Engineering Applications of Artificial Intelligence 24 (2011) 1295–1299

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

ASOD: Arbitrary shape object detection

Selim Mimaroglu *, Ertunc Erdil

Department of Computer Engineering, Bahcesehir University, Ciragan Caddesi 34353 Besiktas, Istanbul, Turkey

ARTICLE INFO

Article history:
Received 15 March 2011
Received in revised form
18 May 2011
Accepted 12 July 2011
Available online 4 August 2011

Keywords:
Object detection
Shape detection
Clustering
Pattern recognition
Static images

ABSTRACT

Arbitrary shape object detection, which is mostly related to computer vision and image processing, deals with detecting objects from an image. In this paper, we consider the problem of detecting arbitrary shape objects as a clustering application by decomposing images into representative data points, and then performing clustering on these points. Our method for arbitrary shape object detection is based on COMUSA which is an efficient algorithm for combining multiple clusterings. Extensive experimental evaluations on real and synthetically generated data sets demonstrate that our method is very accurate and efficient.

1. Introduction

Automatically detecting arbitrary shape objects can be widely used in civil engineering for designing roads, bridges, and buildings; mechanical engineering for designing engines, aircraft and transportation products; biomedical engineering for designing new diagnosis tools and improving existing ones; mining engineering for spotting, extracting, and processing minerals.

Clustering has so many applications in a variety of disciplines. Some recent applications of clustering can be found in González and García (2006), Elfelly et al. (2010) and García and González (2004). Clustering algorithms has been rarely applied successfully to shape and edge detection. Detecting ring-shaped objects is studied in Man and Gath (2002). Nosovskiy et al. (2008) and Theoharatos et al. (2005) propose boundary detection methods. A 3D object recognition algorithm is proposed in de Trazegnies et al. (2003). Two schemes for palm print identification are proposed in Jia et al. (2008). At another direction, Belongie et al. (2002) measures similarity between shapes and uses it for object recognition. Some other interesting artificial intelligence techniques for engineering applications can be found in Huang and Chau (2008) and Zhang and Chau (2009).

Arbitrary shape object detection, which is a very challenging problem by its nature, can benefit from well established concepts in clustering methodology. Unfortunately, existing sophisticated clustering techniques cannot be successfully applied due to several reasons. Unsatisfactory accuracy results obtained on challenging data sets by clustering methods such as k-means, k-medoids, agnes, etc. is the leading factor for avoiding clustering techniques on arbitrary shape objects. Another disadvantage is the need for the number of output clusters in advance by many clustering algorithms, which is unknown in most cases. Our main contribution is a new arbitrary shape object detection (ASOD) algorithm, which provides very accurate results and automatically detects the number of objects in a data set. ASOD does not make any assumptions about the input, thus it can be applied widely to many engineering disciplines. ASOD is based on a combining multiple clusterings method known as COMUSA (Mimaroglu and Erdil, 2011).

Our paper is structured as follows: In Section 2, ASOD is introduced in detail. Section 3 provides experimental evaluations. In the final section, conclusions and future work are presented.

2. Arbitrary shape object detection (ASOD) algorithm

In this section we explain the general characteristics of arbitrary shape object detection (ASOD), which is presented in Algorithm 1. Our method decomposes the input data into point cloud, and groups the points based on their proximity. Each cluster generated by ASOD represents a distinct object in the input data—the number of clusters is found automatically.

In line 1, objects are decomposed into representative data points using various techniques presented in the literature such as Eldar et al. (1997), Akra et al. (1999), and Sclaroff and Pentland (1995), or uniform distribution. In our experiments we decompose the input data into representative points having equal (or close to equal) density. Following this step, a distance graph of data points is constructed via Euclidean distance, where each edge weight weight (v_i,v_j) shows the distance between corresponding vertices

* Corresponding author. Tel.: +90 212 381 05 55; fax: +90 212 381 05 50.
E-mail addresses: selim.mimaroglu@bahcesehir.edu.tr (S. Mimaroglu), ertunc.erdil@bahcesehir.edu.tr (E. Erdil).

0952-1976/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.engappai.2011.07.005

v_i and v_j. In this graph, each vertex (point) is represented by a pair (df(v_i),sw(v_i)): df(v_i) shows the *degree of freedom* and is the number of edges incident to v_i and sw(v_i) is the sum of weight of edges incident to v_i. The attachment of a vertex v_i is defined as attachment $(v_i) = \mathrm{sw}(v_i)/\mathrm{df}(v_i)$. An unmarked point having the lowest attachment value is chosen as pivot (seed) for initiating a new cluster. Low values of sum of weights indicate low distance with other points. Therefore, a point having low attachment value is strongly connected to its surrounding points, and it is a good place to start a cluster. Direct neighbors of a pivot are considered to expand a cluster (lines 16–21), where each cluster represents an arbitrary shape object. Expansion of a cluster continues up until pivots cannot add any other points into the cluster. If there are remaining unmarked data points, ASOD initiates a new cluster by determining new pivot point and expands the cluster similarly. When all the data points are assigned into a cluster, ASOD terminates. Each cluster constructed by ASOD represents a distinct object, and arbitrary shape object detection is performed in this manner by using our algorithm. Furthermore, number of objects is detected automatically and correctly by ASOD.

Algorithm 1. Arbitrary Shape Object Detection (ASOD)

Input: O: A set of objects
Output: Clustering—each object as a cluster
1 Decompose O into data points D;
2 Initialize an empty queue Q;
3 *clusterId* = 1;
4 Construct distance graph $DG = (D,E)$ using D;
5 Sort D in increasing order with respect to attachment;
6 **while** *there are unmarked objects* **do**
7   Add unmarked object, d_i, with lowest attachment(d_i) to Q;
8   **while** *Q is not empty* **do**
        // pivot object
9     $v =$ remove first element from Q;
10   Add v to cluster *clusterId*;
11   Mark v;
12   **foreach** $(w,v) \in E$ **do**
13   **If** *w is marked* **then**
14   continue;
15   **else**
16   *lghtWeight* = weight(w,v);
17   *isMin* = *true*;
18   **foreach** $(z,w) \in E$ **do**
                    // *minimum* constraint
19   **if** *lghtWeight* $\not\leq$ weight(z,w) **then**
20   *isMin* = *false*;
21   break;
22   **if** *isMin* **then**
23   Add w to Q;
24   *clusterId*++;

2.1. Relaxation

In some cases there can be larger objects for detection: user input relaxation rate is used to achieve this, which is defined as follows. On a set of edge weights $\{w_1, w_2, \ldots, w_n\}$, w_k is said to have minimum value with relaxation r, if $\forall_i(w_k - w_k.r \geq w_i)$ holds. Minimum edge weight constraint is relaxed with the relaxation value, r.

Experimental results demonstrate that the relaxation parameter affects the accuracy of ASOD (see Table 2). Increasing this parameter enables ASOD to assign more points into a cluster by relaxing the minimum edge weight constraint. Thus, larger clusters are obtained which may be advantageous depending on the characteristics of the input data set.

2.2. Improvements of ASOD over COMUSA

Although ASOD is based on COMUSA, ASOD and COMUSA serve completely different purposes. ASOD is a clustering technique which is designed for arbitrary shape object detection, whereas COMUSA is used for combining multiple clusterings.

ASOD works on an input data set representing objects, however COMUSA takes a collection of clusterings of a data set—not the data set itself—as its input.

Finally, ASOD operates on a distance graph constructed by pairwise Euclidean distance between points, but COMUSA operates on a similarity graph of objects which is built by the evidence accumulated in the collection of input clusterings.

3. Experimental evaluations

In this section we present an objective cluster quality measure, experimental data sets, and test results.

3.1. Adjusted rand index (ARI) for evaluating quality

We use adjusted rand index (ARI) in order to measure the extent to which the clustering structure discovered by ASOD matches some external criteria, i.e. class labels which correctly identifies each object. Given a data set of points $D = \{d_1, \ldots, d_n\}$, suppose $U = \{u_1, \ldots, u_r\}$ represents classes (real objects), and $V = \{v_1, \ldots, v_p\}$ represents a clusterings of the D (our findings):

$$\bigcup_{i=1}^{r} u_i = D = \bigcup_{j=1}^{p} v_j$$

and $u_i \cap u_j = \emptyset$ for $1 \leq i,j \leq r$ and $i \neq j$. Also, $v_i \cap v_j = \emptyset$ for $1 \leq i,j \leq p$ and $i \neq j$.

In Table 1, $n_{ij} = |u_i \cap v_j|$, $n_{i.} = \sum_{j=1}^{p} n_{ij}$, and $n_{.j} = \sum_{i=1}^{r} n_{ij}$.

ARI can be formulated as follows:

$$\frac{\sum_{i,j}\binom{n_{ij}}{2} - \left(\sum_i \binom{n_{i.}}{2}\sum_j \binom{n_{.j}}{2}\right)/\binom{n}{2}}{\frac{1}{2}\left(\sum_i \binom{n_{i.}}{2} + \sum_j \binom{n_{.j}}{2}\right) - \left(\sum_i \binom{n_{i.}}{2}\sum_j \binom{n_{.j}}{2}\right)/\binom{n}{2}}$$

ARI takes maximum value at 1, which indicates perfect match to the external criteria.

Table 1
Contingency table.

Class	Cluster				Sums
	$\mathbf{v_1}$	$\mathbf{v_2}$	…	$\mathbf{v_p}$	
$\mathbf{u_1}$	n_{11}	n_{12}	…	n_{1p}	$n_{1.}$
$\mathbf{u_2}$	n_{21}	n_{22}	…	n_{2p}	$n_{2.}$
⋮	⋮	⋮		⋮	
$\mathbf{u_r}$	n_{r1}	n_{r2}	…	n_{rp}	$n_{r.}$
Sums	$n_{.1}$	$n_{.2}$		$n_{.p}$	$n_{..} = n$

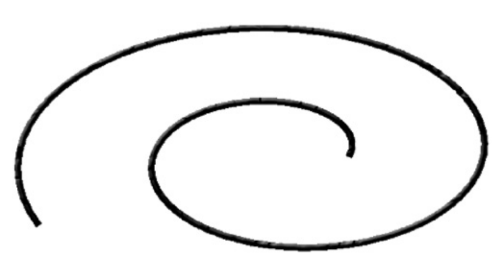

Fig. 1. A spiral shape object.