International Journal of Engineering in Computer Science

E-ISSN: 2663-3590
P-ISSN: 2663-3582
IJECS 2020; 2(1): 26-29
Received: 15-04-2019
Accepted: 20-05-2019

YL Prathapa Reddy
Head, Department of Computer Science, Sri Vijayadurga Degree College, Kurnool, Andhra Pradesh, India

Correspondence
YL Prathapa Reddy
Head, Department of Computer Science, Sri Vijayadurga Degree College, Kurnool, Andhra Pradesh, India

Design a effective parallel structure clustering approach for adaptive RGB image segmentation

YL Prathapa Reddy

DOI: https://doi.org/10.33545/26633582.2020.v2.i2a.37

Abstract
In a pattern recognition and image understanding applications, one of the most significant processes considered is the colour image segmentation. In this paper, an effective parallel structure clustering approach for adaptive unsupervised based bottom up red–green–blue (RGB) colour histogram search approach is proposed to achieve colour image segmentation. The RGB histogram is processed first by usinga double-scan procedure for the determination of important modes in every histogram. The bottom-up histogram search approach is used in the nest step to process the each mode for the RGB triplet formation. These triplets are used as the cluster centroids which cluster the pixels into regions and produce the resulting segmented image. In addition, a parallel processing structure is designed and implemented for the proposed scheme on agraphics processing unit (GPU) to reduce computational complexity. Experimental results show that the proposed algorithm outperforms state-of-the art algorithms both in terms of computational complexity and execution speed. For computational complexity, the proposed scheme running on a GPU provided19 and 15 times lower complexity than the proposed scheme running on a CPU and the *L*0-based scheme, respectively.

Keywords: Image segmentation, Histogram generation, parallel processing, low-complexity

1. Introduction

The edge information of an image is important for human visual perception [1] and is commonly used in many image processing applications. For better performance in image reconstruction, segmentation, object classification, and three-dimensional content generation, an edge-preserving smoothing scheme is used at the pre-processing step to preserve critical edges so as to maintain the main structure of a given image, while removing trivial details which are small details that need to be smoothed or continuously changed. Image segmentation is an extremely important preliminary step in a lot of pattern recognition and computer vision applications. Image segmentation divides the image into segments such that each pixel in a segment is homogeneous to all the other pixels contained by that segment [1]. Image segmentation techniques can be classified into threshold-based, region-based and clustering-based approaches [2]. Clustering techniques have a wide application in transforming pixels into regions using the similarity features (e.g. colour, texture etc.). Clustering is an unsupervised approach, having no prerequisite of labelled class data for clustering unknown data. Kmeans and fuzzy c-means (FCM) [2] are the most popular and widely used clustering techniques for image segmentation.

An important prerequisite for applying clustering techniques to an image is the provision of initialization parameters, i.e. cluster centroids and the number of clusters. The final segmented image quality heavily depends upon the initialization parameters. The process of initialization parameters determination is a challenging task both in terms of preserving the significant image features and also maintaining low computational complexity.

In this study, an efficient parallel structure clustering algorithm for adaptive unsupervised colour image segmentation is presented. In this, firstly, peaks and the corresponding colour channel intensity levels are determined for all the colour channels using the histograms. Secondly, the individual colour channel intensity levels corresponding to the peaks are combined in an efficient way to construct the sets of RGB triplets. In addition, to reduce computational complexity, we implemented the proposed scheme on a parallel processor, such as a graphics processing unit (GPU), because it has a high computational complexity. In particular, linearly quantized images with RGB components are processed into three layers to evaluate the segmentation method objectively.

After that, perceptually close RGB triplets are merged into a single RGB triplet, decreasing the number of clusters to mitigate the computational complexity and the over-segmentation effect. The set of RGB triplets serve as the initialization parameters while the amount of RGB triplets termed as the number of clusters. In the next step, K-means clusters the pixels in the image using the above-mentioned initialization parameters.

2. Literature Survey

Although several people have reported work on color segmentation, in most cases the effects of noise and in particular non-homogeneous noise are neglected. Quite often, the segmentation is performed in non-perceptually uniform color spaces, treating color as spectral information and not as a property of surfaces that is meaningful only with respect to the human vision system. For example, a color segmentation algorithm based on the physical properties of the reflected spectrum from various materials and assuming negligible sensor noise is presented earlier. Another example is the work that used a combination of scale space filters and projection of the three-dimensional (3-D) color histogram onto three one-dimensional (1-D) histograms to perform color segmentation.

Fukunaga and Hostetler [10] presented a non-parametric clustering algorithm with exemption from a pre-defined number of cluster centroids. Mean shift (MS) algorithm has the computational complexity of O(Tn2) which is far more as compared to the K-means $O(knT)$, additionally, the selection of the optimum values for various parameters in MS algorithm is non-adaptive. Yu *et al.* [4] proposed ant-colony fuzzy c-means hybrid algorithm (AFHA), an adaptive unsupervised algorithm, which takes advantage of both ant colony system and FCM techniques. In [4], the ant system [9] determines the initialization parameters while FCM clusters pixels to produce the final segmented image. The high-computational complexity of the ant system (AS) module decreases the overall efficiency of AFHA. To increase the overall efficiency of AFHA, Yu *et al.* proposed a modified version named IAFHA in [4]. IAFHA considered only a small portion of pixels instead of the whole image to determine the initialization parameters. Even though IAFHA reduces the computational complexity yet obviously there is a trade-off between computational complexity and the algorithm performance which affects the segmentation quality.

In [1], Zheng *et al.* proposed an adaptive image segmentation approach based on FCM and spatial information, the huge processing time of this approach makes it unsuitable for the real-time image segmentation. The modified K-means (MKM) algorithm presented by Zhang *et al.* [2] is another image segmentation based on adaptive unsupervised clustering. MKM repeats bisecting the clusters until the inter cluster similarity is not reached. Thus, the value of the sensitivity threshold affects the performance of the MK Malgorithm.

3. Proposed Method

K-means being a local search method possess a tendency of falling into local minima during the iterative optimization process. However, the performance of the K-means clustering technique is extremely sensitive to the initialization parameters, i.e. initial cluster centres and the number of clusters. Thus, the image segmentation results produced by K-means are heavily dependent upon the selection of initialization parameters. Usually, determining good initialization conditions becomes a laborious process due to the need for experiments being performed. The flow diagram of our proposed approach is presented in Figure (1).

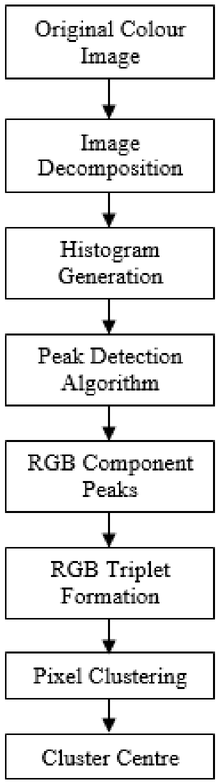

Fig 1: Proposed block diagram

The proposed method adaptive unsupervised efficient parallel structure clustering algorithm aims to improve the conventional K-means algorithm's colour image segmentation quality by optimizing the determination of initialization parameters. More specifically, every pixel in an RGB colour image is a triplet composed of red, green, and blue colour channel intensities. The combination of these three colour channels provides global information about an image. The peak finding algorithm (PFA) is applied to determine the initialization parameters for the K-means algorithm in a novel way. First, PFA is applied on R, G, and B colour channels, determining significant peaks in each of the colour channel histograms with the parallel structure mechanism which is implemented by the parallel processor, such as a graphics processing unit (GPU). As each pixel in a RGB image is represented by a 3D feature vector, the two unknown channel intensity values are determined by applying PFA upon the histogram of the pixels corresponding to the initially determined peaks channel intensities. With the addition of two more colour channel intensity values in each of the initially determined peak channel intensity, RGB triplets are completed. In the next step, perceptually close RGB triplets are merged, producing the initialization parameters for the K-means clustering algorithm.

3.1 The region splitting can be implemented as follows

1. Select significant peaks in each colour channel histogram by detecting the channel intensities with a

maximum occurrence value among the neighbouring intensity levels. The peaks are detected in the red, green, and blue colour channels.

2. Organize RGB triplets by determining the missing colour channel intensity levels by detecting the peaks in the histogram of the pixels belonging to the already detected colour channel intensity level. The peaks are detected in the red colour channel.
3. Calculate the Manhattan distance between all the pairs of RGB triplets using the following equation. The green and blue color channels are detected peaks for a given red channel peak.

$$D(c_i, c_j) = |R_i - R_i| + |G_i - G_i| + |B_i - B_i|, \ \forall i \neq j, \quad (1)$$

Where $1 \leq i \leq M$, $1 \leq j \leq N$, M and N represent the number of RGB triplets, R_i, G_i and B_i are the intensities of the red, green, and blue colour channels of the ith RGB triplet. (Note: the Manhattan distance is found to be a better similarity measure than the Euclidean distance as Manhattan has a stable visual colour similarity whereas the later produces a wider variation of the same colour.) The segmentation results obtained after K-means clustering can be represented as $U_{i-1}^{n} c_i = C, U_{i-1}^{n} l_i = L,$ and $U_{i-1}^{n} u = U,$ where each cluster c_i is a set of pixels: $c_i = \{p_k\} 1 \leq k \leq N_i, N_i$ is the number of pixels that belong to the cluster c_i, l_i is the label assigned to c_i cluster elements, u_i is the cluster centre of c_i and U is the set containing the entire cluster centres. $\sum_{i-1}^{n} N_i = N$, where n is the number of clusters and N is the total number of pixels in the input image C.

3.2 GPU-based parallel processing for the proposed scheme

Reduction of the computational cost of image segmentation methods is required for several varieties of applications. For this purpose, parallel processor-based complexity reduction can be considered. In this work, an effective and fast image segmentation method on a GPU is presented by designing a parallel processing structure. A CPU core supports several software threads at a time for parallel processing, and its operation is optimized for sequential processing. In contrast, a GPU is architecturally composed of hundreds of cores that execute in parallel with other cores. Each core can execute a sequential thread, and all the cores in the same group execute the same instruction simultaneously. The kernels, which are units of a parallel program on a GPU, are executed sequentially. Especially, the threads for a kernel are identified in terms of a hierarchical indexing structure. In addition, threads are grouped into blocks, and blocks are grouped into a grid. Each thread has a unique local index in its block, and each block has a unique index in the grid. To design a parallel processing scheme for image segmentation on a GPU, we have to divide an image into several blocks. The blocks, which have independent physical memory space and processing units, cannot share input data and results with other blocks, and they cannot send results to the other blocks. For fast processing, the calculation of the smoothed image is performed using FFTs, and thus its complexity critically affects segmentation complexity. A two-dimensional-CUDA-based FFT (CuFFT) library, a fast FFT on a GPU, is used for the calculation of the segmented image to reduce computational cost. The derivatives of the segmented image obtained at the previous iteration are executed and then computed an optimal solution for the intensity components R, G and B module, the smoothed image is obtained from the pre-operated terms and updated intensity components R, G and B and then the weight parameter b is updated by multiplying by the increment ratio as b =b ×*ratio*. For the iteration, the updated b is compared with the predefined maximum value, and the segmentation is repeated until the predefined condition is met.

4. Results

The proposed algorithm was compared with the MATLAB implementation results of CPU graph based and really quick shift method. The latest version of the Berkeley segmentation dataset and benchmark (BSD500) [21, 23] was selected to perform the experiments. Some sample images from BSD500 data set given in Figure (2) to show the quality and characteristics of the segmentation result. The segmentation results on these sample images for the proposed method are depicted in Figure (3) compared to other two methods.

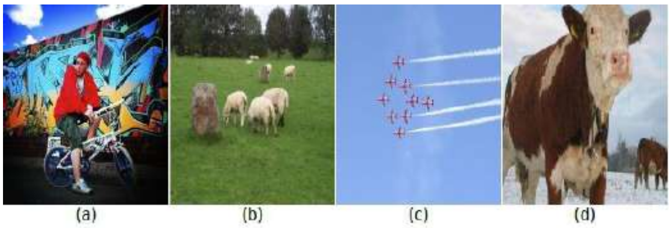

Fig 2: Input images used for evaluation

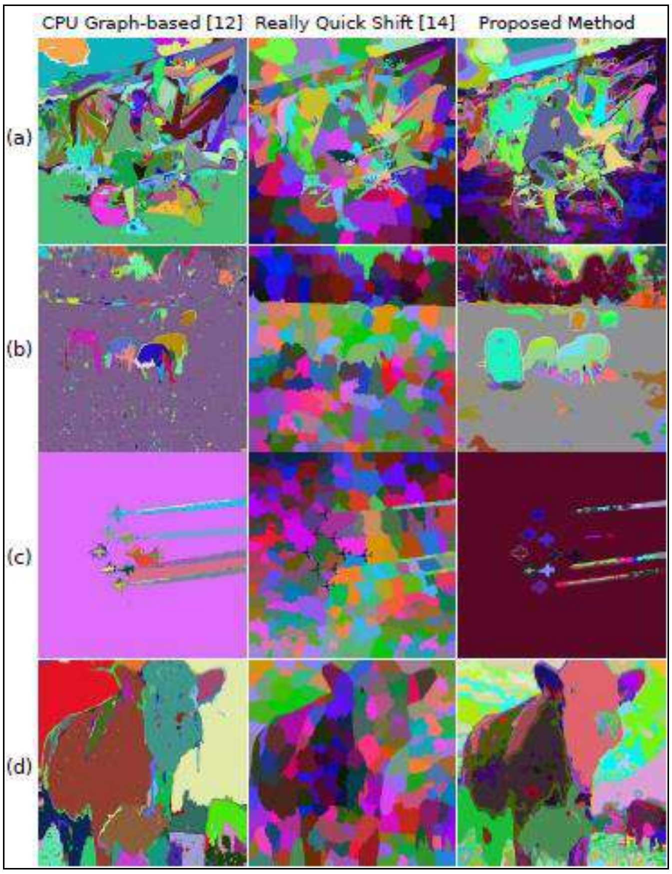

Fig 3: Comparision visualization of segmentation results

To evaluate the computational complexity for sequential and parallel processing, we implemented the proposed scheme for sequential processing in MATLAB. The previous four images were used, and the average computational time was measured, as shown in Table IV. For the *L*0-based and the proposed schemes, iterative operation for the smoothing step was repeated 15 times. The proposed approach requires more computational time than the*L*0-based scheme for sequential processing in the MATLAB implementation. However, when the proposed scheme was implemented on a GPU, the computational time was reduced by about 16 and 18 times compared to the proposed and *L*0-based schemes, respectively, implemented on a CPU.

Table 1: Computational time with different schemes

Sl. No	Method	Computation time (s)			
1	Lo-based scheme in MATLAB	2	.	6	3
2	Proposed scheme in MATLAB	2	.	1	2
3	Proposed scheme on a GPU	0	.	1	8

5. Conclusion

In this paper, an effective parallel novel adaptive unsupervised parallel structure segmentation algorithm based on the histogram thresholding is proposed to generate high-quality segmented images. The proposed algorithmis capable of automatic determination of the significant number of clusters and optimized cluster centroid for a given set of pixels. Moreover, the segmentation cost function is separated into pre-operation and updated terms for the parallel processing of the proposed scheme to reduce computational cost. Then, the pre-operation terms were processed before the iterative operation, and the updated terms were computed using the iterative scheme that was implemented in CUDA on a GPU. Moreover, the experimental results on a variety of images prove the proposed algorithm is extremely agile as compared to other algorithms reported in this study, making the proposed algorithm perfectly suitable for real-time image segmentation applications.

6. References

1. Zheng J, Zhang D, Huang K *et al*. ‘Adaptive image segmentation method based on the fuzzy C-means with spatial information’, IET Image Process. 2018;12(5):785–792.
2. Zhang Q, Chi Y, He N. ‘Color image segmentation based on a modified K-means algorithm’. Proc. 7th Int. Conf. on Internet Multimedia Computing and Service, Zhangjiajie, Hunan, China, 2015, pp 46.
3. Tariq S. An Introduction to GPU Computing and CUDA Architecture, NVIDA Corporation, 2011.
4. Yu Z, Au OC, Zou R *et al*. ‘An adaptive unsupervised approach toward pixel clustering and color image segmentation’, Pattern Recognit 2010;43(5):1889-1906.
5. Ugarriza LG, Saber E, Vantaram SR *et al*. ‘Automatic image segmentation by dynamic region growth and multiresolution merging’, IEEE Trans. Image Process 2009;18(10):2275–2288
6. Ilea DE, Whelan PF. ‘CTex – an adaptive unsupervised segmentation algorithm based on color-texture coherence’, IEEE Trans. Image Process 2008;17(10):1926–1939
7. Chan RH, Ho CW, Nikolova M. Salt-and-Pepper Noise Removal by Median-Type Noise Detectors and Detail-Preserving Regularization, IEEE T. on Image Processing, 2005, 14(10).
8. Wang Z, Bovik C, Sheikh HR, Simoncelli EP. Image quality assessment: from error visibility to structural similarity, IEEE T. on Image Processing 2004, 13(4).
9. Colorni A, Dorigo M, Maniezzo V. ‘Distributed optimization by ant colonies’, in Varela, F., Bourgine, P. (Eds.): ‘Int. J Image Process. (IJIP)’ (MIT Press, 1992, pp 134–142
10. Fukunaga K, Hostetler L. ‘The estimation of the gradient of a density function, with applications in pattern recognition’, IEEE Trans. Inf. Theory 1975;21(1):32–40