Int. J. Electron. Commun. (AEÜ) 68 (2014) 260–269

Contents lists available at ScienceDirect

International Journal of Electronics and Communications (AEÜ)

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

# Automatic segmentation of color images with transitive closure

M. Ozan Incetas [a], Recep Demirci [b,*], H. Guclu Yavuzcan [a]

[a] Industrial Technology Education Department, Industrial Arts Education Faculty, Gazi University, Ankara 06370, Turkey
[b] Computer Engineering Department, Technology Faculty, Gazi University, Ankara 06500, Turkey

ARTICLE INFO

*Article history:*
Received 3 November 2012
Accepted 10 September 2013

*Keywords:*
Image segmentation
Region merging
Transitive closure

ABSTRACT

In this paper, a mask based automatic segmentation algorithm for color images which uses pixel similarity has been presented. Main concept of the algorithm relies on spatial mask for course segmentation and the Warshall's transitive closure (TC) computation algorithm for region merging. Although the proposed spatial mask approach reduces the computational burden required for segmentation or clustering techniques such as seeded region growing (SRG) or fuzzy c-means (FCM) in which user supplied parameters are essential, it has over segmentation drawback. Therefore, the transitive closure algorithm, which uses adjacency and similarity matrix associated to undirected graph of the over segmented image, has been employed to merge the regions. After comparing to existing methods, the obtained experimental results confirmed that the color images as well as gray level images could be segmented with considerable accuracy. Also computational complexity of image segmentation is significantly reduced. Furthermore, there is no need any user supplied parameter such as the number of clusters or seed points.



## 1. Introduction

Segmentation is still one of the main difficulty in the image processing field. It was observed in literature that variety of different methods was developed to solve the problem. Some of these methods claim that they do more accurate segmentation whereas some claim that they do faster segmentation than others. In general, edge, region or clustering-based techniques have been considered for the segmentation applications. Clustering, region based segmentations and graph-based merging approaches are especially discussed below in terms of computational complexity.

The *K*-means and FCM are common image segmentation algorithms based on clustering approach. These techniques are quite successful in clustering of images having the certain number of clusters. Nevertheless pixel classification could be incorrect in consequence of an overlapping in color space of adjacent clusters [1]. When the number of clusters is unknown which is typical for segmentation process, clustering is more difficult [2]. Furthermore these algorithms are iterative and so pixels are passed out more than once. As the number of iteration increases and time consuming of process grows [3]. The computational complexity of FCM is $O(\boldsymbol{nkl})$ where $\boldsymbol{n}$, $\boldsymbol{k}$ and $\boldsymbol{l}$ is the number of pixels, clusters and iteration, respectively [4].

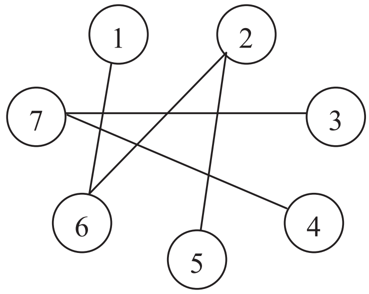
The relationships among these parameters are given by $\boldsymbol{l} \ll \boldsymbol{n}$ and $\boldsymbol{k} \ll \boldsymbol{n}$ so the computational complexity takes $\approx O(\boldsymbol{n})$. However, since the centers of clusters are set randomly, clustering process needs to be repeated more than once to reach correct results [5]. The time consuming increases. Also, since FCM is a clustering method rather than segmentation, it is necessary to employ another procedure with FCM to complete the segmentation process. So, the computational complexity and the time consuming will be increased.

On the other hand, region-based segmentation techniques are based on finding adjacent pixels with similar features [6]. They generate a segmentation map starting with small regions known as seed [7,8]. Neighboring pixels are evaluated to grow these seeds to larger regions. If a pixel sufficiently similar to an adjacent region, this pixel is included to the region. The seed regions are created either automatically or selected by user. Automatic creation of these seeds brings an additional computational cost. Additionally, SRG approach has order dependency. The insertion order of two pixels with same features into a region affects the performance of segmentation process. If the order dependency is eliminated with any additional algorithm, the processing time is also increased [9].

It is observed in some studies that the over segmentation intricacy occurs after the automatic segmentation applications and a merging process is required to overcome this issue [10–13]. Therefore, utilizing advantage of the graph theories for the region merging process is quite common [10,14–19]. Especially, the approaches, which use adjacency matrix of regions, have been general solutions [17,19–21]. Those algorithms are recursive and their

* Corresponding author. Tel.: +90 3122028566.
*E-mail addresses:* oincetas@ankara.edu.tr (M.O. Incetas), rdemirci@gazi.edu.tr (R. Demirci), gyavuzcan@gazi.edu.tr (H.G. Yavuzcan).

1434-8411/$ – see front matter © 2013 Elsevier GmbH. All rights reserved.
http://dx.doi.org/10.1016/j.aeue.2013.09.001




(a)

| i \ j | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 2 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| 3 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 4 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 5 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 6 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| 7 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |

(b)

**Fig. 1.** A typical graph and its matrix (a) an undirected graph (b) symmetric matrix.

computational complexities are O($\boldsymbol{n}^2$). However, using weighted graphs has a disadvantage due to similarity computation load for all nodes [22]. Furthermore, one of the main dilemmas in merging approaches is that result of merging is not symmetric. If there is a path between two pixels having adjacent pixels which are similar to them, these pixels should be in the same region after the segmentation process in symmetric segmentation approach [23]. On the other hand, solving the symmetric segmentation problem increases the number of process, so it increases the computational time.

In this study, we present a novel approach consisting of two steps to solve the time consumption and symmetric segmentation problems. In the first step, a spatial mask based segmentation algorithm for the gray level images with geometrical shapes, which was developed by Demirci is utilized [2]. Although this method is quite fast, it has an over segmentation problem because of spatial mask used. In order to eliminate the drawback of the algorithm, a graph based region merging process is integrated in the second stage of proposed approach. Initially, each region in over segmented image is represented by a node and accordingly an adjacency matrix is established. Then the transitive closure (TC) of adjacency matrix related to course segmentation graph is calculated via Warshall's algorithm. Consequently, the merging procedure with TC matrix is implemented to achieve reasonable segmentation results and computational complexity of segmentation process is decreased. Furthermore, recently, the use of the TC concept in image segmentation field has become innovative and it may allow new ideas on region merging process.

## 2. Transitive closure

Let $R$ be a binary relation on a set $A$. The binary relation $R$ is a set of ordered pairs of the elements from set $A$. If the ordered pair $(a, b)$ is in the relation $R$, it is usually written $aRb$. The transitivity in a relation implies that if $aRb$ and $bRc$ for all $a$, $b$ and $c$ then $aRc$. If new elements that ensure the requirement mentioned above for all ordered pairs are added to relation $R$, this new relation is named transitive closure of relation $R$ and this is denoted by $R^*$. Mathematically, transitive closure of relation $R$ is the smallest relation $R^*$ such that $R \subseteq R^*$ and $R^*$ is transitive on the set $A$ [24]. That is, the transitive closure of the relation $R$ is the transitive relation by adding the minimum number of ordered pair to $R$. If the relation $R$ is transitive, there is no need to add a pair to $R$. For example, let $R = \{(a,b), (a,c), (b,c), (b,d)\}$ be a relation on a set $A = \{a, b, c, d\}$. This relation is not transitive, because ordered pairs $(a,b)$ and $(b,d)$ are in $R$, but ordered pair $(a,d)$ is not. Therefore, it is clear that the relation $R^* = \{(a,b), (a,c), (b,c), (b,d), (a,d)\}$ which is obtained by adding ordered pair $(a,d)$ to relation $R$ is TC of the relation $R$. Since the ordered pairs $(a,b)$ and $(b,c)$ are in the relation $R^*$, ordered pair $(a,c)$ is in it. In the same way, ordered pair $(a,d)$ is in $R^*$ as ordered pairs $(a,b)$ and $(b,d)$ are in it.

A binary matrix can be used to define a relation. For example, if ordered pair $(a, b)$ is in a relation $R$, 1 is inserted to the matrix in the location corresponding this pair, elsewhere 0 is inserted. The same matrix representation can be used for graphs. Let $\boldsymbol{G} = (\boldsymbol{V},\boldsymbol{E})$ is a graph where $\boldsymbol{V}$ is the set of vertexes (nodes) and $\boldsymbol{E}$ is the set of edges. If there is an edge between the vertexes $\boldsymbol{i}$ and $\boldsymbol{j}$ of graph $\boldsymbol{G}$ then $\boldsymbol{M}(\boldsymbol{i},\boldsymbol{j}) = 1$ elsewhere $\boldsymbol{M}(\boldsymbol{i},\boldsymbol{j}) = 0$ in matrix representation. An example of undirected graph $\boldsymbol{G}$ and the matrix $\boldsymbol{M}$ representing this graph are given in Fig. 1(a) and (b) respectively. Since the graph $\boldsymbol{G}$ is undirected, the matrix $\boldsymbol{M}$ is symmetric. The first algorithm to compute the TC of a binary relation matrix was developed in 1962 by Warshall [25]. Pseudo codes related to this algorithm are given as follows:

**Algorithm 1.** Warshall's algorithm

```
1.  M* = M;
2.  for i = 0 to n−1 do
3.    for j = 0 to n−1 do begin
4.      if M*[i][j] = 1 then do begin
5.        for k = 0 to n−1 do begin
6.          if M*[j][k] = 1 then do begin
7.            set M*[i][k] = 1;
8.          end;
9.        end;
10.     end;
11.   end;
12. end.
```

In the first line of Algorithm 1, the content of matrix M is copied into M*. Then, the matrix M* which is TC of the matrix M is computed by adding new edges between every vertex $i$ and other vertexes have an edge between the vertex $j$ which has an edge with vertex $i$ in the next lines of Algorithm 1.

Graph G* obtained using Algorithm 1 and the matrix M* presentation of the G* are shown in Fig. 2(a) and (b), respectively. The loops are not shown in graph G* as they are not used in this study. Also, new edges are shown with bold lines. Similarly, the new elements of the matrix M* in Fig. 2(b) is represented by bold characters. Although the computational complexity of original Warshall's algorithm is O($\boldsymbol{n}^3$), it was reduced to O($\boldsymbol{n}^2$) level in the recently developed and more complex TC computation algorithms [26,28].

## 3. Pixel similarity

The proposed method in this study is based on the assignment of similar and adjacent pixels into the same regions so that homogenous regions could be obtained. Therefore the similarity of neighboring pixels is an important criterion to be considered. Let $P_1$ and $P_2$ be neighbor pixels and their RGB components are $R_1$,