Braille Document Image Mosaicing: A Novel Approach

Srinath.S
Research Scholar
JSS Research Foundation
JSS TI's Campus, Manasagangothri, Mysore - 6

C.N. Ravikumar
Scientist
JSS Research Foundation
JSS TI's Campus, Manasagangothri, Mysore - 6

ABSTRACT

Mosaic is an area of art, where small pieces of glass are assembled in a systematic way to get a decorative image. Glasses of different colors are used in mosaic art. In Digital image processing, image mosaicing is the technique of assembling multiple pieces of images to get the combined image. This can be applied on document image or on general image. It is the art of creating big image with congregation of small images. There are many situations where it is not possible to capture large documents with the given imaging media such as digital camera, scanner or copying machines in a single stretch. This results in capture of a large document in terms of split components of a document image. Hence there is a need for mosaicing. Digital Image Mosaicing requires overlapping region to generate a single and complete image. Braille is the language used by visually impaired community for read and write communication. Conventional Mosaicing approach applied on document images fail to work on Braille documents. In this paper, we present a novel and efficient approach to mosaic two split pieces of Braille document.

General Terms

Document image processing, image mosaicing, Braille document image mosaicing

Keywords

Braille document, Braille dot column mapping, overlapping region, Braille document mosaicing.

1. INTRODUCTION

1.1 Braille Document:

Braille is the reading and writing language used by visually impaired community all over the world. Braille document contains embossed dots arranged in rows and columns. Braille character is made of embossing combination of six dots placed in 3 rows and two columns. Braille document is read by sensing the embossed dots [8-10].

Every language has its own Braille representation. Out of six dots used for a character, we can have 64 different combinations and a language which has less than 64 characters can be easily mapped on to Braille. For other languages more than one Braille box is used to represent a character [12-15].

1.2 Image mosaicing:

Two or more pieces of a larger image is combined to get the complete image is generally called as image mosaicing. Different techniques have been adopted by different researchers for image mosaicing. Image blending and Image registration are the two key components for image mosaicing. Major activity in image mosaicing is to find the geometrical relationship between the to be mosaiced images. Figure-1 shows pixel column mapping based approach used in image mosaicing.

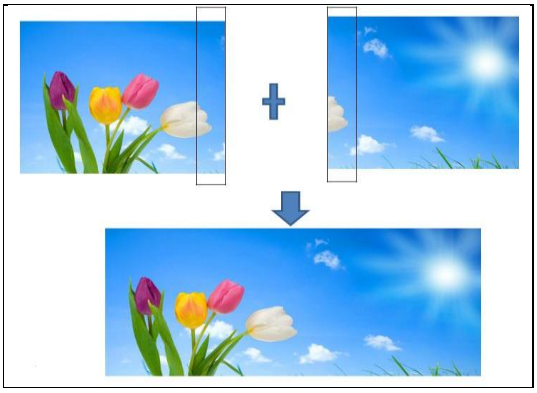


Figure-1 Image Mosaicing

1.3 Document Image mosaicing:

Image mosaicing, if applied to documents, then it is called document image mosaicing. Methods applied on natural images can also be applied on document images. Computation time can be reduced by applying techniques by looking into the nature of document. Most of the document image processing is done on black and white images and hence mosaicing such images will further reduce the time complexity.

1.4 Braille document mosaicing:

Before processing the Braille document for recognition of Characters, it is digitized. Digitization of Braille document is mainly done through Scanner. The sheet used for Braille document is usually bigger than A4 size sheet and hence it cannot be scanned in one shot using A4 size scanner. The alternate solution is to use A3 size scanner, which is expensive. Hence the document will be scanned in two stages and later they need mosaicing to get the full document for optical Braille recognition.

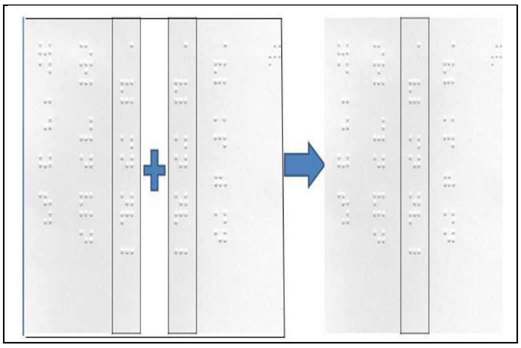


Figure-2. Braille document image Mosaicing

The techniques used for conventional document mosaicing cannot be used for Braille document mosaicing. The main reason for this is the Braille characters are aligned in columns. The gap between characters will be uniform throughout the document for all the lines. It contains uniform pixel values in between Braille dots. This may mismatch the overlapping areas. Hence the conventional mosaicing algorithms fail. There is a need for designing new algorithms for Braille document mosaicing by considering the nature of Braille document. Figure-2 shows the process of Braille document image mosaicing. Two Braille document images are used in processing and they will be having common portions. The experiment conducted in our work is through scanning the Braille document through Digital scanner. The document which is bigger than A4 size scanner is scanned in two shots.

Literature survey on the related area is presented in Section 2. Section 3 presents the proposed methodology and concluding remarks are given in section 4.

2. LITERATURE SURVEY

Ground control points (GCP) are collected in computing document mosaicing. If no GCPs are provided, but both images already have compatible geo-referencing, then an appropriate translation and scaling will be applied instead of polynomial transformation. The method can be used for remote sensed images, bio-medical images or other digital images. The paper by Indumati [1]attempts to develop a package for mosaicing multiple images. It has three modules. Each module is run independently. These modules are (1) Images are displayed in overview, full resolution/zoomed modes, (2) Registration and layout file generation, (3) Polygon filling, blending and displaying of mosaiced images. These modules are integrated to form a full-fledged system.

A novel and simple approach with columns of split images are used to identify the common or overlapping region is proposed by Hemanth kumar et.al. [2] It is applied on document images.

In the paper by Simon T.Y Suen et.al. [3]. proposed image mosaicing with optimized matching of Global and local contents with curvature domain image stitching approach given by them.

Image mosaicing with mobile phones is proposed by Jari Hannuksel et al. [4]. The basic idea in this paper is to apply online camera motion estimation to the mobile phone which assists the user in image scanning process. The scanning direction is not restricted. The camera motion was estimated during movement and the user was informed when the suitable overlap between image is taken. Mosaicing is based on robust estimation with feature point detector.

Nadege Rebiere et al. [5] addressed the image mosaicing with the help of local optical flow registration in which they proposed the use of optical flow in order to find a pixel wise registration in the overlapping region.

Ramesh babu and M.Ravishankar [6] brought the concept of quad tree technique for image mosaicing. It reduced the search space in finding the correspondence between split images.

Strip search algorithm based on novel similarity measure is proposed by Kantilal and Bhirud [7]. Hierarchical seam line estimation and vertical strip registration were introduced for precise image warping and distortion less vertical transformation.

To conclude, literature survey shows lot of work related to natural and document image mosaicing. No work on Braille document images are seen in the literature. The work presented in this paper, presents a novel approach for mosaicing Braille document images.

3. PROPOSED METHODOLOGY:

Given pieces of Braille document is pre-processed to eliminate all noise and correct the skew if any. Use this document for further processing. The main aim in the proposed methodology is find the overlapping regions.

Consider both the Braille document images individually and process row wise from top left corner pixel to bottom right corner. Identify the top and bottom edge of each Braille dot row. Repeat the process till the end of the document. Repeat the process column wise and identify the left and right edge for every dot column. Repeat the process till the end of the document.

Note the row and column numbers in the first Braille document image as r11,r12,r13…r1n and c11,c12,c13,…c1n respectively and store the pixel row and column values in the database. Similarly the second Braille document's row and column values are referenced as r21,r22,r23…r2n and c21,c22,c23,…c2n respectively. Consider the sub-image between c1n and c1 (n-1) (between last two columns, which give the last Braille dot column in the first image 'img1'). It is as shown in Figure-3.

The sub-image is formed out of co-ordinates:

(r11,c1(n-1)) and (r1n,c1n)

Find the objects within this sub-image and its pixel values. Record the objects as obj11…obj1n and their centroids of each object as (x11, y11) to (x1n, y1n). Count of number of objects is c1.

In the second Braille document image 'img2' has first Braille dot column between column c21, c22. Extract the first Braille dot column as sub-image with the co-ordinates:

(r21,c21)) and (r2n, c22)

Find the objects within this sub-image and its pixel values. Record the objects as obj21…obj2n and their centroids of each object as (x21, y21) to (x2n, y2n). Count of number of objects is c2.

If the numbers of objects in both the cases are same,

c1 equal to c2

then further comparison is done:

Get the difference between

d11= (x11-x21) and d12= (y11-y21)

d1= d11 + d12

d12 =(x12-x22) and d22 = (y12-y22)

d2=d12+d22

.

.

dn=d1n+d2n

If the 'd1' to 'dn' value is less than some threshold then the Braille dot columns are overlapping column. In our experiment threshold is fixed at10. If this condition fails, then we conclude that this is not the overlapping columns and move to the next left Braille dot column in the 'img1'. It is mapped with the first Braille dot column of 'img2'.

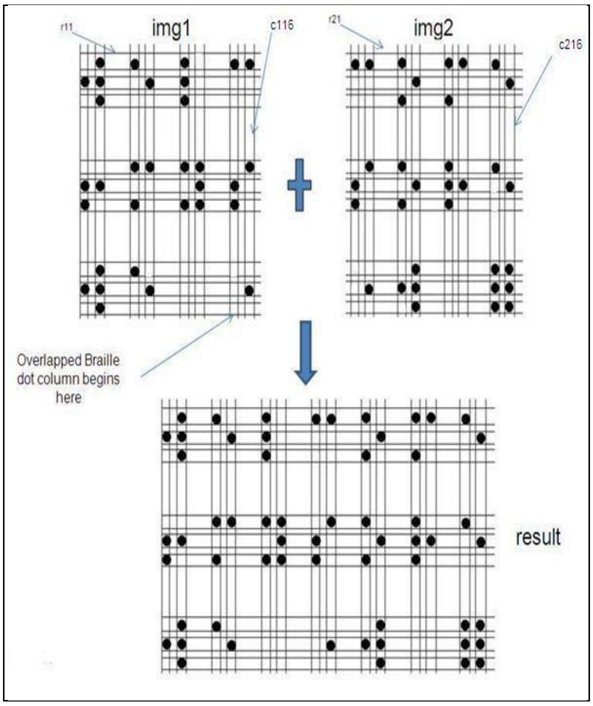


Figure-3 Braille dot column mapping for mosaicing

Repeat the process till the mapping is successful. Once the mapping column for the first Braille dot column in the second image is found in the first image, then copy contents of 'img1' and 'img2' to 'result' image as given below:

Consider 'img1' co-ordinates

(0, 0) – Top left co-ordinate

(m,n) – Bottom right co-ordinate

Consider 'img2' co-ordinates

(0,0) – Top left co-ordinate

(x, y) – Bottom right co-ordinate

Mapping Braille dot column co-ordinates in 'img1'

(0,m1)- Top left co-ordinate

(m, n1)- Bottom right co-ordinate

Copy sub-image with co-ordinates:

Copy from 'img1' to 'result'

(0, 0) - ((m1-1), n) : in img1

Append all the pixel columns of 'img2' to 'result'

(0, 0) – (x, y) :in img2

Now the resultant image 'result' is the mosaicked image of 'img1' and 'img2' eliminating the overlapping regions.

3.1 Algorithm for Braille document mosaicing:

1. Start

2.Read the two Braille document image pieces 'img1' and 'img2'.

3. Preprocess the Braille documents to eliminate the noise and extra dot which are not the part of Braille character.

4. Process step 5 and 6 on each Braille document pieces.

5. Process row wise from top left corner pixel to bottom right corner and identify the top and bottom edge of each Braille dot row. Repeat the process till the end of the document.

6. Repeat the process column wise and identify the left and right edge for every dot column. Repeat the process till the end of the document.

7. Record the row and column numbers in the first Braille document image as r11, r12, r13…r1n and c11, c12, c13,…c1n respectively and store the pixel row and column values in the database.

8. Similarly the second Braille document's row and column values are identified as

r21, r22, r23…r2n and c21, c22, c23,…c2n respectively.

9. Between the c1n and c1 (n-1), the last two columns of first image we have the last Braille dot column. Extract this as sub-image and get the number of objects 'ob1' and the co-ordinate values of centroid's of each of this object. The centroid co-ordinates are recorded as (x11, y11), (x12, y12),…(x1n, y1n). Where each dot, is a true Braille dot of Braille character.

10. Step 9 is repeated on the second Braille document image 'img2' and extract the first Braille dot column and get the count of objects 'ob2' in this sub-image and the co-ordinate values as (x21, y21), (x22, y22),…(x2n, y2n).

11. If ob1 and ob2 are same then go to step 12 else step 14.

12. Get the Euclidean distance 'd' between set of co-ordinate values of objects of 'img1' and 'img2'.

13. If the distance is close to zero then these two columns are the overlapping columns, go to step 14.

Else go to step 15.

14. Mosasicked image 'result' is constructed by first copying pixels from first column till one pixel column before the mapping column of the first Braille document image 'img1' to 'result'. Now append all the pixel column of 'img2' to 'result' image. Go to step 18.

15. Move from right to left in extracting next Braille dot column in 'img1'.

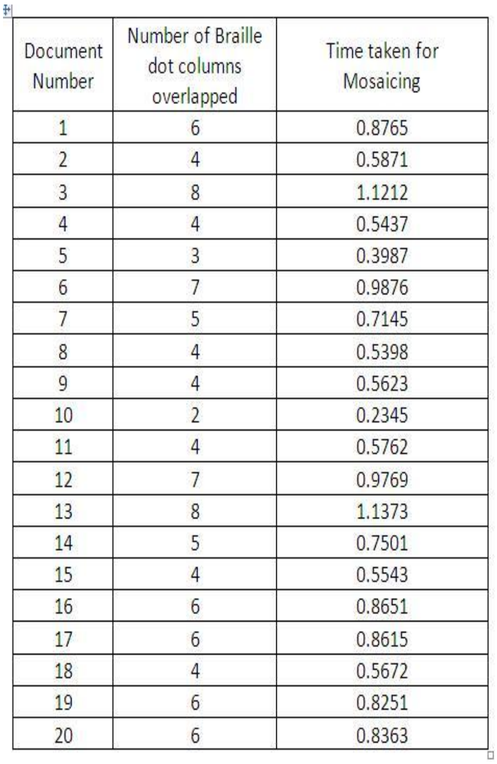
16. If the end of Braille dot columns is reached then no overlapping region found and mosaicing is not possible, go to step 18 else go to step 17.

17. Get the count of objects 'ob1' and the object's co-ordinates from 'img1'. The count of objects 'ob2' and object's co-ordinates of 'img2' are retained and go to step 11.

18. Stop.

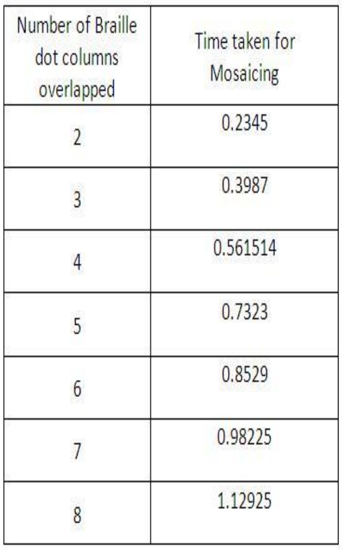
3.2 Results and Discussion

Performance of this algorithm is given in Table-1. The performnace is linear with the number of Braille dot columns overlapped. Table-2 gives the average performance in terms of number of Braille dot columns overlapped. Figure-4.6 shows the chart for the average performance by considering the number of Braille dot columns overlapped against the time taken for mosaicing.



Document Number	Number of Braille dot columns overlapped	Time taken for Mosaicing
1	6	0.8765
2	4	0.5871
3	8	1.1212
4	4	0.5437
5	3	0.3987
6	7	0.9876
7	5	0.7145
8	4	0.5398
9	4	0.5623
10	2	0.2345
11	4	0.5762
12	7	0.9769
13	8	1.1373
14	5	0.7501
15	4	0.5543
16	6	0.8651
17	6	0.8615
18	4	0.5672
19	6	0.8251
20	6	0.8363

Table-1
Performance of Braille Mosaicing



Number of Braille dot columns overlapped	Time taken for Mosaicing
2	0.2345
3	0.3987
4	0.561514
5	0.7323
6	0.8529
7	0.98225
8	1.12925

Table-2
Average Performance against Number of Overlapped Braille Dot Columns

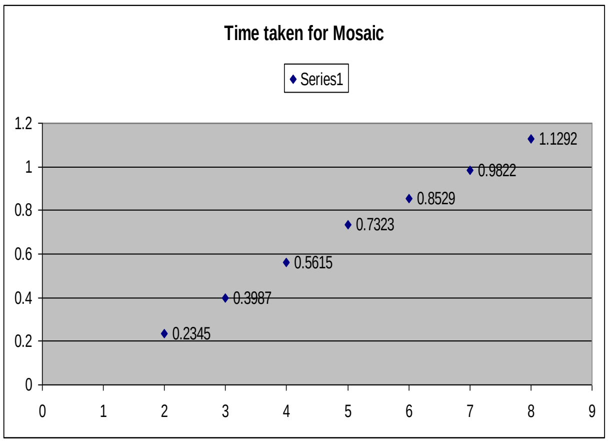


Figure- 4 Number of Overlapped Columns vs. Time Taken

4. CONCLUSION

The proposed algorithm presented here has successfully and efficiently mosaiced the Braille documents. Braille dot positions are used in mosaicing. Document used in the experiment are free from noise and are skew corrected. To the best of our knowledge, it is the first and important mile stone in Braille document mosaicing. Only two pieces of Braille documents are used for mosaicing. Algorithm can be extended by considering more than two pieces for mosaicing. Algorithm should also verify whether the Braille documents are aligned or they are placed upside down. Design of algorithm for mosaicing inter-point Braille document will be more challenging. Robust algorithm for mosaicing Braille documents having no overlapping regions is challenging.

5. ACKNOWLEDGMENTS

Our sincere thanks to Visvesvaraya Technological University for funding the Equipments required for this work under VTU Research grant Scheme.

6. REFERENCES

[1] Indumathi, “Image mosaicing”, Proceddings of Geosciences and remote sensing symposium, pp2363-2365, 1998

[2] G.Hemanthkumar, P.Shivakumar,D.S.Guru and P.Nagabhushan, “Document image mosaicing: A novel approach”, Sadhana, vol 29, part 3, pp 329-341, 2004.

[3] Simon T.Y, Suen, Edmund Y.Lam, Kenneth K.Y Wong, “Digital Photograph Stitching with Optimized Matching of Gradient and Curvature”, Proceedings of the SPIE, VMLXIX, pp139-154, 2006

[4] Jari Hannuksela, Pekka Sangi, Janne Heikkila, Xu Liu, David Doermann, “ Document Image Mosaicing with Mobile phones”, Proceedings of International conference on Image Analysis and Processing, pp575-582, 2007

[5] Nadege Rebiere, Marie Flavie Auclair-Fortier, Francois Deschenes, “Image Mosaicing using local optical flow registration, Proceedings of International conference on Pattern recognition pp1-5, 2008.

[6] D.R. Ramesh Babu, M. Ravishankar, “Automatic Seamless Image Mosaicing” An approach Based on Quad Tree Technique”, Proceedings of the world congress on Engineering, pp687-691, 2010

[7] Kantilal P.Rane and S.G. Bhirud, “Image mosaicing with strip search algorithm based on a novel similarity measure”, International Journal of Computer Application, XXVII(2), pp42-47, 2011.

[8] Srinath.S and C.N. Ravikumar, “An Insight into Optical Braille Character Recognition since its Conceptualization”, International Journal of Computer Applications, Volume 33-No.6 November 2011

[9] Aisha Mousa, hazem Hiary, Raja Alomari, and Loai Alnemer, “Smart Braille System Recognizer”, IJCSI International Journal of Computer Science issues vol10, Issue 6, No.1 November 2013

[10] Srinath.S and C.N. Ravikumar, “A novel and efficient algorithm to recognize any universally accepted Braille characters: a case with Kannada language”, IEEE proceedings of the international conferece on Signal and Image processing, Bangalore, pp292-296, 2014

[11] R.C. Gonzalez and R.E. Woods, "Digital Image Processing". Prentice Hall

[12] Different equipments for Visually Impaired: http://www.karishmaenterprises.com/

[13] Different equipments for Visually Impaired: http://www.sparshproducts.com/

[14] National Association for Blind at: http://www.nabindia.org/

[15] World Health organization at: http://www.who.int