An Image Segmentation using Normalised Cuts in Multistage Approach

Mei Yeen Choong, Wei Yeang Kow, Wei Leong Khong, Chung Fan Liau, Kenneth Tze Kin Teo
Modelling, Simulation & Computing Laboratory, Material & Mineral Research Unit
School of Engineering and Information Technology
Universiti Malaysia Sabah
Kota Kinabalu, Malaysia
msclab@ums.edu.my, ktkteo@ieee.org

Abstract — **Normalised cuts algorithm is complex for image segmentation but it produces good segmentation result. At present, digital camera can produce high detail image. To avoid underutilising the high detail image, resizing image into smaller resolution is discouraged. This creates a constraint in resizing image to smaller resolution while preserving the important detail in the image. An image segmentation method using normalised cuts done in two-stage manner is proposed here to solve the issue of the image resolution is excessively reduced prior to image segmentation. In this work, an image is first separated into several regions (named as image cells). The locally produced segments from each of the image cells are then undergone for second stage segmentation to look for possibility of merging them up. This paper includes the experimental results using the mentioned approach and the experiment shows that it is capable to produce meaningful segments.**

Keywords - image segmentation, normalised cuts, clustering

I. INTRODUCTION

Image segmentation has been widely applied in various fields. Many of the computer vision applications require image segmentation for information extraction. An example of information extraction in medical computer vision is detecting abnormities in certain body organ. Often, the goal of image segmentation is to decompose an image into meaningful segments that correlates with the real world and these meaningful segments are used for further important analysis. Extensive research has been done to create and develop various methods to produce segmentation results that closely related to human's visual perceptions. Feature detection based method such as Canny edge detector [1], that detects changes of image brightness for particular sets of pixels. This edge detection method has certain drawbacks. It cannot guarantee in generating closed edge contours of segments [2]. Thus, extraction for image segmentation should be performed in global view to solve the limitation. A method based on computer science's graph theory is introduced here. It overcomes the limitation by doing image segmentation using graph partitioning method. Wu and Leahy pointed out that this approach guarantees the formation of closed edge contour [3, 4]. Clustering algorithm is then applied to construct meaningful segments by clustering the eigenvector generated from the Laplacian matrix. K-means clustering is one of the simple clustering algorithm can be used by providing the appropriate number of clusters for it to construct.

The details of the segmentation process based on the proposed approach in this paper are organized as follows. In section II, implementation of the image segmentation in normalized cuts and image resizing are described. Explanation of the segmentation based on proposed approach is presented in section III. In section IV, the experimental results and discussions are provided. Lastly, this paper is concluded in section V.

II. IMAGE SEGMENTATION

It is not always that image segmentation result can match to human perspective especially when the segmentation method is based on similarity using primitive information, such as image brightness [5]. Thus, the aim of the segmentation should be focused on segmenting objects with closed boundaries so that the image segmentation results can be meaningful and easier to analyse.

A. Image Segmentation in Graph Partitioning

In this paper, the image segmentation method using the graph partitioning method basically viewing image as a graph model. This graph modelling eventually leads to grouping similar pixels into homogeneous segments.

A set of image pixels can be represented as weighted graph **G** = (**V**, **E**) where **V** represents the vertices which are image pixels (a vertex is made up one node) and **E** represents the edges in the form of weights, w. Each of the w gives a measurement of the similarity between node i and node j. A graph is bi-partitioned into two distinct sub-graphs A and B with the condition that it minimizes the value of

$$cut(A,B) = \sum_{i\in A,\, j\in B} w(i,j)\ , \tag{1}$$

with constraint $A \cup B = \mathbf{V}$ and $A \cap B = \varnothing$ [6, 7]. Since each of these pixels holds colour information, grouping these pixels according their similarities and dissimilarities can lead to achieving the image segmentation goal. The degree of the dissimilarity between two sub-graphs A and sub-graphs B, which is the sum of weights of the pairs of nodes are to be removed is shown in (1) [5]. The bi-partitioning process recursively finds the minimum cuts until a number of k sub-graphs are formed, with the condition that the maximum possible cut across the sub-graphs is minimized. In other words, the formed distinct sub-graphs have high similarity within the sub-graphs and low similarity across different sub-graphs [7]. However, the minimum cut criterion falls short that its cut algorithm favors in cutting isolated nodes to form sub-graphs. This lead to proposing an improved cut algorithm based on normalised cut (*Ncut*) criterion to alleviate the problem [5]. Fig. 1 illustrates *Ncut* and the ordinary minimum cut. The normalised cut criterion is derived as in

$$\begin{aligned} & Ncut(A,B) \\ & = \frac{cut(A,B)}{assoc(A,V)} - \frac{cut(A,B)}{assoc(B,V)} \\ & = \frac{assoc(A,V) - assoc(A,A)}{assoc(A,V)} \\ & + \frac{assoc(B,V) - assoc(B,B)}{assoc(B,V)} \\ & = 2 - \left[\frac{assoc(A,A)}{assoc(A,V)} + \frac{assoc(B,B)}{assoc(B,V)} \right] \\ & = 2 - Nassoc(A,B) \end{aligned} , \qquad (2)$$

whereby $assoc(A,V) = \sum_{i \in A, j \in V} w(i,j)$. Based on the (2), Shi and Malik solved the isolated notes cutting issue by introducing a disassociation measure for the cut. The measure eliminates the occurrence of partition that cuts out isolated small set of pixels by formulating the cut as a fraction of the total edges that paired with all the pixels in the graph. The relation between disassociation and association can be described as in (2). Minimising the dissociation between partitions denotes maximising the association between partitions.

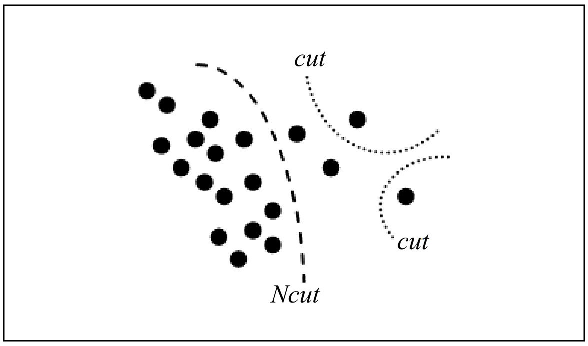


Figure 1. The *Ncut* and the ordinary cut.

The $Ncut(A,B)$ is then transformed to

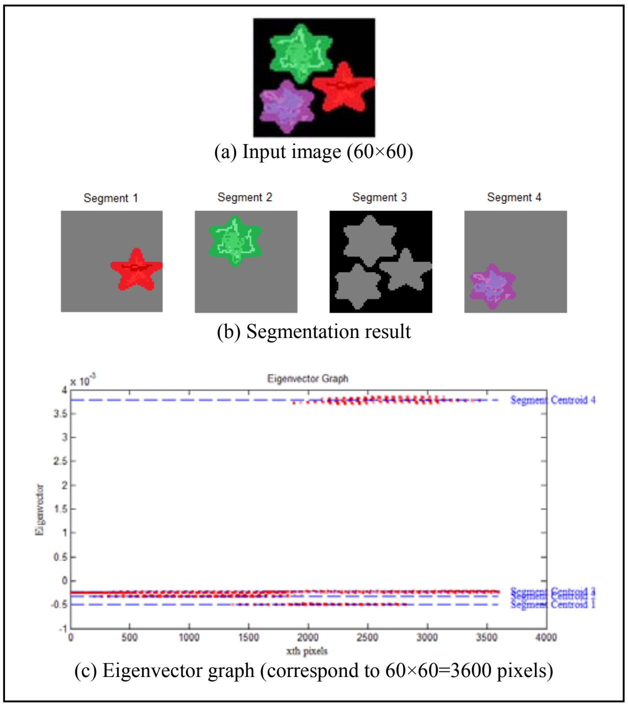
$$(\mathbf{D} - \mathbf{W})y = \lambda \mathbf{D} y , \qquad (3)$$

for solving the eigenvectors y and eigenvalues λ. $\mathbf{D} - \mathbf{W}$ is called the Laplacian matrix whereby $\mathbf{W}$ is a symmetrical matrix that contains element $w(i, j)$. Each of the weight elements in matrix $\mathbf{W}$ is defined as

$$w(i,j) = e^{\frac{-\|\mathbf{F}(i)-\mathbf{F}(j)\|_2^2}{\sigma_I}} * \begin{cases} e^{\frac{-\|\mathbf{X}(i)-\mathbf{X}(j)\|_2^2}{\sigma_X}} , & \text{if } -\|\mathbf{X}(i)-\mathbf{X}(j)\|_2^2 < r \\ 0 , & \text{otherwise} \end{cases} \qquad (4)$$

where $\mathbf{F}(i)$ is feature vector based on the intensity value in colour of node i and $\mathbf{X}(i)$ is the spatial location of the node. When a pair of node i and node j is more than r number of pixels apart the weight $w(i, j)$ is considered 0. $\mathbf{D}$ is a diagonal matrix with $d(i)=\sum_{j=1} w(i, j)$ on its diagonal [5, 6, 7].

Image segmentation result is then produced by clustering on the eigenvector. Each of the element in the computed eigenvector contains description value corresponds to each of the pixel in terms of spatial and brightness (pixel intensity) measurement in the image. Ideally, eigenvector with the smallest nonzero eigenvalue is chosen to perform clustering on it. In this paper, eigenvector corresponds to the second smallest eigenvalue is chosen for k-means clustering algorithm to do the clustering on the eigenvector. After the clustering process on the eigenvector, segmentation result is finally obtained based on the clustered eigenvector. This clustered eigenvector is then mapped according to pixels allocation in the image. Fig. 2 illustrates the segmentation result and its corresponding eigenvector graph labelled with final clusters' centroids in it.



(a) Input image (60×60)

(b) Segmentation result

(c) Eigenvector graph (correspond to 60×60=3600 pixels)

Figure 2. The image segmentation result and the eigenvector graph.

DOI 10.5013/IJSSST.a.13.3B.02

ISSN: 1473-804x online, 1473-8031 print

B. Image Resizing

Image resizing is normally conducted before performing image segmentation. There is a significant issue that lower resolution images suffered information lost as fine details of the image turned out to be coarse. Thus, producing a good segmentation result of an image shall not be aimed when the image is resized to the extent that important features in the image are missed out [8]. For example in the application of locating vehicle license plate, image resizing affects the characters outline in the plate region such that the characters are hardly to be seen when they are coarsened [9]. This particular region will fail the pattern recognition miserably since the characters are not recognisable. However, performing image segmentation on high resolution image is not feasible for online systems. Fig. 3 demonstrates how an image appears at different resolution. Notice that the head of the bird is not visible in the image with resolution of 18×24. In the image with 90×120, it is then only known that the bird is being kept in a cage.

III. Image Cells and Image Segmentation

Normalised cuts implementation in image segmentation has a drawback in its computation. An image that has a size of $m \times n$ pixels will require the normalised cuts algorithm to form the **W** matrix with a size of $(m \times n) \times (m \times n)$. For example, an image with a size of 120×90 will end up computing a 10800×10800 adjacency matrix, which is considered a very large matrix for a computer to solve for eigenvalues. This indicates the size of **W** is proportionally to the square of the image size. With the current digital camera that can produce an image with image size more than 10 mega pixels, doing image segmentation using normalised cuts algorithm in this image size would be impractical. An image segmentation done in hierarchical manner is proposed here to counter the issue.

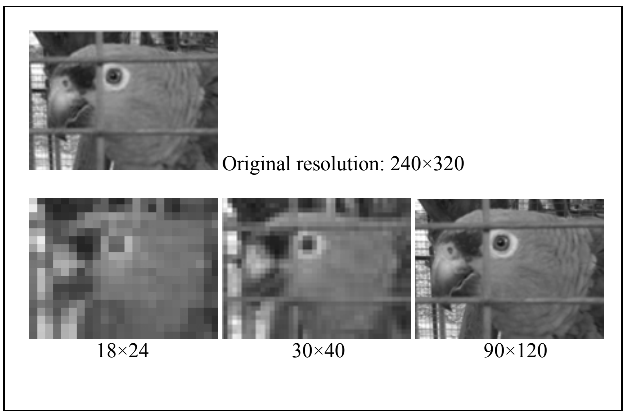


Figure 3. Image at different resolution.

A. Image Cells and First Stage Segmentation

To avoid processing such large matrix, it is suggested that a high resolution image be divided into equal size of box-shaped image cells [11]. Since natural scene image has irregular pixels distribution such that the frequency of image pixels for particular range occurred unevenly throughout the whole area of the image, a preliminary check is run on every image cell to determine whether the particular cell requires normalised cuts algorithm to perform on it. The first stage of segmentation is begun by providing the k_1 number of segments, normalized cuts algorithm is then performed to segment out k_1 number of clusters for the particular cell. In this segmentation stage, over-segmentation would likely to be occurred as discrimination power in an image cell is reduced compared with the discrimination power on the whole image. Nevertheless, it helps to reduce the tendency of important object boundaries to be missed out. In every image cell, the implementation of simple k-means clustering plays an important part on clustering out the generated eigenvector is based on the given number of segments, k_1.

Fig. 4 shows an image is divided into designated number of cells and local segmentation on each of the cells. The segmentation on each of the cells is done independently. In other words, the segmentation done in a cell is not co-related with the segmentation done in other cell. Segmented clusters from the image cells are then used for second stage segmentation.

B. Segments Merging and Second Stage Segmentation

To proceed to the second stage segmentation, simple representations of the segments are computed. Each produced segment has a node to represent each of them by calculating the median value of the segment and centroid location of the segment. The computed nodes are acted as pixels and used for second stage segmentation using normalised cuts algorithm. The value of the threshold, r, which is the maximum distance for a pair of pixels can be apart, is increased because the computed nodes are scattered in sparse manner.

The number of nodes is corresponded with the total number of segments from the first stage segmentation. For example, 18 produced segments from the first stage segmentation will have 18 nodes to be computed. Normalised cuts algorithm is performed on second time based on these computed nodes. Segments will be merged together when their nodes are grouped together and share common similarity based on color and spatial location information, while segments which are distinct to each other are retained as separate segments. Table I presents pseudocode of normalised cuts with the implementation of image cells division. Fig. 5 illustrates the second stage segmentation produces the final segmentation result (based on input image in Fig. 4).

DOI 10.5013/IJSSST.a.13.3B.02

ISSN: 1473-804x online, 1473-8031 print

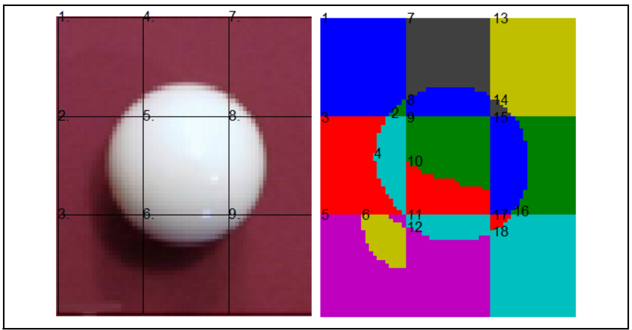


Figure 4. 3 × 3 image cells and its segmentation on the image cells individually.

TABLE I. PSEUDOCODE OF NORMALISED CUTS IN TWO STAGES

	Input: Colour/ grayscale image
1	**Read** input image
2	**Construct** $M \times N = n$ image cells
3	**Perform** statistical analysis of the image cells

4	**Determine** whether i^{th} *image* cell required segmentation
	    if > threshold value
	        proceed to local normalised cuts segmentation
	    else
	        compute background node in i^{th} image cell

5	**First stage segmentation** Normalised cuts local segmentation into k_1 segments on i^{th} image cell until n^{th} image cell.
6	**Nodes computation** Each of the segments from every image cell is represented by a node.
7	**Second stage segmentation** Perform normalised cuts segmentation based on the computed nodes to k_2 segments.
	Output: Produce segmented segments and display result.

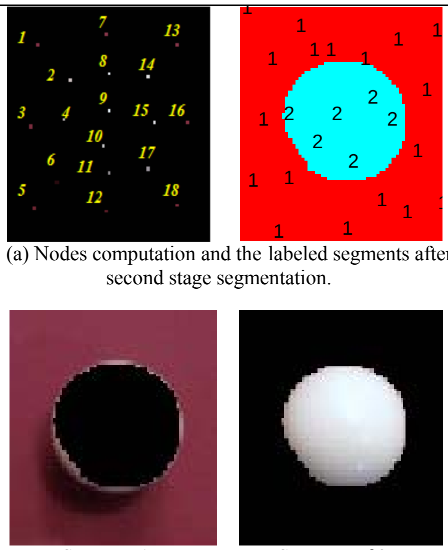
IV. EXPERIMENTAL RESULTS AND DISCUSSIONS

To illustrate the implementation of normalised cuts in image segmentation, segmentation on synthetic and natural images is done. Some of the natural images used for the segmentation are taken from the Berkeley Segmentation Dataset (BSDS500) [10].

A. Synthetic Image Segmentation

The synthetic image is self generated and designed to demonstrate the capability of the normalised cuts in producing a desired segmentation result. Due to computation expensive for the normalised cuts, the image sizes used are 30 × 40 and 60 × 80. Image segmentation with normalised cuts in two-stage manner is then performed on image with size of 75 × 100.

Based on Fig. 6, the segmentation in 30 × 40 image gives a computation time of 3.6 seconds, while 60 × 80 image gives a computation time of 118.1 seconds which is almost 2 minutes. Although the difference of these two image sizes is not large, their computation time can differ greatly, which is about 114.5 seconds difference. It can be said that using normalised cuts algorithm for image segmentation on image



(a) Nodes computation and the labeled segments after second stage segmentation.

Segment 1 Segment of 2

(b) Result of the two-stage image segmentation.

Figure 5. Second stage segmentation and its result.

with large size may severely hamper the segmentation efficiency. This shows that large image segmentation using normalised cuts algorithm may not meet the requirement of certain computer vision application in reality.

To retain the use of normalised cuts algorithm but done in an optimal way, two-stage segmentation approach is applied here for larger image. The same image with a larger size of 75 × 100 is processed by using the two-stage approach segmentation which is also shown in Fig. 6. Even though the result of the image segmentation done in two-stage approach may not be as good as the previous two images done with the ordinary approach, it is still able to segment out the important objects although it is lacking of some extend of accuracy in the segmentation.

B. Natural Image Segmentation

Because of the natural images are large in size, image segmentation using normalised cuts algorithm is performed in two-stage manner. A simple natural image with the size of 300 × 400 shown in Fig. 7 contains the flowers as foreground and the area surrounding the flowers as the background is selected for the segmentation. It can be seen that the flower petal can be segmented out from the background. The boundaries of the segment follow nicely with the edge of the flower petal in the image. The segmentation for the image whereby the image is divided into 6 × 6 image cells takes about 6 minutes which is considered long in computation time.

In the attempt to reduce the computation time, image segmentation on image with 8×8 image cells is performed as shown in Fig. 8. Eventually, the computation takes about 3 minutes which is less than the previous segmentation about half of the time. This indicates increasing number of cells helps to reduce the computation time as the area of an image

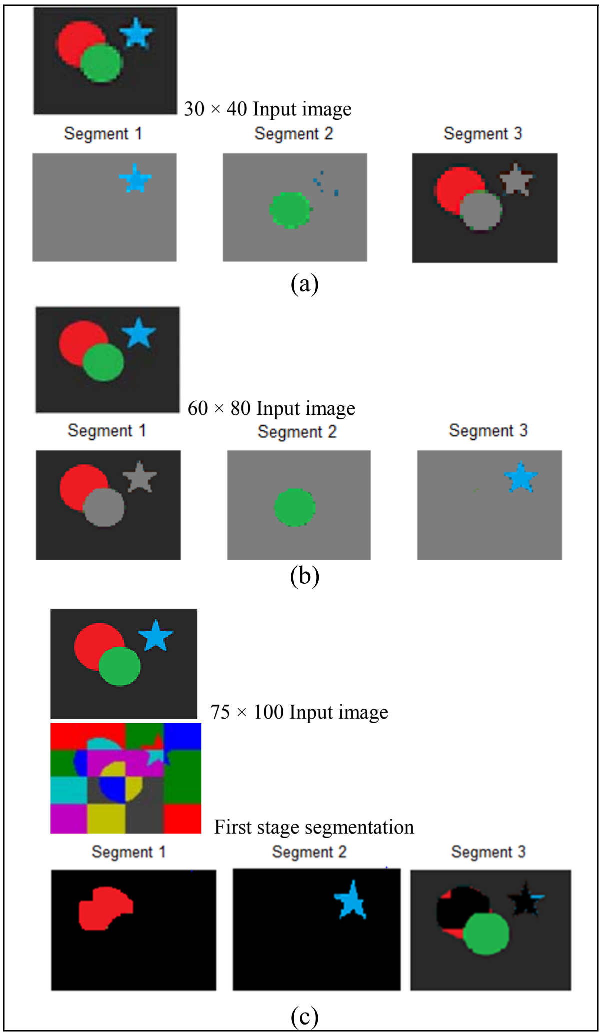


Figure 6. Image segmentation on synthetic image: (a) 30 × 40 image, (b) 60 × 80 image, (c) 75 × 100 image (two-stage segmentation).

cell has become smaller despite more cells is added for first stage segmentation. Small size of image cell to some extend can help to reduce the occurrence of important boundaries to be missed out during segmentation process. However, small image cell is not suitable for image with large objects in it. This leads to the segmentation on the image cells is performed unnecessary. Fig. 8 shows the segmentation result when the number of cells is increased.

Often image captured in real world contains objects that are constructed with tiny lines. An image from the Berkeley Segmentation Dataset (BSDS500) [10] contains the steel roller coaster tracks is tested for the segmentation and its result is shown in Fig. 9. As it can be seen that for the 6 × 6 image cells segmentation, the part marked with red circle ring on the top left of the clustered segment 2 has a sharp right angle edge. This is due to normalised cuts algorithm is not run on the corresponded cell. However, for the 5 × 5 image cells segmentation, the sharp corner edge does not occur. The edges of the segment follow closely with the object boundaries. When the number of image cells increased on the same image, the area covered by each of the image cells has a zoom in effect on it. This decreases the discrimination power for the area. In the first cell (top left) of 5 × 5 image cells, there is a critical edge boundary that is supposed to be identified. However, the edge does not take up much of the area in the cell. The normalised cuts algorithm omits the cell that required segmentation in it and eventually ends up having the final produced segments appeared to be blockish. This happens to other image cells when this similar situation occurred. Notice that the roller coaster track is not segmented out together with blue background. This shows the normalised cuts algorithm is

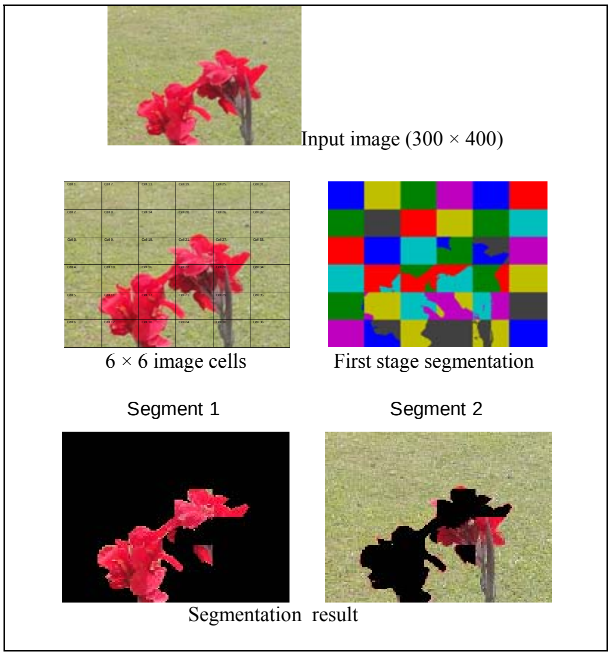


Figure 7. 6 × 6 image cells and segmentation result.

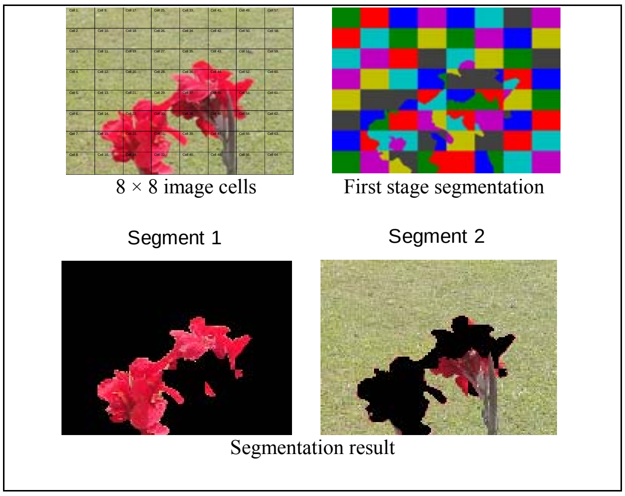


Figure 8. 8 × 8 image cells and segmentation result.

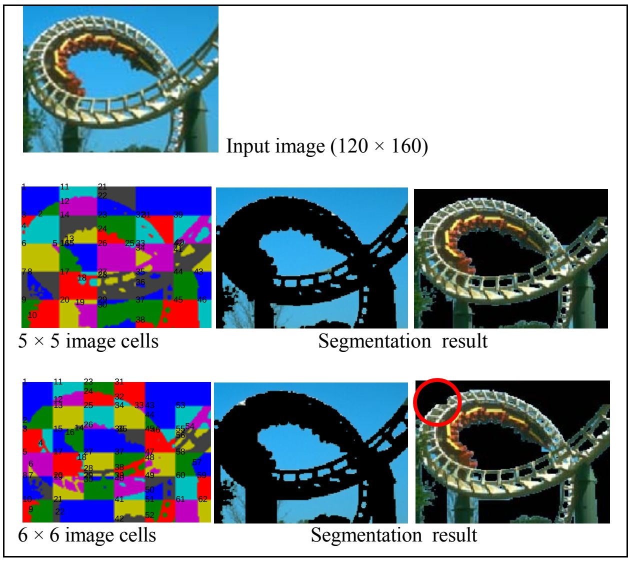


Figure 9. Comparison between 5 × 5 and 6 × 6 image cells and their segmentation results.

able to tackle fine details of the objects in the image and also across the image cells individually. Fine tuning the number of clustered segments per cell gives an indicator such that to what extent that the segmentation has to be accurately performed. Increasing the number of segments per cell does not significantly increase the computation time if the number of cells to be divided for the image is constant. Fig. 10 shows the effect of changing the number of clustered segments per cell on the segmentation process while retaining the number of image cells in the image. As the number of segments per cell increased, more detailed contents of the cell are segmented out. For the segmentation done with 10 clustered segments per cell, the letter at the tail section of the aircraft is segmented out. However, the number of clustered segments per cell is increased to the extent where some of the edges in the image such as the shading and shadow of the same objects are also segmented out and considers them as distinct segments. Thus, selecting an optimal number of clustered segments per cell should be based on the image complexity to reduce the tendency of the image being segmented undesirably. Otherwise, customising number of image cells for a certain region of the image can be assigned when some objects in particular region of the image need more detailed segmentation on it.

In general, the simplest task for the image segmentation can go by just segmenting out objects that are acted as foreground from the background in an image. However, when the background of an image contains multiple colour components, it gives challenges to segment out the objects from the background. Especially the object's edges seemingly appear to be diffused with background or considered as part of the background components. Unclear or blurry boundaries even give more challenges for primitive image segmentation method to segment the image into desired segments. Fig. 11 shows an image with vast variety

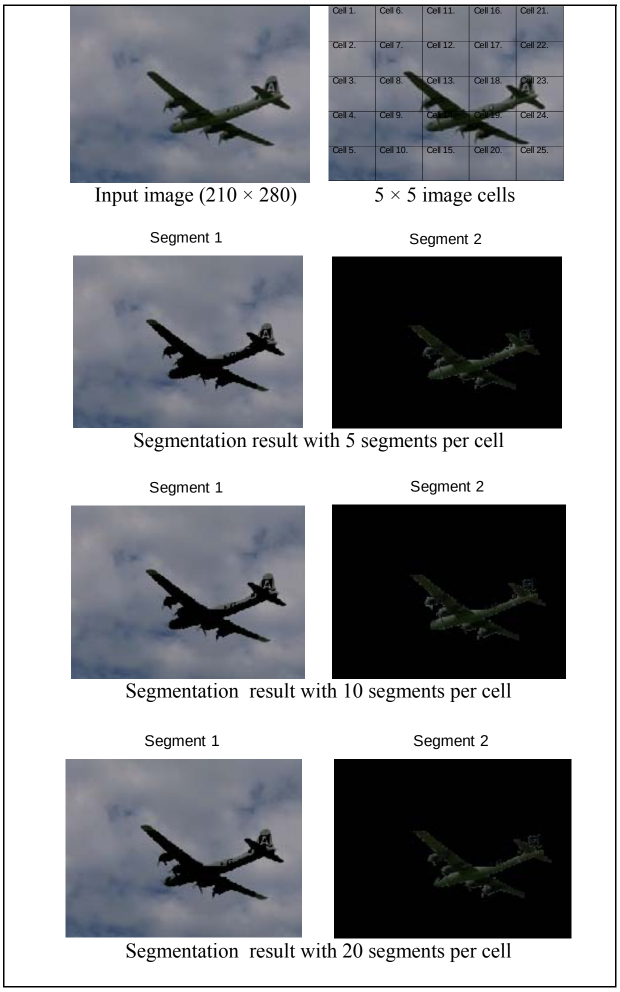


Figure 10. Comparison of image cells segmentations with different number of clusters per cell.

of colour components being segmented out. Although the Fig. 11 shows that the segmentation does not produce desirable results, but in the least the running humans in the image are segmented out from the background which appears to be the crowd and grass field. The two segmented humans (foreground objects) have fewer colour components as compared to the background that have more colour components because of the humans' bodies are largely covered by the skin colour. This helps the normalised cuts algorithm to segment out the humans without much effort. The image cell size covering the human body in the image is adequate small enough to perform the first stage segmentation since the discrimination degree in each of the cells is sufficient.

V. CONCLUSIONS

An alternative approach for normalised cuts algorithm to perform the image segmentation in a more efficient way has been carried out. With this approach, it enables normalised cuts algorithms to perform segmentation on image part by part individually instead of performing segmentation on

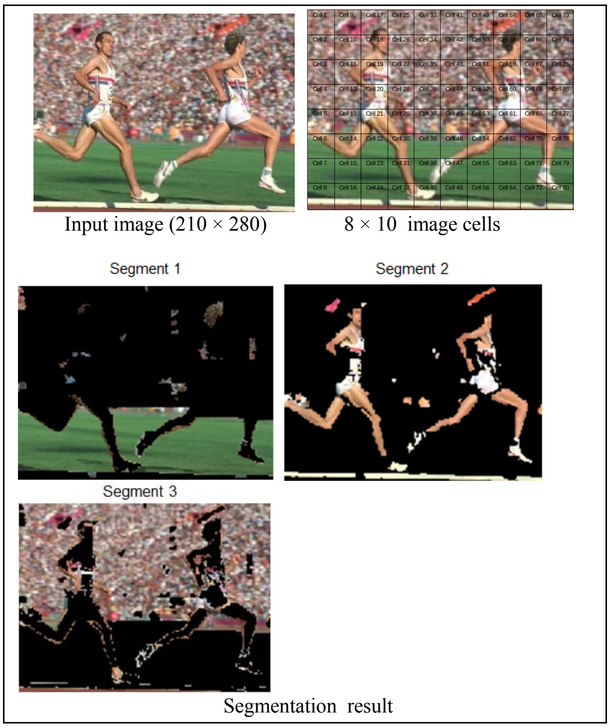


Figure 11. 8×10 image cells segmentation and its result.

whole image in one stage. This helps to speed up the normalised cuts algorithm.

Further improvement can be done by balancing the trade-off between computation efficiency and effectiveness such that the segmentation can meet the optimal performance. Another improvement task is that the number of image cells to be formed can be adaptively defined according to the image content. When foreground area takes larger portion than the background area, the number of image cells should be cautiously not too many to reduce the tendency of waste in computation for the local segmentation in every image cell. Irregular sizes of the image cells can be formed in an image since the frequency content of the image is not evenly distributed. For example, region with fine details suggested to be covered with smaller image cells, while background that contains fewer details is suggested to be covered with bigger image cells or less number of image cells. Nevertheless, the tuning of the image cell size still needs to meet optimal trade-off without affecting either the efficiency or accuracy significantly.

A higher level knowledge can be implemented to assist the segmentation using normalised cuts algorithm. However, there exists a potential where the use of different primitive domains to do the segmentation in the absence of high level knowledge can be done [12].

ACKNOWLEDGMENT

The authors would like to acknowledge the funding assistance of Universiti Malaysia Sabah (UMS) under UMS Research Grant Scheme (SGPUMS), no. SBK0026-TK-1/2012, and Postgraduate Centre of UMS under Postgraduate Assistance Scheme.

REFERENCES

[1] J. Canny, “A Computational Approach to Edge Detection,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 8, issue 6, 1986, pp. 679–698.

[2] S. C. Zhu, and A. Yuille, “Region Competition: Unifying Snakes, Region Growing, and Bayes/MDL for Multiband Image Segmentation,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 18, Sep. 1996, pp. 884-900, doi:10.1109/34.537343.

[3] Z. Wu and R. Leahy, “An optimal graph theoretic approach to data clustering: theory and its application to image segmentation,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 15, Nov 1993, pp. 1101 -1113, doi: 10.1109/34.244673.

[4] S. Maji, N. Vishnoi, and J. Malik, “Biased normalized cuts,” *2011* IEEE Conference onComputer Vision and Pattern Recognition (CVPR), June 2011 ,pp. 2057 -2064.

[5] J. Shi and J. Malik, “Normalized cuts and image segmentation,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, Aug 2000, pp. 888 -905, doi: 10.1109/34.86868.

[6] A. M. Martínez, P. Mittrapiyanuruk, and A. C. Kak, “On combining graph-partitioning with non-parametric clustering for image segmentation,” Computer Vision and Image Understanding, vol. 95, July 2004, pp. 72-85, doi: 10.1016/j.cviu.2004.01.003.

[7] L. G. Shapiro and G. C. Stockman. *Computer Vision*. Prentice Hall, 2001.

[8] M.Y. Choong, W.L. Khong, W.Y. Kow, L. Angeline, and K.T.K. Teo, “Graph-based Image Segmentation using K-Means Clustering and Normalised Cuts,” 4th International Conference on Computational Intelligence, Communication Systems and Networks (CICSyN), 24-26 July 2012, pp. 307-312.

[9] L. Angeline, W.L. Khong, F. Wong, I. Saad, and K.T.K. Teo, “Multiple Vehicles’ License Plate Tracking and Recognition via Isotropic Dilation,” 3rd International Conference on Computational Intelligence, Communication Systems and Networks (CICSyN), 26-28 July 2011, pp. 54-59, doi: 10.1109/CICSyN.2011.24.

[10] P. Arbeláez, M. Maire, C. Fowlkes, and J. Malik, “Contour Detection and Hierarchical Image Segmentation,” IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), vol. 33, no. 5, May 2011, pp. 898-916, doi: 10.1109/TPAMI.2010.161.

[11] M.Y. Choong, C.F. Liau, J. Mountstephens, M.S. Arifianto, and K.T.K. Teo, “Multistage Image Clustering and Segmentation with Normalised Cuts,” 3rd International Conference on Intelligent Systems, Modelling and Simulation (ISMS), February 2012, pp. 362-367, doi:10.1109/ISMS.2012.112.

[12] M. Petrou and C. Petrou. *Image Processing: The Fundamentals*. John Wiley & Sons, 2010.