

New Penalty Scheme for Optimal Subsequence Bijection

Laura Alejandra Pinilla-Buitrago, José Francisco Martínez-Trinidad,
and Jesús Ariel Carrasco-Ochoa

Instituto Nacional de Astrofísica, Óptica y Electrónica
Departamento de Ciencias Computacionales
Luis Enrique Erro # 1, Puebla, México
{laurapin, fmartine, ariel}@inaoep.mx

Abstract. Optimal Subsequence Bijection (OSB) is a method that allows comparing two sequences of endnodes of two skeleton graphs which represent articulated shapes of 2D images. The OSB dissimilarity function uses a constant penalty cost for all endnodes not matching between two skeleton graphs; this can be a problem, especially in those cases where there is a big amount of not matching endnodes. In this paper, a new penalty scheme for OSB, assigning variable penalties on endnodes not matching between two skeleton graphs, is proposed. The experimental results show that the new penalty scheme improves the results on supervised classification, compared with the original OSB.

Keywords: skeleton graph, classification, matching.

1 Introduction

In object recognition, the use of shape similarity based on skeleton matching usually performs better for articulated shape recognition than contour or other shape descriptors in the presence of partial occlusion and articulation of parts [1-4]. The skeleton integrates geometrical and topological features of an object, which makes it a good descriptor. Moreover, skeleton-based recognition methods compared with contour matching or other methods have lower sensitivity to articulation or rearrangement of parts [5].

In this work, we develop a method based on the work proposed in [5], which computes a dissimilarity value between skeleton graphs that represent articulated shapes. The dissimilarity value is computed from the distances between the shortest paths for each pair of endnodes in the skeleton graph, calculated by applying the OSB method, but unlike [6] where the same penalty cost is assigned for those not matching endnodes; in our proposal we include a variable penalty cost for all not matching endnodes.

Maintaining a fixed penalty, as proposed in the original OSB method, could have a negative effect in those cases where the penalty is a small value and the number of not matching nodes is big.

This paper is organized as follow: In the section 2 we review the related work. In the section 3 we explain the proposed modification to the OSB method. The section 4

shows experimental results of the proposed method compared against the original OSB. Finally, conclusions and future work are shown in section 5.

2 Related Work

In the literature, there are many approaches for computing skeleton similarity [7-12]. However one recent approach for computing shape similarity, which has shown good results, was proposed in [5]. This approach unlike other approaches, does not explicitly consider the topological structure of skeleton graphs. Instead, it is focused on the similarity of skeletons by comparing paths between skeleton endpoints. In this paper, we propose an extension to [5]. For this reason in this section we describe the original approach:

Definition 1. A skeleton point having only one adjacent point is called endpoint (skeleton endpoint); a skeleton point having three or more adjacent points is called a skeleton junction point. If a skeleton point is not an endpoint or a junction point, it is called a connection point.

Definition 2. The sequence of connection points between two directly connected skeleton points is called a skeleton branch.

Definition 3. In a skeleton graph the endpoints and junction points are chosen as the nodes for the graph, and all the skeleton branches between the nodes are the edges. An endpoint in a skeleton graph is called an endnode; a junction point is called a junction node.

Definition 4. The shortest path between a pair of nodes in a skeleton graph is called an skeleton path.

Suppose that we have a skeleton graph G with N endnodes, and $v_i (i = 1, 2, \dots, N)$ denotes the i th endnode along the shape contour labeled in clockwise direction, from the first contour pixel found in the y axis from top to bottom. $p(v_m, v_n)$ denotes the skeleton path from v_m to v_n nodes.

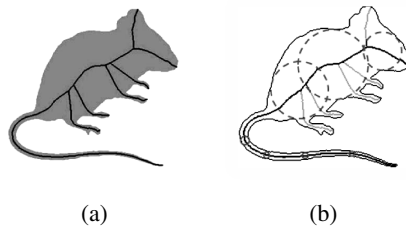


Fig. 1. (a) Skeleton of a shape (mouse) and (b) sequence of maximal disks between two endnodes on the skeleton path in (a). (Source: [13])

Given a set of M equidistant points on the path $p(v_m, v_n)$, $R_{m,n}(t)$ denotes the radius of the maximal disk (Fig. 1.) inside of the shape in the point t in $p(v_m, v_n)$.

The vector containing the radius of the maximal disk centered in the M equidistant points in $p(v_m, v_n)$ is denoted as:

$$R_{m,n} = (R_{m,n}(t))_{t=1,2,\dots,M} = (r_1, r_2, \dots, r_M) \tag{1}$$

Definition 5. *The shape dissimilarity between two skeleton paths is called a path distance. If R and R' denote the vectors of radii of two shape paths $p(u, v)$ and $p(u', v')$, respectively, the path distance is defined as:*

$$pd(p(u, v), p(u', v')) = \sum_{i=1}^M \frac{(r_i - r'_i)^2}{r_i + r'_i} + \alpha \frac{(l - l')^2}{l + l'} \tag{2}$$

Where l y l' are the lengths of $p(u, v)$ and $p(u', v')$ and α is a weight factor. The OSB method is scale invariant since both the path length and radii are normalized.

The *Optimal Subsequence Bijection* (OSB) method works over two sequences of different lengths m and n . The OSB method uses two finite sequences of endnodes of two skeletons $a = (a_1, \dots, a_m)$ and $b = (b_1, \dots, b_n)$ respectively. The aim is to find subsequences a' of a and b' of b such that a' best matches b' .

2.1 Dissimilarity between Two Endnodes

Let G and G' be two skeleton graphs to be matched, with endnodes $v_i (i = 0, 1, \dots, K)$ and $v'_j (j = 0, 1, \dots, N)$ respectively and $K \leq N$. The dissimilarity between the two endnodes v_i and v'_j is estimated by computing all path distances that emerge from nodes v_i in G and v'_j in G' , obtaining in this way a matrix of size $K \times N$.

The matrix in (3) contains the path distances (computed using (2)) that emerge from $v_i = v_{i0}$ in G and $v'_j = v'_{j0}$ in G' :

$$pd(v_{i0}, v'_{j0}) = \begin{pmatrix} pd(p(v_{i0}, v_{i1}), p(v'_{j0}, v'_{j1})) & \dots & pd(p(v_{i0}, v_{i1}), p(v'_{j0}, v'_{jN})) \\ pd(p(v_{i0}, v_{i2}), p(v'_{j0}, v'_{j1})) & \dots & pd(p(v_{i0}, v_{i2}), p(v'_{j0}, v'_{jN})) \\ \dots & \ddots & \dots \\ pd(p(v_{i0}, v_{iK}), p(v'_{j0}, v'_{j1})) & \dots & pd(p(v_{i0}, v_{iK}), p(v'_{j0}, v'_{jN})) \end{pmatrix} \tag{3}$$

In the order to obtain the dissimilarity value between two endnodes, the OSB method is applied to the matrix (3), obtaining the matrix (4) of size $(K + 1) \times (N + 1)$.

$$C(G, G') = \begin{pmatrix} OSB(pd(v_0, v'_0)) & \dots & OSB(pd(v_0, v'_N)) \\ OSB(pd(v_1, v'_0)) & \dots & OSB(pd(v_1, v'_N)) \\ \vdots & \ddots & \vdots \\ OSB(pd(v_K, v'_0)) & \dots & OSB(pd(v_K, v'_N)) \end{pmatrix} \tag{4}$$

For ensuring consistence in the matching between pairs of endnodes and to obtain the dissimilarity value between two shapes, which are represented by graphs, we applied OSB to the matrix $C(G, G')$ (4), as in [6].

In [6], at applying OSB in $C(G, G')$ the penalty for not matching endnodes is constant as proposed in the original method. In this case, we maintain the conditions that allow the link between the endnodes of skeletons, but, unlike the original OSB we apply a variable penalty cost on those not matching endnodes.

2.2 Optimal Subsequence Bijection

OSB works over two finite sequences of skeleton endnodes $a = (a_1, \dots, a_m)$ and $b = (b_1, b_2, \dots, b_n)$ for $|a| \leq |b|$. The main property of OSB is that some endnodes that form part of the sequences a and b can be jumped if it is necessary. However, excluding too many endnodes could influence in the dissimilarity value obtained. Each time that one endnode is jumped, a penalty is added. The cost of jumping some elements is called *jumpcost*. In [6] the jumpcost is constant when two specific sequences are compared and it is computed as:

$$jumpcost = mean_i \left(min_j \left(d(a_i, b_j) \right) \right) + std_i \left(min_j \left(d(a_i, b_j) \right) \right) \tag{5}$$

Where $d(a_i, b_j)$ is the dissimilarity value computed between two elements of a and b . In this paper, the distance d is the path distance *pd* defined in equation (2).

For computing the optimal correspondence between the sequences a and b we must enforce the following conditions:

$$d(a_i, b_j) \text{ is matched to } d(a_k, a_l) \text{ if and only if: } k - i = 1 \text{ and } j < l$$

Respecting the matching conditions and applying the penalty over not matching endnodes, we can define the matching cost as:

$$w((i, j), (k, l)) = \begin{cases} d(a_i, b_j) & \text{if } j + 1 = l \text{ y } i + 1 \leq k \\ (k + i - l).jumpcost & \text{if } j + 1 < l \text{ y } i + 1 \leq k \\ \infty & \text{Otherwise} \end{cases} \tag{6}$$

3 Variable Penalty

Unlike [5], we propose to use variable penalty for all not matching endnodes, in dependence of each element jumped in the sequence b .

Applying a variable penalty over not matching endnodes, allows that the penalty depends on the currently jumped endnode. If the distance computed from the jumped endnode to all the endnodes in the other skeleton graph is big, the penalty cost will be big and vice versa. The above, allow us to obtain a dissimilarity value smaller or bigger, depending on the amount and the kind of jumped endnodes between two sub-sequences.

Maintaining matching conditions of the original method and applying variable penalty cost over not matching endnodes, we can define the matching cost as:

$$w((i, j), (k, l)) = \begin{cases} d(a_i, b_j) & \text{if } j + 1 = l \text{ y } i + 1 \leq k \\ \sum_{j=j+1}^{l-1} \min_j C(i, j) & \text{if } j + 1 < l \text{ y } i + 1 \leq k \\ \infty & \text{Otherwise} \end{cases} \quad (7)$$

In this way, the total dissimilarity between skeleton paths or skeleton endnodes is given by the sum of the matching cost $w((i, j), (k, l))$.

4 Experiments

We built two databases for our experiments. The first database contains 36 shapes and 9 classes (4 shapes by class): *elephant*, *fork*, *heart*, *horse*, *human*, *L-star*, *star*, *tortoise* and *whale* (Fig. 2.). In this case we try to build a similar dataset to that one used in [6] which unfortunately is not available online.

The second database contains 32 shapes and 4 classes (8 shapes by class): *butterfly*, *cat*, *dolphin* and *fish* (Fig. 3.).

The shapes were obtained from different sources and the skeletons were semi-automatically constructed. First we compute an approximation to the skeleton of each figure, we used the source code by Bai¹ which uses *Discrete Skeleton Evolution* [14]. The skeleton achieved by this method allows us obtain an approximation to axis medial (or skeleton) of shapes, by means of the radii of maximal disk inside the shapes. Later, some branches were manually pruned, connected and/or added in order to ensure that the skeleton is simple and connected. Notice that this does not affect the modification proposed to the OSB method, since our method, in the same way as the original one, computes the dissimilarity between skeleton graphs once they have been built.

For our experiments on classification the *KNN* with $k = 1$, the proposed OSB with variable penalty and *leave-one-out cross validation* were used.



Fig. 2. Database containing 36 shapes and 9 classes

¹ <https://sites.google.com/site/xiangbai/>

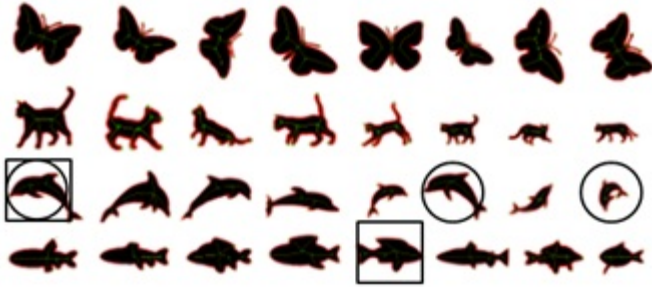


Fig. 3. Database containing 32 shapes and 4 classes

In Fig. 2 and Fig. 3, the errors obtained by applying OSB with constant penalty cost correspond to shapes labeled with a circle and shapes marked with squares are errors with variable penalty, which is proposed in this paper. In the table 1 we can observe the obtained accuracy.

Table 1. Accuracy obtained for two databases with the original OSB and the proposed modification

	Database Fig. 2	Database Fig.3
OSB with constant penalty cost	77.7%	90.62%
OSB with variable penalty cost	88.8%	93.7%

In both databases, the new penalty scheme for OSB obtains better results than the original OSB.

In order to explain why we get better results with the new penalty scheme, let us consider the shapes in Fig. 4.

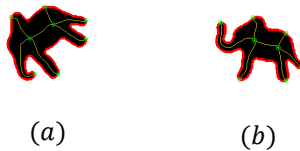


Fig. 4. The shape (a) is correctly classified with the proposed penalty scheme for OSB obtaining the smallest dissimilarity value with the shape (b)

The matrix (8) contains the dissimilarity values between endnodes for this shapes. In this matrix a complete correspondence is obtained (values marked) and therefore the total dissimilarity value between shapes (a) and (b) from Fig. 3. is 9,04 no matter if the original or the proposed OSB is applied.

$$\begin{pmatrix}
 1.82 & 14.93 & 12.90 & 24.98 & 37.67 & 14.95 \\
 12.36 & 3.16 & 3.49 & 13.52 & 22.30 & 16.08 \\
 9.73 & 1.40 & 0.90 & 5.41 & 12.70 & 11.01 \\
 13.48 & 3.76 & 3.56 & 1.30 & 4.40 & 7.89 \\
 27.67 & 8.37 & 8.89 & 7.66 & 1.98 & 20.52 \\
 4.27 & 12.83 & 10.38 & 12.14 & 19.54 & 2.34
 \end{pmatrix} \tag{8}$$

Nevertheless, if we consider the shapes in Fig. 5, the original OSB method would obtain a smaller dissimilarity value than the dissimilarity value obtained by the proposed variable penalty scheme, it will produce a mistake at classification stage. This is because, in the original OSB the *jumpcost* for each node jumped in the matrix (9) is 1,40. In this matrix, three endnodes are jumped and three endnodes are linked (values marked). Therefore the total value of dissimilarity for this matrix using the original OSB is 9.04.

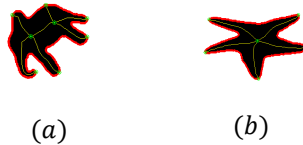


Fig. 5. The shape (a) is erroneously classified into class of star with the original OSB method, obtaining the smallest dissimilarity value with the shape (b).

$$\begin{pmatrix} 10.04 & 3.66 & 3.82 & \boxed{1.35} & 3.66 & 6.38 \\ 10.71 & 2.78 & 2.91 & 1.77 & \boxed{2.20} & 7.11 \\ 3.42 & 7.19 & 5.67 & 4.99 & 11.41 & \boxed{1.29} \\ 16.15 & 4.49 & 5.09 & 3.19 & 1.08 & 11.21 \\ 17.20 & 4.79 & 5.38 & 3.96 & 1.09 & 12.03 \end{pmatrix} \quad (9)$$

However, using the proposed variable penalty scheme the three nodes jumped in the matrix (9) have a total jump cost of 9,11 and the total dissimilarity between shapes (a) and (b) from Fig. 5. is 13,95. Since the last value is bigger than the value obtained with the original OSB, the new scheme allows to correctly classify the shape (a) from Fig. 5.

5 Conclusions and Future Work

Building skeletons is a technique that allows representing and catching structural information of shapes. In this work, we proposed a variable penalty scheme for the OSB comparison method. The experimental results show that using the proposed penalty scheme allows to get better classification results that those results obtained by the original OSB.

As future work, we propose, to include in the description of the skeletons other characteristics, in order to distinguish important segments inside the figures, which would allow to get better classification results.

Acknowledgment. This work was partly supported by the National Council of Science and Technology of Mexico (CONACyT) through the project grants CB2008-106443 and CB2008-106366; and the scholarship grant 283120.

References

1. Sebastian, T.B., Kimia, B.B.: Curves vs Skeleton in Object Recognition. *Signal Processing* 85(2), 247–263 (2005)
2. Basri, R., Costa, L., Geiger, D., Jacobs, D.: Determining the Similarity of Deformable Shapes. *Vision Research* 38, 2365–2385 (1998)
3. Huttenlocher, D.P., Klandeman, G.A., Rucklidge, W.J.: Comparing Images Using the Hausdorff Distance. *IEEE Transaction Pattern Analysis and Machine Intelligence* 15(9), 850–863 (1993)
4. Belongie, S., Puzhicha, J., Malik, J.: Shape Matching and Object Recognition Using Shape Contexts. *IEEE Transaction Pattern Analysis and Machine Intelligence* 24(4), 509–522 (2002)
5. Bai, X., Latecki, L.: Path Similarity Skeleton Graph Matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30, 1282–1292 (2008)
6. Shen, W., Wang, Y., Bai, X., Wang, L., Latecki, L.: Shape Clustering Common Structure Discovery. *Pattern Recognition* 64, 539–550 (2013)
7. Zhu, S.C., Yuille, A.L.: FORMS: A Flexible Object Recognition and Modeling System. *Proceedings of International Journal of Computer Vision* 20(3), 187–212 (1996)
8. Liu, T., Geiger, D.: Approximate Tree Matching and Shape Similarity. In: *Proceedings of IEEE 7th International Conference on Computer Vision*, pp. 456–462 (1999)
9. Siddiqi, K., Shkouflandeh, A., Zucker, S.: Shock Graphs and Shape Matching. *Proceedings of International Journal of Computer Vision* 35, 13–32 (1998)
10. Shokoufandeh, A., Macrini, D., Dickinson, S., Siddiqi, K., Zucker, S.W.: Indexing Hierarchical Structures Using Graphs Spectra. *IEEE Trans. Pattern Analysis and Machine Intelligence* 27, 1125–1140 (2005)
11. Sebastian, T.B., Klein, P., Kimia, B.B.: Recognition of Shapes by Editing Shocks Graphs. In: *Proceedings of International Conference in Computer Vision*, pp. 755–762 (2001)
12. Torsello, A., Hancock, E.R.: A Skeletal Measure of 2D Shape Similarity. *Computer Vision and Image Understanding* 95, 1–29 (2004)
13. Bai, X., Liu, W., Tu, Z.: Integrating contour and skeleton for shape classification. In: *IEEE 12th International Conference on Computer Vision Workshops* (2009)
14. Bai, X., Latecki, L.J.: Discrete Skeleton Evolution. In: Yuille, A.L., Zhu, S.-C., Cremers, D., Wang, Y. (eds.) *EMMCVPR 2007*. LNCS, vol. 4679, pp. 362–374. Springer, Heidelberg (2007)