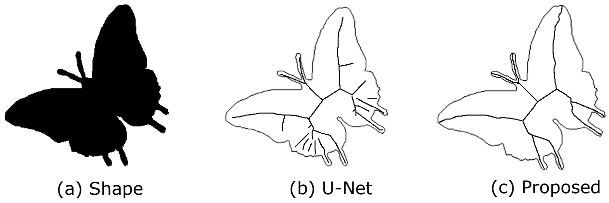

(a) Shape (b) U-Net (c) Proposed

Fig. 14. Extracted skeletons of the *butterfly* shape using the U-Net [25] and proposed method.

4.3 High-order Matching

To further evaluate the quality of matching, we evaluated the skeleton matching task on two datasets, with two specific purposes: (1) Kimia99 dataset was used for visually evaluating the extent of the improvement achieved. To eliminate the influence of skeleton quality, we employed the skeletons generated by Manual-DCE and varied the matching algorithm for comparison. As the shapes are mostly noisy and occluded, the Kimia99 dataset poses a greater challenge for matching algorithms. (2) Kimia216 dataset was used for quantitatively evaluating the performance improvement in retrieval scenarios.

4.3.1 *Visual Evaluation*

To visually count the incorrect correspondences, five volunteers participated in a time-consuming exercise to obtain statistics based on the printed matching samples, as illustrated in Figure 15. For example, two tools with deformations on both sides are matched using only the first-order Hungarian algorithm and its combination with the third-order potentials. As shown in Figure 15 (left), there are two mismatched pairs because of somewhat similar features in both shapes. Moreover, geometrical relations among the endpoints are not considered. Figure 15 (right) shows that our proposed high-order matching method yields an appropriate matching.

Table 7 depicts a comparison of the number of incorrect correspondences among different matching algorithms. We first counted the total number of true correspondences (*i.e.* lines in Figure 15) between skeletons of the same class (the second column). The remaining columns indicate the number of incorrect correspondences by the competing methods: the first-order Hungarian algorithm (the third column) and other high-order matching protocols (HGM [45], RRWHM [46] and Dual-Decomposition [31]) using our proposed potential functions.

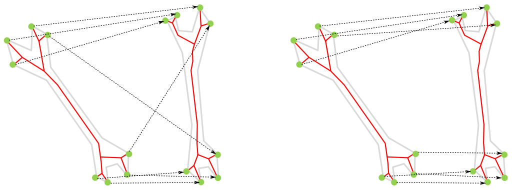


Fig. 15. Sample correspondences using different matching algorithms: first-order potential (left), first- and third-order potentials (right).

TABLE 7
Comparison of incorrect correspondences between Hungarian and high-order matching protocols using our proposed potential functions. The second column illustrates the total number of true correspondences in each class. DD is short for Dual-Decomposition.

Class	Total	Hungarian	HGM	RRWHM	DD
animal	1070	270	261	166	176
bunny	852	115	84	4	14
dude	1034	157	178	83	94
fish	414	49	62	27	28
hand	1319	497	286	180	180
hat	712	193	140	78	82
key	947	1	0	0	0
plane	1120	450	292	213	203
tool	718	130	85	18	26
All	8186	1862	1388	**769**	803

It can be observed that compared to the traditional Hungarian algorithm that is the baseline, all three high-order protocols significantly reduce the number of incorrect matches: HGM by ∼5%, RRWHM by ∼13% and Dual-Decomposition by ∼12%. Note that the percentages in reduction here are determined with respect to the total number of correspondences (second column). This observation echoes the result of the high-order-based interesting point matching reported in [31]. It is to be expected as, regardless of whether interesting points or skeleton endpoints are utilised for matching, the RRWHM protocol can reflect the one-to-one matching constraints more effectively than the HGM protocol, which relies on random walks [46]. In practise, Dual-Decomposition is recommended because its computational complexity is more reasonable [33] with an accuracy that is slightly lower than that of the RRWHM protocol. Overall, Table 7 indicates that our proposed potential functions yield a significant improvement in terms of shape matching among the three high-order protocols.

4.3.2 Quantitative Evaluation

Our quantitative evaluation is built on a retrieval framework similar to Section 4.1.3. Using the generated skeletons from the DCE-Manual and our proposed methods, we applied the matching algorithms in Section 3 and the retrieval results are listed in Table 8. We can see that the retrieval performance on both skeleton extraction methods have improved with the use of our proposed potential functions. Moreover, the DCE-Manual and our proposed method achieved almost similar performances with high-order matching. Most importantly, our method can be applied automatically without any manual tuning.

TABLE 8
Retrieval performance after applying our proposed high-order matching algorithm (With-HO). No-HO refers to shape retrieval without applying high-order matching. DCE-M is short for DCE-Manual method.

	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th
DCE-M (No-HO)	216	213	211	211	208	200	191	183	170	161
DCE-M (With-HO)	**216**	**215**	**214**	**212**	**211**	**207**	**194**	**189**	**178**	**170**
Proposed (No-HO)	216	212	212	209	210	203	191	182	173	160
Proposed (With-HO)	**216**	**215**	**215**	**211**	**210**	**207**	**196**	**187**	**177**	**170**

In Table 9, we compared the shape retrieval performance using our proposed third-order potential in Section 3.2 against the one introduced in [31]. As the main purpose is to validate the usefulness of the rotation-invariant feature of our proposed method, we applied two comparisons based on the skeletons from original Kimia216 and the randomly rotated Kimia216 (R-Kimia216). In other words, we employed the skeletons from [31] and then randomly rotated them. To be fair, other elements such as the parameters, matching protocol, and evaluation methods remain the same. With Kimia216 dataset, we find that our proposed third-order potential achieved marginally better performance, as indicated by the better numbers closer to the top (1st). With the R-Kimia216 dataset, the performance improvement is more obvious. We also find that the retrieval performance on Kimia216 and R-Kimia216 are close to each other using our proposed third-order potential. This is because our proposed method is rotation-invariant and robust to rotated shapes.

TABLE 9
Retrieval performance of our proposed third-order potential versus the method introduced in [31].

Kimia216	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th
[31]	215	**216**	213	212	209	208	**193**	181	**175**	**172**
Proposed	**216**	215	**214**	**212**	**209**	**209**	192	**183**	174	169
R-Kimia216	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th
[31]	215	213	211	211	207	205	190	178	**174**	168
Proposed	**216**	**215**	**214**	**212**	**210**	**208**	**191**	**184**	173	**169**

4.4 Robustness of Recognition

Here, we evaluated the robustness of our proposed skeleton extraction and matching methods. For fairness, we compared its performance against other state-of-the-art methods [3] with the following considerations: (1) The ranking position score (RPS) [4] and the bulls-eye score (BES) [47] are employed on Animal2000 and MPEG7 datasets, respectively. RPS is an integration of correctly matched shape numbers and their ranking positions. BES is a ratio of the total number of correctly matched shapes to the total number of possible matches. (2) We selected only those works from the literature where both shape representation and matching algorithms were proposed so that the influence of other uncontrolled factors (e.g. ensembling of different representation and matching algorithms) can be reduced.

4.4.1 Experiment on Animal2000

For a comparison against our proposed method, we selected the following approaches from the Animal2000 dataset: Inner Distance (ID) [48], Shape Context (SC) [49], Path Similarity (PS) [49] and Hierarchical [4], as their performances on Animal2000 are reported in their respective publications. As shown in Table 10, our proposed method performs better than the popularly used Shape Context and Path Similarity approaches. This is because such representation algorithms are not able to obtain optimal representations using fixed parameters. The Hierarchical approach [4], which integrates the matching properties of around 12 hierarchical skeleton pairs, performs slightly better than our proposed method. However, considering the complexity of its integrated skeleton generation and matching procedure, our proposed method is more efficient than the hierarchical method, as we only extract one skeleton for matching. As more than 40% of shapes in Animal2000 datasets are with noise, the RPS of Inner-Distance is around 39% higher than our proposed method. In other words, Inner-Distance is more robust against shape noise than the skeleton-based methods, as shape noise generally affects the process of skeleton generation. One possible way to solve this is to smoothen the shape boundaries before skeletonisation. Besides, we can also consider employing a suitable feature representation for these skeletons to operate under supervised learning scenarios.

TABLE 10
Comparison of ranking position score (RPS) on Animal2000.

Method	**ID**	SC	PS	Hierarchical	Proposed
RPS	**452.66**	193.79	241.33	348.73	326.31

4.4.2 Experiment on MPEG7

We also report results on the MPEG7 dataset as shown in Table 11. Similar to the benchmarks in [4], these results are clustered into two groups: pairwise and context-based matchings. The first group is similar to the experiments above, while results in the second group are obtained by the underlying structure of the shape manifold [50]. In other words, similarity scores in the second group are post-processed to increase the discriminability between different shape classes. We can see that our proposed skeleton extraction with high-order matching achieves 81.55% BES, which is better than those of traditional skeleton-based methods [1], [5], [51]. We also find that the hierarchical approach [4] performs slightly better than our proposed method (the antepenultimate row in Table 10), as it integrates 12 skeletons for hierarchical matching. Unlike [4], our proposed method is faster because we only extract a single skeleton for matching.

For an easier comparison with the other approaches, we used the same configuration of [4] in the context-based group by employing the Mutual Graph (MG) method [52] based on the similarity between all shapes. Our method achieved 99.02% BES, outperforming most state-of-the-art methods and comes close to the hierarchical methods. But considering the aspect of applicability (no manual tuning) and speed (low computational complexity described in Section 4.5), our proposed method is more suitable for practical applications than the hierarchical approach. It should be noted that a generic framework for diffusion processes is introduced in [53], which achieved a 100% accuracy on MPEG7 dataset. In contrast to the diffusion approach, our performance is very close to 100% but we opt for the simple but fast MG method. However, we argue that 100% BES is impossible in practice since the human perception of some shapes in the same class are totally different.

TABLE 11
Comparison of bulls-eye score (BES) on MPEG7 dataset.

Pairwise Matching	BES	Context-based	BES
Shape Contexts [54]	76.51%	INSC + CDM [55]	88.30%
Skeletal Context [51]	79.92%	IDSC + LP [56]	91.00%
Optimized CSS [57]	81.12%	SC + LP [56]	92.91%
Multiscale Rep. [49]	84.93%	IDSC + LCDP [58]	93.32%
Shape Rouge [59]	85.25%	SC + GM + Meta [60]	92.51%
Inner Distance [48]	85.40%	IDSC + PS + LDCP [61]	95.60%
Hier.Procrustes [62]	86.35%	HF + LCDP [63]	96.45%
Shape Tree [64]	87.70%	SC + IDSC + Co-T [50]	97.72%
CPDH [65]	76.56%	IDSC + MG [52]	93.40%
HF [63]	**89.66%**	AIR [66]	93.67%
IP [67]	80.28%	IP+HG [67]	96.43%
Path Similarity [5]	75.16%	ASC + TN + TPG [68]	96.47%
Hierarchical1 [4]	81.62%	Hierarchical + MG	99.21%
Hierarchical2 [13]	78.21%	**Diffusion [53]**	**100%**
TCD [69]	75.50%	ASC + LDCP [70]	95.96%
Proposed	81.55%	Proposed + MG	99.02%

4.5 Computational Complexity

For generating backbone and dense skeletons, the time complexity is $O(log(N'))$, where N' is the number of boundary vertices. For skeleton grafting, it can be finished in $O(1)$. This is because the number of endpoints is small. By fusing these tasks, the overall complexity of the skeleton generation is $O(log(N'))$. For skeleton matching, as introduced in [71], the Hungarian algorithm can be solved in $O(N^3)$ time. For the third-order potential, assuming there are m_1 and m_2 endpoints in S_1 and S_2, respectively. Theoretically, it can be finished in $O(m_1^3 m_2^3)$ time, as there are $m_1^3 m_2^3$ possible triplet pairs. In practise, searching all possible pairs is not needed as introduced in [33]. Instead, we just fix a constant number of triplet pairs for all experiments. This allows our third-order potential to be finished close to $O(1)$ time.

5 Conclusion

We presented a novel skeleton generation approach for skeleton-based shape matching. Inspired by tree grafting, our proposed method extracts a skeleton in a coarse-to-fine fashion to extend the backbone skeleton for a better shape fitting. Experiments show that the extracted skeletons are perceptually meaningful, and can garner a similar level of performance as manually generated skeletons in shape retrieval scenarios. In order to fully express the skeleton properties for shape matching, we introduced two potential functions in the high-order matching protocol. A comprehensive set of experiments on various datasets and high-order matching protocols were conducted to demonstrate the effectiveness of our proposed potential functions.

Acknowledgments

This work was supported by the German Research Foundation under Grant GRK 1564.

References

[1] X. Bai and L. Latecki, "Path similarity skeleton graph matching," *IEEE PAMI*, vol. 30, no. 7, pp. 1282–1292, 2008.

[2] C. Yang, O. Tiebe, K. Shirahama, E. Łukasik, and M. Grzegorzek, "Evaluating contour segment descriptors," *Machine Vision & Applications*, vol. 28, no. 3-4, pp. 373–391, 2017.

[3] L. Kurnianggoro *et al.*, "A survey of 2d shape representation: Methods, evaluations, and future research directions," *Neurocomputing*, vol. 300, pp. 1–16, 2018.

[4] C. Yang *et al.*, "Object matching with hierarchical skeletons," *Pattern Recognition*, vol. 55, pp. 183–197, 2016.

[5] X. Bai *et al.*, "Skeleton pruning by contour partitioning with discrete curve evolution," *IEEE PAMI*, vol. 29, no. 3, pp. 449–462, 2007.

[6] C. Yang, *Object Shape Generation, Representation and Matching*. Berlin, Germany: Logos Verlag, 12 2016.

[7] R. Ogniewicz and M. Ilg, "Voronoi skeletons: theory and applications," in *IEEE CVPR*, 1992, pp. 63–69.

[8] W. Choi *et al.*, "Extraction of the euclidean skeleton based on a connectivity criterion," *Pattern Recognition*, vol. 36, no. 3, pp. 721–729, 2003.

[9] M. Eede *et al.*, "Canonical skeletons for shape matching," in *International Conference on Pattern Recognition*, 2006, pp. 64–69.

[10] S. Krinidis and V. Chatzis, "A skeleton family generator via physics-based deformable models," *IEEE TIP*, vol. 18, no. 1, pp. 1–11, 2009.

[11] W. Shen *et al.*, "Skeleton growing and pruning with bending potential ratio," *Pattern Recognition*, vol. 44, no. 2, pp. 196–209, 2011.

[12] A. Leborgne *et al.*, "Noise-resistant digital euclidean connected skeleton for graph-based shape matching," *VCIP*, vol. 31, pp. 165–176, 2015.

[13] A. Leborgne, J. Mille, and L. Tougne, "Hierarchical skeleton for shape matching," in *IEEE ICIP*, 2016, pp. 3603–3607.

[14] J. Mille *et al.*, "Euclidean distance-based skeletons: A few notes on average outward flux and ridgeness," *Journal of Mathematical Imaging and Vision*, vol. 61, no. 3, pp. 310–330, 2019.

[15] B. Durix *et al.*, "The propagated skeleton: A robust detail-preserving approach," in *Discrete Geometry for Computer Imagery*, 2019, pp. 343–354.

[16] B. Durix *et al.*, "One step compact skeletonization," *Eurographics*, pp. 21–24, 2019.

[17] S. Ozer, "Parametric shape modeling and skeleton extraction with radial basis functions using similarity domains network," in *IEEE CVPR Workshops*, 2019, pp. 1–5.

[18] L. J. Latecki *et al.*, "Skeletonization using ssm of the distance transform," in *IEEE ICIP*, vol. 5, 2007, pp. 349–352.

[19] C. Y. Suen *et al.*, *Thinning methodologies for pattern recognition*. World Scientific, 1994, vol. 8.

[20] J. Feldman and M. Singh, "Bayesian estimation of the shape skeleton," *Proceedings of the National Academy of Sciences*, vol. 103, no. 47, pp. 18 014–18 019, 2006.

[21] F. Gao *et al.*, "2d skeleton extraction based on heat equation," *Computers & Graphics*, vol. 74, pp. 99–108, 2018.

[22] L. Yang *et al.*, "A novel algorithm for skeleton extraction from images using topological graph analysis," in *IEEE CVPR Workshops*, 2019, pp. 1–5.

[23] C. Liu *et al.*, "Parametric skeleton generation via gaussian mixture models," in *IEEE CVPR Workshops*, 2019, pp. 1–5.

[24] N. Jiang *et al.*, "Feature hourglass network for skeleton detection," in *IEEE CVPR Workshops*, 2019, pp. 1–5.

[25] O. Panichev *et al.*, "U-net based convolutional neural network for skeleton extraction," in *IEEE CVPR Workshops*, 2019, pp. 1–4.

[26] S. Nathan *et al.*, "Skeletonnet: Shape pixel to skeleton pixel," in *IEEE CVPR Workshops*, 2019, pp. 1–5.

[27] R. Atienza *et al.*, "Pyramid u-network for skeleton extraction from shape points," in *IEEE CVPR Workshops*, 2019, pp. 1–4.

[28] C. Firestone and B. J. Scholl, "Please tap the shape, anywhere you like: Shape skeletons in human vision revealed by an exceedingly simple measure," *Psychological Science*, vol. 25, no. 2, pp. 377–386, 2014.

[29] T. O. Perry, "Dormancy of trees in winter," *Science*, vol. 171, no. 3966, pp. 29–36, 1971.

[30] C. Yang *et al.*, "Investigations on skeleton completeness for skeleton-based shape matching," in *IEEE SPA*, 2016, pp. 113–118.

[31] C. Yang *et al.*, "Shape-based object matching using interesting points and high-order graphs," *Pattern Recognition Letters*, vol. 83, pp. 251–260, 2016.

[32] J. Lee *et al.*, "Hyper-graph matching via reweighted random walks," in *IEEE CVPR*, 2011, pp. 1633–1640.

[33] O. Duchenne *et al.*, "A tensor-based algorithm for high-order graph matching," *IEEE PAMI*, vol. 33, no. 12, pp. 2383–2395, 2011.

[34] B. Kimia *et al.*, "Differential geometry in edge detection: Accurate estimation of position, orientation and curvature," *IEEE PAMI*, vol. 41, no. 7, pp. 1573–1586, 2019.

[35] J. Feldman and M. Singh, "Information along contours and object boundaries." *Psychological Review*, vol. 112, no. 1, pp. 243–252, 2005.

[36] K. Mardia *et al.*, "Statistics of directional data," *Journal of the Royal Statistical Society*, vol. 37, no. 3, pp. 349–371, 1975.

[37] M. Wright *et al.*, "Skeletonization using an extended euclidean distance transform," *Image and Vision Computing*, vol. 13, no. 5, pp. 367–375, 1995.

[38] L. Latecki *et al.*, "Optimal subsequence bijection," in *IEEE ICDM*, Oct 2007, pp. 565–570.

[39] T. B. Sebastian *et al.*, "Recognition of shapes by editing their shock graphs," *IEEE PAMI*, vol. 26, no. 5, pp. 550–571, 2004.

[40] C. Asian and S. Tari, "An axis-based representation for recognition," in *ICCV*, 2005, pp. 1339–1346.

[41] X. Bai *et al.*, "Integrating contour and skeleton for shape classification," in *ICCV Workshops*, 2009, pp. 360–367.

[42] D. Ilke *et al.*, "Skelneton 2019: Dataset and challenge on deep learning for geometric shape understanding," in *IEEE CVPR Workshops*, 2019, pp. 1–9.

[43] Z. Zhu *et al.*, "Binary shape classification using convolutional neural networks," *The IIOAB Journal*, vol. 7, no. 5, p. 332336, 2016.

[44] S. Eslami *et al.*, "The shape boltzmann machine: a strong model of object shape," *IJCV*, vol. 107, no. 2, p. 155176, 2014.

[45] R. Zass and A. Shashua, "Probabilistic graph and hypergraph matching," in *IEEE CVPR*, 2008, pp. 1–8.

[46] J. Lee *et al.*, "Hyper-graph matching via reweighted random walks," in *IEEE CVPR*, 2011, pp. 1633–1640.

[47] L. J. Latecki *et al.*, "Shape descriptors for non-rigid shapes with a single closed contour," in *IEEE CVPR*, 2000, pp. 424–429.

[48] H. Ling *et al.*, "Shape classification using the inner-distance," *IEEE PAMI*, vol. 29, no. 2, pp. 286–299, 2007.

[49] T. Adamek *et al.*, "A multiscale representation method for nonrigid shapes with a single closed contour," *IEEE TCSVT*, vol. 14, no. 5, pp. 742–753, 2004.

[50] X. Bai *et al.*, "Co-transduction for shape retrieval," *IEEE TIP*, vol. 21, no. 5, pp. 2747–2757, 2012.

[51] J. Xie *et al.*, "Shape matching and modeling using skeletal context," *Pattern Recognition*, vol. 41, no. 5, pp. 1756–1767, 2008.

[52] P. Kontschieder *et al.*, "Beyond pairwise shape similarity analysis," in *ACCV*, 2010, pp. 655–666.

[53] M. Donoser and H. Bischof, "Diffusion processes for retrieval revisited," in *IEEE CVPR*, 2013, pp. 1320–1327.

[54] S. Belongie *et al.*, "Shape matching and object recognition using shape contexts," *IEEE PAMI*, vol. 24, no. 4, pp. 509–522, 2002.

[55] H. Jegou *et al.*, "Accurate image search using the contextual dissimilarity measure," *IEEE PAMI*, vol. 32, no. 1, pp. 2–11, 2010.

[56] X. Bai *et al.*, "Learning context-sensitive shape similarity by graph transduction," *IEEE PAMI*, vol. 32, no. 5, pp. 861–874, 2010.

[57] F. Mokhtarian *et al.*, "Curvature scale space representation: Theory, applications and mpeg-7 standardization," in *Computational Imaging and Vision*. Kluwer Academic Pub, 2003.

[58] X. Yang *et al.*, "Locally constrained diffusion process on locally densified distance spaces with applications to shape retrieval," in *IEEE CVPR*, 2009, pp. 357–364.

[59] A. Peter *et al.*, "Shape iane rough: Sliding wavelets for indexing and retrieval," in *IEEE CVPR*, 2008, pp. 1–8.

[60] A. Egozi *et al.*, "Improving shape retrieval by spectral matching and meta similarity," *IEEE TIP*, vol. 19, no. 5, pp. 1319–1327, 2010.

[61] A. Temlyakov *et al.*, "Two perceptually motivated strategies for shape classification," *IEEE CVPR*, pp. 2289–2296, 2010.

[62] G. McNeill *et al.*, "Hierarchical procrustes matching for shape retrieval," in *IEEE CVPR*, 2006, pp. 885–894.

[63] J. Wang *et al.*, "Shape matching and classification using height functions," *Pattern Recognition Letters*, vol. 33, no. 2, pp. 134–143, 2012.

[64] P. F. Felzenszwalb *et al.*, "Hierarchical matching of deformable shapes," in *IEEE CVPR*, 2007, pp. 1–8.

[65] X. Shu *et al.*, "A novel contour descriptor for 2d shape matching and its application to image retrieval," *Image and Vision Computing*, vol. 29, no. 4, pp. 286–294, 2011.

[66] R. Gopalan *et al.*, "Articulation-invariant representation of non-planar shapes," in *ECCV*, 2010, pp. 286–299.

[67] C. Yang *et al.*, "Shape-based object matching using point context," in *ICMR*, 2015, pp. 519–522.

[68] X. Yang, *et al.*, "Affinity learning with diffusion on tensor product graph," *IEEE PAMI*, vol. 35, no. 1, pp. 28–38, 2013.

[69] C. Yang *et al.*, "A novel method for 2d nonrigid partial shape matching," *Neurocomputing*, vol. 275, pp. 1160–1176, 2018.

[70] H. Ling *et al.*, "Balancing deformability and discriminability for shape matching," in *ECCV*, 2010, pp. 411–424.

[71] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, pp. 83–97, 1955.

Cong Yang is a senior researcher at Horizon Robotics since 2019. Before that, he was a Post-doc researcher at the MAGRIT team in INRIA (France). Later, he worked scientifically at the Computer Vision and Machine Learning team in Clobotics. His main research interests are pattern recognition and its interdisciplinary applications. Cong earned his Ph.D. degree in computer vision and pattern recognition from the University of Siegen (Germany) in 2016.

Bipin Indurkhya is a professor of Cognitive Science at the Jagiellonian University, Cracow, Poland. His main research interests are social robotics, usability engineering, affective computing and creativity. He received his Masters degree in Electronics Engineering from the Philips International Institute, Eindhoven (The Netherlands) in 1981, and PhD in Computer Science from the University of Massachusetts at Amherst in 1985.

John See is a professor of Multimedia University. He received his B.Eng., M.EngSc. and Ph.D degrees from Multimedia University. He leads the Pattern Recognition and Analysis sub-group under the Center for Visual Computing. His research interests are in the field of computer vision and machine learning, particularly with the goal of finding effective and efficient algorithms for visual recognition tasks.

Marcin Grzegorzek is a professor of medical informatics with his research focus on medical data science at the University of Luebeck. He obtained his doctor of engineering degree in pattern recognition from the University of Erlangen-Nuremberg in 2007. Later, he worked scientifically at the Queen Mary University of London and the University of Koblenz-Landau. From 2010 to 2018, he was assistant professor of pattern recognition at the University of Siegen.