Kangweon-Kyungki Math. Jour. **14** (2006), No. 1, pp. 47–55

APPLICATION OF FLOYD-WARSHALL LABELLING TECHNIQUE: IDENTIFICATION OF CONNECTED PIXEL COMPONENTS IN BINARY IMAGE

Hyunkyung Shin* and Joong Sang Shin

Abstract. This paper presents an image analysis technique using connected components by applying SPA (shortest path algorithm) from graph theory. Software implementation of Floyd-Warshall transitive matrix labelling method is discussed. Results are also presented.

1. Introduction

Image segmentation is a generic term for separation between an object and background material residing in an image. The primary purpose of segmentation is to remove undesirable artifacts from bonafide object components in the image[12]. As the elementary exemplar tools for image segmentation, various threshold techniques and boundary/edge detection methods are proposed [4]. In practice, nonetheless, labelling is a primary segmentation method so that each object labelled differently can be analyzed separately. However, its negative aspect is operational ineffectiveness due to complexity of underlying labelling algorithm. Reducing operational complexity of the algorithm is of great importance. Connected components of vertices using labelling method is a fundamental concept in graph theory appeared first in [10]. Its definition states as follows,

Definition 1.1. In an undirected graph, a connected components is a maximal connected sub-graph. Two vertices are in the same connected components if and only if there exists a path between them.

Received December 26, 2005.

2000 Mathematics Subject Classification: 68R10, 68U10, 90C35.

Key words and phrases: Image Processing, Image Analysis, SPA, Connected Components.

* Corresponding author

A path between two vertices u and v, appeared in Definition 1.1, represents a set of edges starting from u and ending to v if it exists. Path in graph can be formalized in algebraic terms.

Definition 1.2. In an undirected graph, the existence of path between two vertices u and v is an equivalence class.

Image is a two dimensional array of pixels, where pixel is abbreviation of picture element. Pixel is an unit of picture representation in a computer memory, typically has one (Gray), three (RGB color) or four dimensions (Printing devices). Each dimension of pixel usually takes 1 (document, photo) or 2 (medical, GPS) bytes. In binary image, pixel has one dimension and only two pixel values are available, black(1) and white(0).

Definition 1.3. Two dimensional binary image having size of width, w, and height, h, can be written as

$$a \in \{0, 1\}^X,$$

where X indicates $w \times h$ grid. $a(i, j)$ indicates the pixel value at i-th column and j-th row of image array a.

There are two types of connectivity which defines topology of neighborhood from a pixel: 4-connectivity (North, West, South, East) and 8-connectivity (N, NW, W, SW, S, SE, E, NE). Neighborhood of a pixel y, $N(a)(y)$ is formulated in Definition 1.4, for details refer to[10],

Definition 1.4. $[N(a)]_4(y), [N(a)]_8(y)$ represent 4- and 8- connected neighborhood of y, respectively.

$$\begin{aligned}
&[N(a)]_4(y)\\
&= \{y\} \cup \{x \mid x = y + (0, i) \text{ or } y + (i, 0) : a(x) \neq 0,\ i = 1 \text{ or } -1\}\\
&[N(a)]_8(y)\\
&= \{y\} \cup \{x \mid x = y + (i, j)\ \ : a(x) \neq 0,\ i,\ j = -1 \parallel 0 \parallel 1\}
\end{aligned}$$

Two pixels in an image are connected if there exists a recursive path of neighborhood pixels to the other pixels. Existence of path needs to be formalized mathematically, With the configurations described above, it is obvious that connected components in a binary image can be well-defined.

Component labelling is originated from the algorithm by Rosenfeld and Pfalz[11]. The algorithm performs in two steps: the first pass initializes the labels for each component, and the second pass finds equivalent labels to merge into one. As an application of the labelling algorithm to image analysis, image labelling problem is to uniquely label all of the connected components in an image. Several researchers[7, 6, 8] drew attention a structural problem for the second pass when input image is huge. In worst case, the size of equivalence array can be out of range. The various efforts to resolve the problem in identifying equivalence labels have been attempted. To resolve physical memory problem, parallel algorithms have been proposed [1, 2, 13]. For a comparative survey of the parallel algorithms for image labelling, refer to [5]. A serial algorithm, use a bracket table to associate equivalent set in [14], or grow connected components from a seed pixel in [8], or trace the boundaries of connected components in [3].

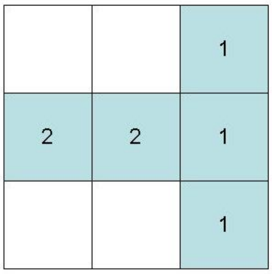
In this paper, we develop a fast and serial connected pixel components algorithm in binary image by applying Floyd-Warshall SPA. In §2 the details of algorithm implementation is discussed. In §3 the results and discussions are presented.

2. Implementation of Algorithm

Our algorithm using Floyd-Warshall transitive matrix performs in three steps: Step 1, image labelling in first pass throughout image from top to bottom and left to right, Step 2, set up transitive matrix in second pass throughout image from bottom to top and left to right, Step 3, merge labels without referencing image.

In step 1, each black pixel in an input image of size $w \times h$, $a(i,j)$, will be visited from top to bottom and left to right.

```
for $j = 0 \to h$
    for $i = 0 \to w$
        label $a(i,j)$
```



		1
2	2	1
		1

FIG. 1. As a typical result of initial labelling, a component can have multiple labels due to top-bottom, left-right directional partiality while pixel sweeping.

where the “label” function performs labelling for each black pixels in image. The functionality of “label” is as follows,

if $a(i, j) = 0$(white) then skip, or
if $a(i, j) = 1$(black) then do labelling.

Details of labelling procedure are described as follows:

if $a(i-1, j) = 1$ then use the label assigned at $(i-1, j)$, else
if $a(i-1, j-1) = 1$ then use the label assigned at $(i-1, j-1)$, else
if $a(i, j-1) = 1$ then use the label assigned at $(i, j-1)$, else
if $a(i-1, j+1) = 1$ then use the label assigned at $(i-1, j+1)$, else
if no labelled neighbor found, then assign new label value to $I(i, j)$

Refer to Fig. 1 for typical problem occurring in the initial labelling described above. At the position $(2, 1)$, the algorithm has no information whether the (labelled) pixel at $(1, 3)$ is connected to it.

Once the initial labelling is completed, an image labelled, I, is produced. In order for resolving the case of multiple labels in a connected components as seen in Fig. 1, in this step 2, a transitive matrix, T, will be constructed in the manner described as the following. For pixel sweeping, in order to resolve directional partiality, take the reverse order as oppose to the order used in step 1, i.e., from bottom to top, right to left.

```
if $I(i, j) = 0$ then skip, or
else check neighborhood label to see whether different label
value is assigned
```

In case a component contains the multiple labels, for example, if two pixels with label 1 and 2, resp, are turned out to be connected each other and two pixels with label 2 and 4 are also connected in the same components, a transitive matrix T can be constructed in the manner described as below.

$$T = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}$$

Transitive closure will be then applied to the transitive matrix. In order for enhancing operational simplicity, we modified the typical transitive closure process which is demonstrated as follows,

```
for $j = h \to 0$
  for $i = w \to j$ $(*)$
    if $a(i, j) \neq 0$
      for $k = 0 \to h$
        $a(i, k)| = a(j, k)$
        $a(j, k)| = a(i, k)$
```

The step $(*)$ in above formula reduces the complexity of algorithm from $O(n^3)$ to $O(n^2 \log(n))$. Additionally, notice the reverse ordering in this second pass sweep. After transitive closure, the matrix T becomes

$$T = \begin{pmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix}$$

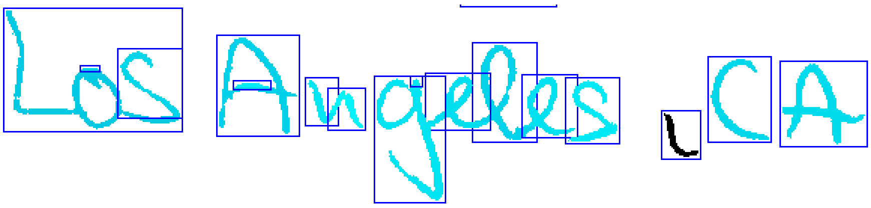


Fig. 2. Connected component algorithm run on characters in document image acquired from scanner. Bounding boxes were added for better visualization of the result.

In step 3, duplicated labels in a connect components will be merged to a unique label. For the clarity of understanding, in transitive matrix, the row indicates original label from initial labelling and the column indicates new label. As for an example, in the matrix T, $T(2,1) = 1$ implies that originally assigned label 2 will be merged to label 1. Since the step 3 is trivial, our algorithm is similar to the original labelling algorithm in [11].

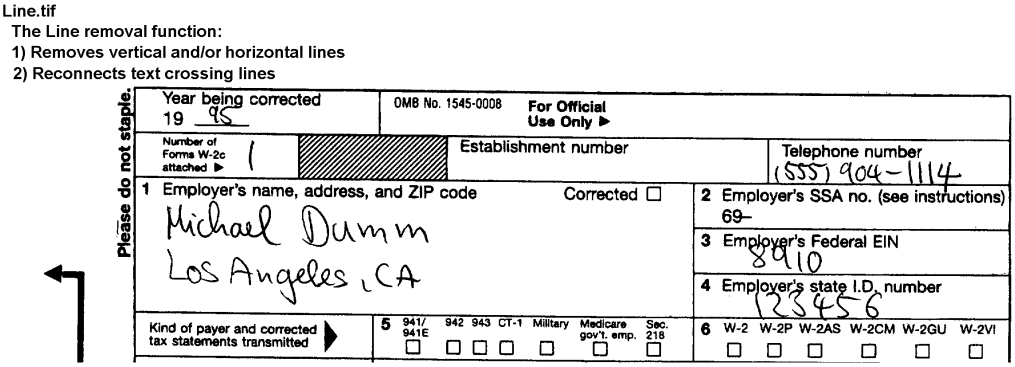
3. Results and Discussion

The algorithm was implemented using C# language of .NET framework 2003 and run on sample images. The test images were loaded into two dimensional pixel array by GDI+ in .NET.

Fig. 2 illustrates an example of connected components identified in image. For the purpose of visualizing connected components, rectangular bounding boxes for each individual connected components are displayed. From the figure, we can distinguish between the connected pixels in characters and the isolated characters.

Small bounding boxes appeared on top of char 'O' (within 'Los') in Fig. 2 are due to isolated artifacts of the character. This proves that the connected component method can also be used for denoising/despeckle, or blob removal.

For study of this paper, a binary document image having huge size of 4000 × 3000 in pixels was adopted as a test sample. For simplicity of presentation, Fig. 4 shows only part of image cropped out of the



Line.tif
The Line removal function:
1) Removes vertical and/or horizontal lines
2) Reconnects text crossing lines

Please do not staple.

Year being corrected
19 95

OMB No. 1545-0008 For Official Use Only ▶

Number of Forms W-2c attached ▶ 1

Establishment number

Telephone number
(555) 904-1114

1 Employer's name, address, and ZIP code Corrected □
Michael Dumm
Los Angeles, CA

2 Employer's SSA no. (see instructions)
69-

3 Employer's Federal EIN
8910

4 Employer's state I.D. number
123456

Kind of payer and corrected tax statements transmitted ▶

5 941/941E □ 942 □ 943 □ CT-1 □ Military □ Medicare gov't. emp. □ Sec. 218 □

6 W-2 □ W-2P □ W-2AS □ W-2CM □ W-2GU □ W-2VI □

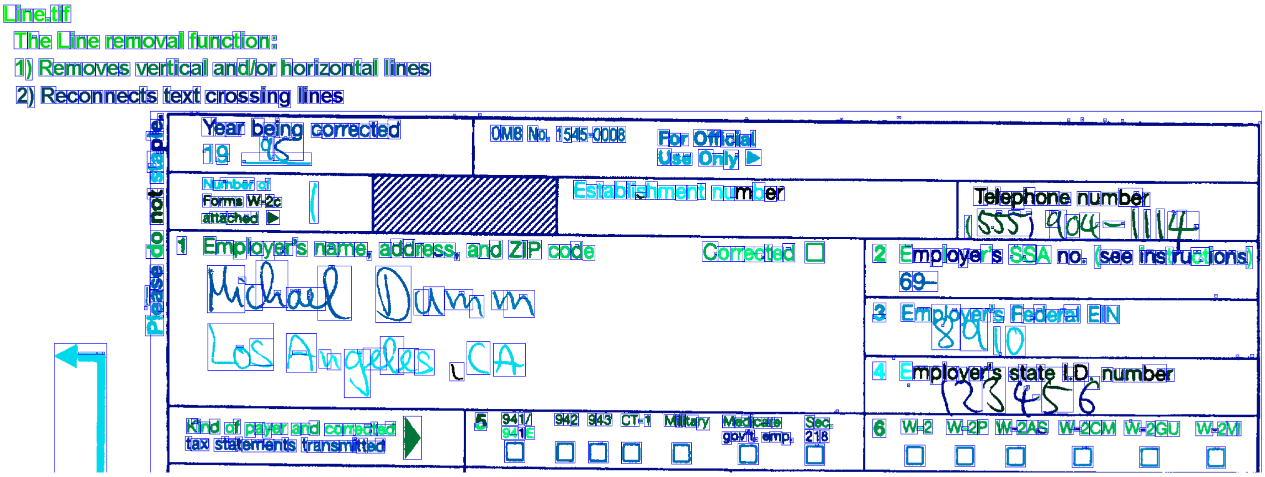
FIG. 3. Original document image sample. Image courtesy of Atalasoft, Inc.

sample input image.

The run results of connected components software is presented in Fig. 4. For the clarity of presentation, the pixels in a component have different color with the pixels in another component. The run time for this sample test took less than a second, while the other advanced technique announced that it took 2 second for image of size 1769×1168 in pixel [9].

As seen in Fig. 3, the original image contains many lines, forms and symbols as well as characters. Those non-character features thwart good performance on identification for connected components of characters in image. This implies necessity pre-processing of form, line and blob removal. In Fig. 4, unlike the original image which is black and white binary format, each connected component has different colors in order to stress distinction between different connected components.

Our study was focused on reducing process time by using linear method, transitive matrix closure. Using transitive matrix without bracketing provided efficiency in processing time. One of the authors in this paper developed an connected component algorithm in 3 dimension image by adopting Dijkstra shortest path algorithm. Dijkstra algorithm typically provides benefit of small memory usage but it consumes hundreds times of processing time due to searching operations between each pair of pixels. This algorithm will work best until an



Line.tif

The Line removal function:

1) Removes vertical and/or horizontal lines

2) Reconnects text crossing lines

Please do not staple.

Year being corrected 19 95

OMB No. 1545-0008

For Official Use Only ▶

Number of Forms W-2c attached ▶ 1

Establishment number

Telephone number (555) 904-1114

1 Employer's name, address, and ZIP code Corrected □

Michael Dumm

Los Angeles, CA

2 Employer's SSA no. (see instructions) 69-

3 Employer's Federal EIN 8910

4 Employer's state I.D. number 123456

Kind of payer and corrected tax statements transmitted ▶

5 941/941E □ 942 □ 943 □ CT-1 □ Military □ Medicare gov't. emp. □ Sec. 218 □

6 W-2 □ W-2P □ W-2AS □ W-2CM □ W-2GU □ W-2M □

FIG. 4. Black pixels components in the original binary document image are separated by the connected component algorithm.

image has 1,000 number of connected components in an image. In case of bigger image, we will have to rely on parallelized algorithm.

References

1. Alnuweiri, H., and Prasanna, V., *Fast image labelling using local operators on mesh-connected computers*, IEEE Transactions on Pattern Analysis and Machine Intelligence **PAMI-13, no. 2** (1991), 202-207.
2. ______, *Parallel architectures and algorithms for image component labelling*, IEEE Transactions on Pattern Analysis and Machine Intelligence **PAMI-14, no. 10** (1992), 1014–1034.
3. Danielsson, P. E., *An Improved Segmentation and Coding Algorithm for Binary and Non-Binary Images*, IBM Journal of Research and Development **26 (6)** (1982), 698–707.
4. Gonzalez, R.C., and Woods, R.E., *Digital Image Processing*, Prentice Hall, Upper Saddle River, NJ, 2002.
5. Greiner, J., *A Comparison of Parallel Algorithms for Connected Components*, ACM Symposium on Parallel Algorithms and Architectures (1994), 16–25.
6. Lumina, R., *A New Three-dimensional Connected Components Algorithm*, Computer Vision, Graphics, and Image Processing **23** (1983), 207–217.
7. Lumina, R., Shapiro, L., and Zuniga, O., *A New Connected Components Algorithm for Virtual Memory Computers*, Computer Vision, Graphics, and Image Processing **22** (1983), 287–300.

8. Manohar, M., Ramapriyan, H. K., *Connected Component Labelling of Binary Image on a Mesh Connected Massive Parallel Processor*, Vision, Graphics, and Image Processing **45** (1989), 133–149.
9. Park, J. M., Looney, C. G., and Chen, H-C., *Fast Connected Component Labelling Algorithm Using A Divide and Conquer Technique*, Technical Report TR-2000-04, University of Alabama (2000).
10. Ritter, G. X., and Wilson, J. N, *Handbook of Computer Vision Algorithms in Image Algebra*, CRC Press, New York, NY, 2001.
11. Rosenfeld, A., and Pfaltz, J.L., *Sequential operations in digital picture processing*, J. Assoc Comput Machinery **13** (1966), 471–494.
12. Serra, J., *Image Analysis and Mathematical Morphology*, Academic Press, New York, NY, 1982.
13. Shi, H., and Ritter, G., *Image component labelling using local operators*, Image Algebra and Morphological Image Processing IV **2030 of Proceedings of SPIE (San Diego, CA)** (1993), 304–314.
14. Yang, X. D., *An Improved Algorithm for Labeling Connected Components in a Binary Image*, Technical Report TR89-981, Cornel University (1989).

Department of Mathematics Information
Kyung Won University
Sung Nam, 461–701, Korea
E-mail: hyunkyung908@yahoo.com, jsshin@kyungwon.ac.kr