Journal of Information Hiding and Multimedia Signal Processing ©2016 ISSN 2073-4212
Ubiquitous International Volume **7**, Number **6**, November **2016**

Similarity Shape Based on Skeleton Graph Matching

Truong-Giang Ngo, Trong-The Nguyen
Department of Information Technology, Hai-Phong Private University, Vietnam
{giangnt,vnthe}@hpu.edu.vn

Quoc-Tao Ngo, Duc-Dung Nguyen
Institute of Information Technology,Vietnamese Academy of Science and Technology
{nqtao,nddung}@ioit.ac.vn

Shu-Chuan Chu[1,2]
[1]School of Computer Science, Engineering and Mathematics, Flinders University, Australia
[2]Fujian University of International Studies and Trade, Fujian Province, China
scchu0803@gmail.com

Received May 2016; revised July 2016

Abstract. *Using graphs to match two feature sets through embedded high-order relations points has many possible applications in criminal justice, security, and high technology. In this paper, we analyze the method of using the random walk framework to establish correspondence between two skeleton graphs and find out matching points between two shapes. The graphs are matched using a skeleton graph with the descriptors of the relationship between the two edges of the end-nodes ranked on an association graph. Through adopting individual jumps with a reweighting scheme, the new proposed approach effectively reflects the one-to-one matching constraints during the random walk process. Experiments on several benchmark data sets show that the proposed approach clearly outperforms existing algorithms, especially in the presence of noise and outliers.*

Keywords: Skeleton graph, Image matching, Shape retrieval, Skeleton graph matching, Random walks.

1. **Introduction.** Shape matching is a fundamental aspect of many applications in computer vision, and pattern recognition, including object or scene recognition, solving for 3D structure from multiple images, stereo correspondence, and moving tracking [1]. Established correspondence between two feature sets and finding each corresponding feature in the other set while preserving the other related features are widely applied to solve the various vision problems [1, 2]. Recent years have witnessed a popular way in which skeleton involved in the image matching problems[3, 4]. Skeleton can be defined as the set of centers of all maximally inscribed disks that are contained inside the object but not contained in any other such disk [3]. Integrating geometrical and topological feature of the object provides an efficient shape descriptor for object recognition. In some applications included the content-based image retrieval systems, character recognition systems, circuit board inspection systems, and analysis of biomedical images, the skeletal descriptor was applied as an efficient shape descriptor to obtain the better results than those obtained from the other methods which based on boundary or shape descriptors in the presence of partial occlusion and articulation of parts [5, 6, 7].

However, the automatic object skeletal recognizability has also the drawback due to the sensitivity of the skeletons with the object boundary deformations. Fig.1 illustrates the skeleton-based shape matching that faces the difficulties. Siddiqi *et al.,* proposed a

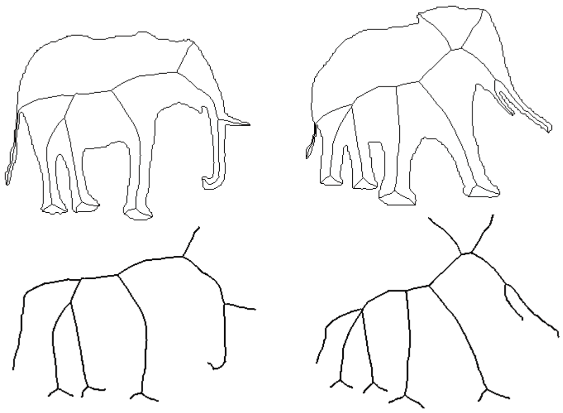

Figure 1. Visually similar shapes have very different skeleton graphs

method called Shock graph as a myriad method based on the skeleton [8]. Since the skeleton contains important structural information of shape, it is natural to organize them into attribute-relational graphs (ARG).The similarity between two objects can be measured by the matching their ARG graphs. The shock graph contains both topology and geometry information of the shape. The geometry information of the shape included the end-points, branch points, and their skeleton segments existed in the shock. Bone graph is an extension of the shock graph, which only retains the non-ligature structures of the shock graph and offers improved stability [9, 10, 11]. Although this method leads to fairly good matches, the errors of fundamental flows can violate the hierarchical relations among parts of the shape. Because the ARG matching task is proven to be NP-hard, several algorithms are proposed to obtain approximate solutions. Zaboli *et al.,* defined the edited distance between the shock graphs [12]. However, its computational complexity is high due to the complex edit operations. A relatively small change in the shape causes the root change into an entirely different topology for the similarity skeleton. The skeleton tree conversion processing may cause the important structural information losing, therefore, the shape matching results could be obtained some negative effects. Han *et al.,*[13] apply an EM algorithm to learn both the structure of the super-graph and the correspondences between the nodes of the sample graphs. They mainly focused on the topological structure and the general graph matching problem. Bai *et al.,*[14] represented each end point by the skeletal shortest paths emanating from it and addressed skeleton graph matching by matching the sequence of end points of the pruned skeletons. It didn't explicitly consider the topological structure of the skeleton graphs. Instead, they focused on the similarity of paths connecting the skeleton end points. However, these methods depend heavily on the selection of pruning parameters selected by heuristic. The different objects can have appropriate pruning thresholds are different. Thus, these methods are only suitable for objects with simple and distinctive shapes.

In this paper, we propose a novel scheme based on the skeleton graph for effective shape matching. In the proposed method, the establishing correspondences between two graphs are considered to model a skeleton hyper-graph (SHG) to overcome the limitations mentioned above. The complex relations among correspondences between two skeleton graphs are called hyper-edges that used in the SHG. A centroid of a node and k-nearest neighbors are used to form a hyper-edge. In this way, both the higher order grouping information and the local relationship between nodes within each hyper-edge are described as the model that is known as a hyper-graph matching. The shape matching problems is now considered as a hyper-graph matching problem, which can be solved feasibly by ranking on an association hyper-graph via random walks [15]. Furthermore, the parameter of the re-weighted random walks is adjusted as a scheme of ranking process by iteratively updating and exploiting the confidences of candidate correspondences. By this way, the unreliable matches are eliminated, and the robust performance is produced, therefore the accuracy of the proposed method of matching is significantly improved even in noise and distortion fields.

2. Generalized Hyper-graph Matching.

2.1. Problem Formulation.

A skeleton graph can be represented by hyper-graph $\mathcal{G}$. Let $\mathcal{V}$ be a finite set of vertices and $\mathcal{E}$ be a family of subsets of $\mathcal{V}$ such that $\cup_{e\in\mathcal{E}} = \mathcal{V}$. The hyper-graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$ consists of nodes $v \in \mathcal{V}$, hyper-edges $e \in \mathcal{E}$, and attributes $a \in \mathcal{A}$ associated with the hyper-edges. The goal of the hyper-graph matching problem is to establish a mapping between nodes of two hyper-graphs $\mathcal{G}^P = (\mathcal{V}^P, \mathcal{E}^P, \mathcal{A}^P)$ and $\mathcal{G}^Q = (\mathcal{V}^Q, \mathcal{E}^Q, \mathcal{A}^Q)$. Suppose a set of all possible node correspondences $\mathcal{C} = \mathcal{V}^P \times \mathcal{V}^Q$, and δ-tuples $c_{w1} = (v^P_{i1}, v^Q_{i1}), ..., c_{w\delta} = (v^P_{i\delta}, v^Q_{i\delta}) \in \mathcal{C}$ among them. For hyper-graph matching, the similarities $H_{w1,...,w\delta}$ of the δ-tuples are measured by comparing attributes of two δ-th order hyper-edges $e^P_{i1,...,i\delta}$ and $e^Q_{j1,...,j\delta}$, which mean the hyper-edges connecting $v^P_{i1}, ..., v^P_{i\delta}$ and $v^Q_{j1}, ..., v^Q_{j\delta}$ respectively. The solution of hyper-graph matching is determined as a subset of candidate correspondences $\mathcal{C}$ and efficiently represented using a binary assignment matrix $X \in \{0,1\}^{n^P \times n^Q}$, where n^P and n^Q are the numbers of nodes in $\mathcal{G}^P$ and $\mathcal{G}^Q$, respectively. $X_{ia} = 1$ implies that node v^P_i corresponds to node v^Q_a, e.g., node i in graph P is matched to node a in the graph Q, and $X_{ia} = 0$ otherwise. The hyper-graph matching problem is mathematically formulated as the following IQP problem that is, finding the assignment vector $\mathbf{x}^*$, which maximizes the matching score function $Score(\mathbf{x})$ as follows.

$$\mathbf{x}^* = \arg\max_{\mathbf{x}} Score(\mathbf{x}) \tag{1}$$

$$st.x \in \{0,1\}^{n^P n^Q}, X\mathbf{1}_{n^Q\times 1} \preceq \mathbf{1}_{n^P\times 1}, X^T\mathbf{1}_{n^P\times 1} \preceq \mathbf{1}_{n^Q\times 1}, \tag{2}$$

Here, $Score(\mathbf{x}) = \sum_{w_1,...,w_\delta} H_{w_1,...,w_\delta}\mathbf{x}_{w1}...\mathbf{x}_{w\delta}$, the product $\mathbf{x}_{w1}...\mathbf{x}_{w\delta}$ will be equal to 1 if the points $v^P_{i1}, ..., v^P_{i\delta}$ are all matched to the points $v^Q_{j1}, ..., v^Q_{j\delta}$, and 0 otherwise. $H_{w1,...,w\delta}$ is a similarity measure, which will be high if the sets of features $v^P_{i1}, ..., v^P_{i\delta}$ similar to the set $v^Q_{j1}, ..., v^Q_{j\delta}$. Using tensor notation [16], Equation(1) under the constraints (2) can be rewrite as:

$$\mathbf{x}^* = \arg\max_{\mathbf{x}}(H \otimes_1 \mathbf{x}... \otimes_\delta \mathbf{x}) \tag{3}$$

where the two-way constraints refer to the one-to-one matching from $\mathcal{G}^P$ to $\mathcal{G}^Q$.

2.2. Random Walk for hyper-Graph Matching.

The formulation of Equation(3) is called the integer quadratic programming (IQP), which is proven to be NP-complete, thus approximate solutions are required. Zass and Shashua proposed hyper-graph matching [17], which introduces a novel view that the matching problem and its corresponding solution are related by the Kronecker product. Tensor matching [16] is a higher-order extension of spectral matching [18]. It takes *rank*-1 approximation of the affinity tensor as a solution by using higher-order tensor power iteration. Chertok and Keller [19] also focused on rank-1 approximation of the affinity tensor by taking the leading left singular vector of the unfolded affinity tensor. However, all these are unable to effectively incorporate the matching constraints during their rank-1 approximation stage. Recently, Lee et.al.[15] proposed a novel state-of-the-art algorithm to solve (3) by ranking on an association hyper-graph via random walks, which effectively reflects matching constraints to produce a robust performance to large deformation and outlier noise. The entire the reweighted random walks scheme on the hyper-graph is is briefly described in Algorithm 1, see [15] for more details.

Algorithm 1 Reweighted Random Walks of Hyper-graph Matching

```
Input: The weight matrix $H$, the reweight factor $\alpha$, and the inflation factor $\beta$.
Output: The mapping matrix $\mathbf{x}$.
Initialize the mapping matrix $\mathbf{x}$ uniformly
$d_{max} = \max_w (H \otimes_2 1 ... \otimes_\delta 1)_w$
Initialize the transition matrix $P = H/d_{max}$
repeat
    $\overline{\mathbf{x}} = P \otimes_2 \mathbf{x} ... \otimes_\delta \mathbf{x}$
    $\mathbf{y} = exp(\beta\overline{\mathbf{x}}/max\,\overline{\mathbf{x}})$ {Reweighting with two-way constraints}
    repeat
        $\mathbf{y}_{ia} = \mathbf{y}_{ia} \,/\, \sum_{a=1}^{n^Q} \mathbf{y}_{ia}$ {normalize across rows}
        $\mathbf{y}_{ia} = \mathbf{y}_{ia} \,/\, \sum_{i=1}^{n^P} \mathbf{y}_{ia}$ {normalize across columns}
    until $\mathbf{y}$ converges
    $\mathbf{y} = \mathbf{y} \,/\, \sum \mathbf{y}_{ia}$
    $\mathbf{x} = \alpha\overline{\mathbf{x}} + (1-\alpha)\mathbf{y}$ {Random walking with reweighted jumps}
    $\mathbf{x} = \mathbf{x} \,/\, \sum \mathbf{x}_{ia}$
until $\mathbf{x}$ converges
Discretize $\mathbf{x}$ by the matching constraints
```

The reweighting procedure consists of two steps: inflation and bistochastic normalization. The inflation step of $exp(\beta\overline{\mathbf{x}}/max\,\overline{\mathbf{x}})$ attenuates small values of $\mathbf{x}$ and amplifies large values of $\mathbf{x}$. In this way, unreliable correspondences contribute insignificantly through the individual exponentials over the components of $\mathbf{x}$. Then, for the two-way constraint that a node in the graph G^P must correspond to only one node in the graph G^Q and vice versa. The bistochastic normalization step makes the current state distribution become more likely to be a permutation matrix, which satisfies the one-to-one constraints. In the final discretization step, any linear assignment algorithm can be adopted, such as a Greedy algorithm or the Hungarian algorithm.

3. **Proposed Method.**

This section describes the skeleton hyper-graph matching problem based on random walks, and the proposed algorithm in the framework is explained.

3.1. **Skeleton Graph Presentation.**

To better describe the proposed method, some common related skeletal concepts are clarified first. A set of skeleton points S is considered to be a continuous and connected set of the medial axis points of a closed curve, and they assumed that the skeleton curve is one pixel wide. Furthermore definitions of endpoint, connection point, skeleton branch, skeleton path and path distance could be found in [20]. Two points in a set are said to be connected if they can be joined by a continuous path of points also in the set. A skeleton point having only one adjacent point is an endpoint. The sequence of connection points between two endpoints is called a skeleton branch. A skeleton path $p(v_i, v_j)$ is the shortest skeleton branch between a pair of nodes v_i, v_j. Let $R(v_i, v_j)(t)$ be the radius of the maximal disk at the skeleton point with index t in $p(v_i, v_j)$. A vector of the radii of the maximal disks centered at the M sample points on $p(v_i, v_j)$ is denoted as:

$$R(v_i, v_j) = (R(v_i, v_j)(t))_{t=1,2...M} = (r_1, r_2, ..., r_M) : \tag{4}$$

where, $R(v_i, v_j)(t)$ is approximated with the values of the distance transform $DT(t)$ at each skeleton point with index t and they are normaled the following original shape S to make the proposed method invariant to the scale:

$$R(v_i, v_j)(t) = \frac{DT(t)}{\frac{1}{N_0}\sum_{i=1}^{N_0} DT(S_i)} \tag{5}$$

where $S_i(i = 1, 2, ..., N_0)$ varies over all N_0 pixels in the shape. The path distance between $p(v_i, v_j)$ and $p(v'_i, v'_j)$ is defined as:

$$pd(p(v_i, v_j), p(v'_i, v'_j)) = \sum_{k=1}^{M} \frac{(r_k - r'_k)^2}{r_k + r'_k} + \gamma\frac{(l - l')^2}{l + l'} \tag{6}$$

where γ is the weight factor and l and l' are the lengths of $p(u, v)$ and $p(u', v')$, respectively. Both the radii and the length are normalized to ensure that the path distance is scale invariant.

Then, we construct an skeleton graph $G = (V, E, A)$ by considering each endpoint as a node $v_i \in V$ for the graph, and the skeleton path between two nodes v_i and v_j as an edge $e_{ij} \in E$. Each edge attribute $a_{ij} \in A$ represents the geometric relationship between the node v_i and v_j in graph G. We present the skeleton graph G by distance matrix ω:

$$\omega = \begin{bmatrix} R(v_1, v_1) & R(v_1, v_2) & \cdots & R(v_1, v_N) \\ R(v_2, v_1) & R(v_2, v_2) & \cdots & R(v_2, v_N) \\ \vdots & & \ddots & \vdots \\ R(v_N, v_1) & R(v_N, v_2) & \cdots & R(v_N, v_N) \end{bmatrix} \tag{7}$$

where, N is numbers of nodes in G, each row of ω can be seen as a descriptor for a node which contains sequences to all other nodes of the same graph.

3.2. **Skeleton Graph Matching.** Given two skeleton graphs G^P with n^P nodes and and G^Q with n^Q nodes, their nodes are $V^P = \{v_1^P, ..., v_{n^P}^P\}$ and $V^Q = \{v_1^Q, ..., v_{n^Q}^Q\}$ respectively. The goal is to find node correspondences between two graph G^P and G^Q, which best preserve the attribute relations under the matching constraints. Previous work [20] only used node-to-node and pair-to-pair comparisons for their matching. However, in this work, we make a change of high-order relations of nodes, and compare the tuples of nodes. So, we define tensors representing high-order relations of nodes by hyper-graph formulations. Let $\mathcal{V}$ be a finite set of vertices and $\mathcal{E}$ a family of subsets of $\mathcal{V}$ such that $\cup_{e \in \mathcal{E}} = \mathcal{V}$. The SHG $\mathcal{G}= (\mathcal{V}, \mathcal{E}, \mathcal{A})$ consists of nodes $v \in \mathcal{V}$, hyper-edges $e \in \mathcal{E}$, and attributes $a \in \mathcal{A}$ associated with the hyper-edges. As illustrated in Fig.2, we construct SHG $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$ by considering each node as a centroid node and form a hyper-edge by a centroid and its k-nearest neighbors. A skeleton hyper-graph presents not only the local grouping information, but also the similarities between nodes in hyper-edges. In this way, the correlation information among nodes is more accurately described.

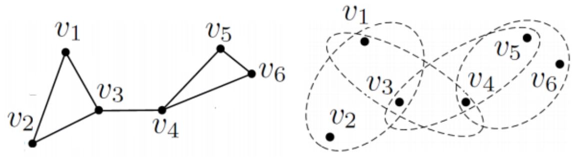

FIGURE 2. [16] Left: A simple graph with six nodes $\{v_1, ..., v_6\}$ in 2-D space. Right: A hyper-graph is built, in which each vertex and its 2 nearest neighbors form a hyper-edge.

The SHG matching problem between $\mathcal{G}^P$ and $\mathcal{G}^Q$ is equivalent to selecting reliable nodes in the graph $\mathcal{G}^w$ since the selected nodes in $\mathcal{G}^w$ correspondents to graph or subgraph matching between $\mathcal{G}^P$ and $\mathcal{G}^Q$. To select the nodes in $\mathcal{G}^w$, we adopt the statistics of the Markov random walks [15] which has been used to compute the ranking or relevance of graphs. Thus, graph matching between $\mathcal{G}^P$ and $\mathcal{G}^Q$ can be transformed into the node ranking and selection problem by random walks on $\mathcal{G}^w$. To solve such problem, we need to define an association SHG. As illustrated in Fig.3, we construct an association SHG $\mathcal{G}^w = (\mathcal{V}^w, \mathcal{E}^w, \mathcal{A}^w)$ by considering each candidate correspondence $c_w = (v_p^P, v_q^Q)$ as a node $v_w \in \mathcal{V}^w$. Here, a random walk from a node v_{w1} to another node v_{w2} on this graph $\mathcal{G}^w$ implies a walk from a correspondence c_{w1} to another correspondence c_{w2} between $\mathcal{G}^P$ and $\mathcal{G}^Q$. A hyper-edge $e_{w1,...,w\delta}$ in the association skeleton hyper-graph $\mathcal{G}^w$ embeds the similarity value $H_{w1,...,w\delta}$ of the δ-tuple and assigned as the attribute weight. For δ-th order skeleton hyper-graph matching, the similarities of the δ-tuples are measured by comparing attributes of two δ-th order hyper-edges $e^P_{p1,...,p\delta}$ and $e^Q_{q1,...,q\delta}$. Since each hyper-edge derived from δ-tuple of correspondences $c_{w1} = (v^P_{p1}, v^Q_{q1})$, ..., $c_{w\delta} = (v^P_{p\delta}, v^Q_{q\delta})$, we need to estimate the matching cost of each corresponding (v^P_{pi}, v^Q_{qi}). Consider two nodes $v_i^P = \{\omega^P_{i1}, \omega^P_{i2}, ...\omega^P_{in^P}\}$ and $v_j^Q = \{\omega^Q_{j1}, \omega^Q_{j2}, ...\omega^Q_{jn^Q}\}$, which are described by the i-th row of matrix ω^P and the j-th row of matrix ω^Q in Equation(7) correspondingly. We find an optimal correspondence $\varphi : \{\omega^P_{i1}, \omega^P_{i2}, ...\omega^P_{in^P}\} \rightarrow \{\omega^Q_{j1}, \omega^Q_{j2}, ...\omega^Q_{jn^Q}, \phi\}$, where $\omega_i^P \in v_i^P$ is mapped to $\varphi(\omega_i^P) \in v_j^Q$, and we allow a many-to-one mapping to ϕ. This is a sequence matching problem, so we compute all distances between the two sequences and obtain the matrix $\phi(\omega_i^P, \omega_j^Q)$ whose entry is the path distance between any two paths emanating from the two nodes.

$$\phi(\omega_i^P, \omega_j^Q) = \begin{bmatrix} pd(\omega^P_{i1}, \omega^Q_{j1}) & pd(\omega^P_{i1}, \omega^Q_{j2}) & \cdots & pd(\omega^P_{i1}, \omega^Q_{jn^Q}) \\ pd(\omega^P_{i2}, \omega^Q_{j1}) & pd(\omega^P_{i2}, \omega^Q_{j2}) & \cdots & pd(\omega^P_{i2}, \omega^Q_{jn^Q}) \\ \vdots & & \ddots & \vdots \\ pd(\omega^P_{in^P}, \omega^Q_{j1}) & pd(\omega^P_{in^P}, \omega^Q_{j2}) & \cdots & pd(\omega^P_{in^P}, \omega^Q_{jn^Q}) \end{bmatrix} \tag{8}$$

where $pd(.,.)$ is the distance between two skeleton paths computed by Equation(6). Since G^P and G^Q may have different numbers of nodes, standard comparison methods cannot effectively. We therefore need an algorithm to calculate the matching cost of feature vectors with different length. The Optimal Subsequence Bijection algorithm (OSB) [21] can work for elastic matching of two sequences of different lengths m and n. The goal of OSB is to find subsequences a' of a and b' of b such that a' best matches b', so it can be adapted to compute the dissimilarity of two nodes v_i and v'_j. By applying OSB to the matrix $\phi(\omega_i^P, \omega_j^Q)$, we obtain the dissimilarity of two nodes v_i^P and v_j^Q:

$$\Omega(\omega_i^P, \omega_j^Q) = OSB(\phi(\omega_i^P, \omega_j^Q) \tag{9}$$

Here, the optimal alignment of two series is not very considered much but rather the total matching cost is considered as a measure of their similarity. In order to define tensors representing high-order relations of nodes, we use the properties of the triangle formed by three points ($\delta = 3$) as shown in Fig. 4. The basic idea is that under a similarity transformation the angles of a triangle are unchanged [16]. In practice, we describe each triangle by the sines of its three angles. Thus, the similarity value $H_{w1,w2,w3}$ is defined as follows:

$$H_{w1,w2,w3} = exp\left[-\frac{1}{\sigma^2}\sum_{i=1}^{3}\|sin(\varphi^A_{wi}) - sin(\varphi^B_{wi})\|^2\right], \tag{10}$$

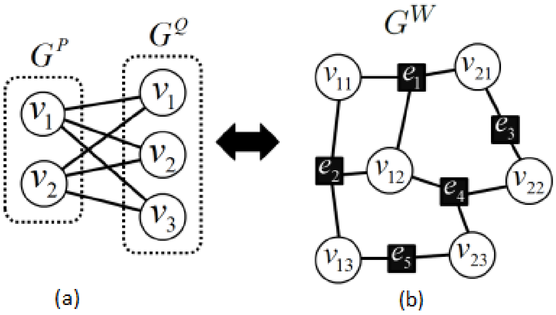

Figure 3. Six possible correspondences between two graphs of an association hyper-graph [15]. (a): There are two and three nodes in the graph G^P and G^Q, respectively. (b): Five hyper-edges which connect some of candidate correspondences.

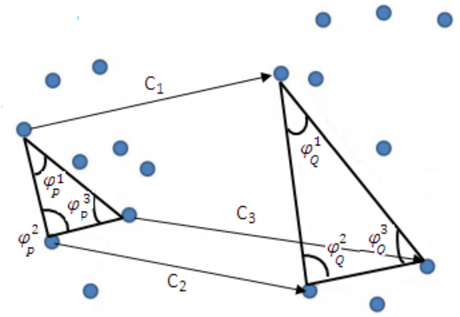

Figure 4. Three candidate correspondences C_1, C_2, C_3 form two triangles in calculating higher-order similarity [16]. Higher-order similarity can be calculated by comparing corresponding angles

where φ_{wi}^A and φ_{wi}^B denote angles of nodes related to the correspondence wi in the domain A and B, respectively. The construction of association skeleton graph $\mathcal{G}^w$ allows us to solve the skeleton hyper-graph matching problem by selecting reliable nodes $v_w \in \mathcal{V}^w$ in $\mathcal{G}^w$. By applying Algorithm.1 to the weight matrix in (10), we obtain the node correspondences between two graph G^P and G^Q.

4. **Experiment results.**

In this section, the experimental evaluation of the proposed method is presented in two parts:(1)matching the nodes in the skeleton graphs,(2) the retrieval performance of the proposed method on shape databases. Three shape database in the Kimia's 99 and Kimia's 216 dattabase that were provided by Kimia's group [22] and the MPEG-7 CE-Shape-1 (Part B) database [23] are used to test the quality performance of the proposed method. The experimental results are compared with those obtained from some popular shape matching methods [22, 24, 20, 25]. For each shape in database, we first compute the skeleton with the algorithm in [23] using default parameters. Then, we formulate the skeletons in the form of graphs and calculate the correlation between two graphs by the proposed method with the parameters M, γ, α, β, tuning empirically as 50, 40, 0.2 and 30, respectively.

4.1. **Correspondence matching.**

Several scenarios are tested for the object matching process which are shown in the figures 5, 6 and 7. Fig.5 shows the correspondence between two horses that one is the nonrigid transformation of the other one. Fig.6 illustrates that the proposed method works correctly if object parts are significantly altered (shortened in this case). Fig.7 demonstrates that the proposed method doesn't explicitly consider the topological structure of the skeleton graphs. The obtained correspondences demonstrate that our matching process has strong performance.

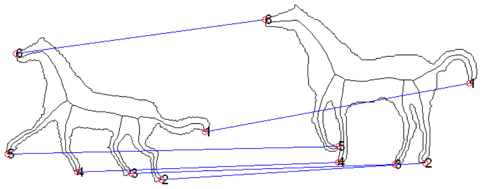

FIGURE 5. The correspondence in the presence of articulation

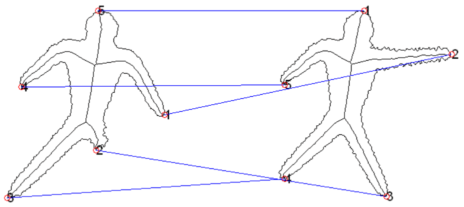

FIGURE 6. The correspondence between two persons with different numbers of legs

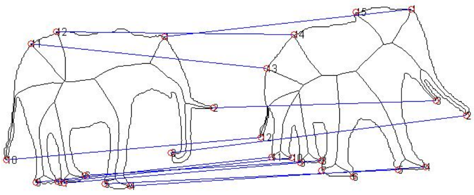

FIGURE 7. The correspondence between two elements that have not the same topological structure

4.2. **Robustness of retrieval.** Setting the same condition for running the proposed method with the other methods to compare the results. The experimental results of the proposed method are compared with those obtained from some popular shape matching methods such as: Graph Edit-distance [22], Path similarity [20], Inner Distance method [24], and Distortion-free Embedding [25].

Two Kimia's data sets are carried out to test the proposed method. The first data set (Kimia's 99) consists of 99 instances from nine classes, and each category has 11 shapes with some shapes have protrusions or missing parts. The second data set (Kimia's 216), which consists of 216 instances with 18 categories, and each category has 12 shapes with variations in form and occlusion of parts. Each shape in the database is used as a query, which is matched against all other shapes in the database. The difference between shapes is computed with the Hungarian algorithm on mapping matrix $\mathbf{x}$ in the final step of algorithm. Recognition scores are given as the number of kth closest matches that fall into the correct class, where k=1,...,10 for Kimia's 99 dataset and k=1,...,11 for Kimia's 216 dataset, respectively. The results are presented in Table 1 and Table 2 show that the proposed method performs comparably to the best methods reported on these datasets.

Clearly, the proposed method provides the results that outperforms the other methods. It is the best case of the obtained results from the methods based on the total number of mismatches in top 10 retrieval results on Kimia's 99 database and top 11 retrieval results on Kimia's 216 database are boldface. These results are recorded as 34, 32, 51, 28, 22 on Kimia's 99 database and 132, 95, 100, 67, 48 on Kimia's 216 database for Graph Edit-distance, Inner Distance, Path Similarity, Distortion-free Embedding and the proposed method, respectively.

Table 1. The number of kth closest matches that fall into the correct class are shown for several algorithms on Kimia's 99 database. The best possible score in each match is 99.

Algorithms	The number of $k-$th closest matches									
	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th
Shock Edit [22]	99	99	99	98	98	97	96	95	93	82
IDSC + DP [24]	99	99	99	98	98	97	97	98	94	79
Many-to-many [25]	99	99	98	99	98	97	98	97	92	84
Path Similarity [20]	99	99	99	99	96	97	95	93	89	73
The proposed method	99	99	99	99	98	97	97	97	**94**	**89**

Table 2. The number of kth closest matches that fall into the correct class are shown for several algorithms on Kimia's 216 database. The best possible score in each match is 216.

Algorithms	The number of $k-$th closest matches										
	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	11th
Shock Edit [22]	216	216	216	215	210	210	207	204	200	187	163
IDSC + DP [24]	216	216	215	216	215	210	210	209	205	190	179
Many-to-many [25]	216	216	216	216	215	213	210	209	206	199	193
Path Similarity [20]	216	216	215	216	213	210	210	207	205	191	177
The proposed method	216	216	216	216	216	215	211	211	**209**	**205**	**197**

Furthermore, we illustrate proposed method on the MPEG-7 dataset. This data set consists of 1400 silhouette images from 70 classes and each class has 20 shapes. Similar to the retrieval experiments presented above, every shape in the database is compared to all other shapes, and the number of shapes from the same class among the 40 most similar shapes are counted. The retrieval score is the ratio of the number of correct hits of all shapes to the complete matches hits, which was so-called the bull's eye score. The retrieval score is calculated as following:

$$\text{Retrieval\_score}=\frac{\sum_{i=1}^{N} S_i}{D \times N} \tag{11}$$

where S is the number of the first corrected hits in a class of similarity shapes, N is number of images in a data set, and D is number of the shapes in a class. The total possible most similar shapes were counted as (20 × 1400). Table 3 lists some of the reported results about the retrieval rates and average matching times of different approaches on the MPEG7 database. Fig.8 shows the numbers of each individual class. Obviously, the proposed method gives rise to the better results in most of the cases in comparisons with those obtained from other method of [20]. This is because such shape observes large variation in deformation, which can be captured by proposed method, but is difficult for the method [20].

Table 3. Retrieval rates and average matching times of different approaches for the MPEG7 dataset

Algorithm	Shock Edit [22]	IDSC + DP [24]	Path similarity [20]	Proposed method
Score %	87.70	85.40	86.70	**89.05**
Time (s)	-	-	0.09	**0.15**

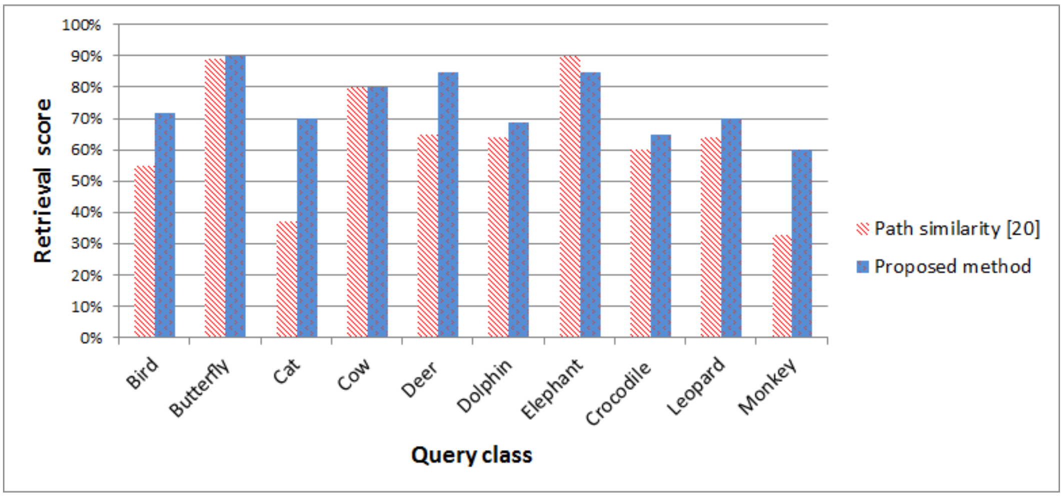

Figure 8. Detailed retrieval rate for some individual class.

5. **Conclusion.** In this paper, we proposed a method of matching points between two shapes based on their skeleton graph for the automatic object recognition applications. In the proposed method, a skeleton hypergraph model (SHG) was used to represent the relationships among relevant nodes. A centroid node was identified by computing based on its skeleton and a hyper-edge is formed by a hybrid between the centroid and k-nearest neighbor method. Matching between the two shapes was considered as a SHG matching problem. Through ranking on an association SHG in random walks, the best solution to this problem can be found out effectively, and reweighting scheme was used to establish correspondences between two SHGs. In experimental section, the silhouette database [22] and MPEG-7 database [26] were used to validate the proposed method quality. Experimental results show that the proposed approach clearly outperforms the other algorithms, especially in the presence of noise and outliers.

Acknowledgment. We would like to thank anonymous reviewers for their valuable comments and suggestions on the submission version of this paper. This work was supported in part by the Vietnam Academy of Science and Technology under the grant VAST.HTQT.Bulgaria.02/15-16: ”Improving the quality of an image management system by using machine learning methods”

REFERENCES

[1] M. Cho, J. Lee, and K. M. Lee, Feature correspondence and deformable object matching via agglomerative correspondence clustering, in *2009 IEEE 12th International Conference on Computer Vision*, pp. 1280–1287, Sept 2009.

[2] M. Gori, M. Maggini, and L. Sarti, Exact and approximate graph matching using random walks, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, pp. 1100–1111, July 2005.

[3] H. Blum, Biological shape and visual science (part i), *Journal of Theoretical Biology*, vol. 38, no. 2, pp. 205 – 287, 1973.

[4] W. B. Goh, Strategies for shape matching using skeletons, *Computer Vision and Image Understanding*, vol. 110, pp. 326–345, June 2008.

[5] T. B.Sebastian and B. B.Kimia, Curves vs. skeletons in object recognition, *Signal Processing*, vol. 85, no. 2, pp. 247 – 263, 2005.

[6] W. Shen, X. Bai, R. Hu, H. Wang, and L. Jan Latecki, Skeleton growing and pruning with bending potential ratio, *Pattern Recogn.*, vol. 44, no. 2, pp. 196–209, 2011.

[7] P. Felzenszwalb and J. Schwartz, Hierarchical matching of deformable shapes, in *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pp. 1–8, 2007.

[8] K. Siddiqi, A. Shokoufandeh, S. Dickinson, and S. Zucker, Shock graphs and shape matching, *International Journal of Computer Vision*, vol. 35, no. 1, pp. 13–32, 1999.

[9] D. Macrini, K. Siddiqi, and S. Dickinson, From skeletons to bone graphs: Medial abstraction for object recognition, in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pp. 1–8, 2008.

[10] D. Macrini, S. Dickinson, D. Fleet, and K. Siddiqi, Object categorization using bone graphs, *Computer Vision and Image Understanding*, vol. 115, pp. 1187–1206, Aug. 2011.

[11] D. Macrini, S. J. Dickinson, D. J. Fleet, and K. Siddiqi, Bone graphs: Medial shape parsing and abstraction, *Computer Vision and Image Understanding*, vol. 115, no. 7, pp. 1044–1061, 2011.

[12] H. Zaboli and M. Rahmati, An improved shock graph approach for shape recognition and retrieval, in *Modelling Simulation, 2007. AMS '07. First Asia International Conference on*, pp. 438–443, 2007.

[13] L. Han, R. Wilson, and E. Hancock, A supergraph-based generative model, in *Proceedings of the 2010 20th International Conference on Pattern Recognition*, pp. 1566–1569, 2010.

[14] X. Bai and L. J. Latecki, Path similarity skeleton graph matching, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 7, pp. 1282–1292, 2008.

[15] J. Lee, M. Cho, and K. M. Lee, Hyper-graph matching via reweighted random walks, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '11, pp. 1633–1640, 2011.

[16] O. Duchenne, F. Bach, I.-S. Kweon, and J. Ponce, A tensor-based algorithm for high-order graph matching, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 12, pp. 2383–2395, 2011.

[17] R. Zass and A. Shashua, Probabilistic graph and hypergraph matching, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, 2008.

[18] M. Leordeanu and M. Hebert, A spectral technique for correspondence problems using pairwise constraints, in *Proceedings of the Tenth IEEE International Conference on Computer Vision - Volume 2*, pp. 1482–1489, 2005.

[19] M. Chertok and Y. Keller, Efficient high order matching, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 12, pp. 2205–2215, 2010.

[20] X. Bai and L. J. Latecki, Path similarity skeleton graph matching, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 7, pp. 1282–1292, 2008.

[21] L. Latecki, Q. Wang, S. Koknar-Tezel, and V. Megalooikonomou, Optimal subsequence bijection, in *Proceedings of the Seventh IEEE International Conference on Data Mining*, pp. 565–570, 2007.

[22] T. Sebastian, P. Klein, and B. Kimia, Recognition of shapes by editing their shock graphs, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 5, pp. 550–571, 2004.

[23] X. Bai, L. Latecki, and W. yu Liu, Skeleton pruning by contour partitioning with discrete curve evolution, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 3, pp. 449–462, 2007.

[24] H. Ling and D. Jacobs, Shape classification using the inner-distance, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 2, pp. 286–299, 2007.

[25] M. F. Demirci, Y. Osmanlioglu, A. Shokoufandeh, and S. Dickinson, Efficient many-to-many feature matching under the l1 norm, *Computer Vision and Image Understanding*, vol. 115, pp. 976–983, July 2011.

[26] L. Latecki, R. Lakamper, and T. Eckhardt, Shape descriptors for non-rigid shapes with a single closed contour, in *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, vol. 1, pp. 424–429 vol.1, 2000.