Tania, S., et al. (2019). Cuboid Colour Image Segmentation Using Intuitive Distance Measure. 2018 International Conference on Image and Vision Computing New Zealand, IVCNZ 2018, IEEE Computer Society.

Which has been published in final form at:

https://doi.org/10.1109/IVCNZ.2018.8634676\

Cuboid Colour Image Segmentation using Intuitive Distance Measure

Sheikh Tania, Manzur Murshed, Shyh Wei Teng, and Gour Karmakar
School of Science, Engineering and Information Technology
Federation University Australia, Gippsland Campus
Churchill Vic 3842, Australia
{t.tania, manzur.murshed, shyh.wei.teng, gour.karmakar}@federation.edu.au

***Abstract*—In this paper, an improved algorithm for cuboid image segmentation is proposed. To address the two main limitations of the recently proposed cuboid segmentation algorithm, the improved algorithm substitutes colour quantization in HCL colour space with infinity norm distance in RGB colour space along with a different way to impose area thresholding. We also propose a new metric to evaluate the quality of segmentation. Experimental results show that the proposed cuboid segmentation algorithm significantly outperforms the existing cuboid segmentation algorithm in terms of quality of segmentation.**

***Keywords*—Cuboid segmentation, infinity norm, segmentation quality metric**

I. Introduction

Image segmentation is defined as a process of decomposing an image into different regions based on some criteria which depend on the application under consideration [1]. The output of image segmentation is a collection of non-overlapping segments (regions) which make up the whole image so that a region or an object of interest can be detected and thus separated. Each of these regions contains pixels which are similar in terms of certain chosen characteristics or attributes, e.g. colour, intensity, and texture. On the other hand, neighbouring regions are to be distinguishable in characteristics. The output of image segmentation greatly assists users or applications in effectively detecting and recognizing regions or objects of interest.

Image segmentation is essential in numerous cutting edge applications like content-based image retrieval, object detection and recognition, medical imaging, traffic control system, and video surveillance. This is also an important processing step to advance the robotic automation further.

Region-based segmentation algorithms can be classified primarily as deterministic and probabilistic. Deterministic algorithms can be classified as supervised (e.g. Mean Shift (MS) [2], Fuzzy C-Means (FCM) [3], and K-Means) and unsupervised (e.g. pulse coupled neural network [4]). Non-deterministic algorithms can be classified as parametric (e.g. Support Vector machine [5], and Markov random field [6]) and non-parametric (e.g. Expectation Maximization [7]). But all those segmentation algorithms do not embed spatial relationship of pixels explicitly.

In [8], the authors proposed a new algorithm for image segmentation in which segments are designated in the form of cuboids. The main idea is to use a greedy heuristic to recursively split a rectangular image into two rectangular halves with an optimal split line, orthogonal to one of the axes so that the value of distance metric is maximized.

The advantages of the cuboid segmentation algorithm are as follows:

- The cuboid segmentation algorithm enforces the spatial relationship among neighbouring pixels which is generally exhibited in a region or object of interest.
- The division of an image into rectangular grids has the potential to address the problem for selecting and prioritizing the blocks used in MPEG4/H.264 video compression technology.
- The algorithm is highly computationally efficient compared to the existing arbitrary-shaped segmentation techniques which can extend the real time video transcoding and processing using low powered handheld mobile devices.
- In image retrieval applications, the query object of a query image is usually specified by one or many rectangular boundaries. Therefore the regions derived using cuboid segmentation better match the manner the query object is specified which improves the matching process.

The study of the effectiveness of the cuboid segmentation algorithm in [8], which we coined as CSeg'17 for the rest of this paper, is on how this algorithm contributes to image retrieval performance. In this paper, we study the effectiveness of CSeg'17 by focusing on the quality of segmentation. From our study, we have identified the following limitations in CSeg'17:

- CSeg'17 uses the first and second orders of RGB colour moments, and converts them into the HCL colour space and then uses colour quantization. Because of coarse colour quantization, a region with similar colour could be unnecessarily further partitioned into two regions.
- When determining if a region should be further partitioned into two sub-regions, CSeg'17 applies an area threshold on each sub-region which results in sub-optimal segments and poorer preservation of the order of segmentation with respect to the number of segments.

To address the above limitations, we propose an improved CSeg algorithm which uses a more intuitive distance measure namely, infinity norm in the RGB colour model. Our proposed algorithm also uses a technique to make sure that segmented regions are distinguishably different. Finally, as existing metrics for evaluating the quality of image regions derived by segmentation algorithms are not suitable for cuboid regions,

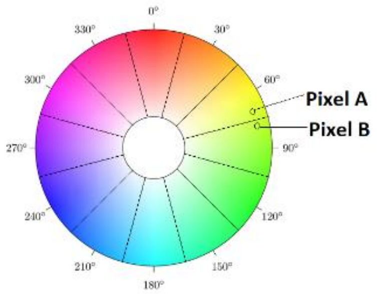


Fig. 1. HCL bi-cone slice showing the position of two pixels belonging to different quantized H values that are perceptually similar.

we have also proposed a new evaluation metric for this purpose. This metric can also be used for other non-cuboid regions. Our performance studies show that our proposed algorithm performs far better than CSeg'17.

The rest of the paper is organised as follows: section II describes the CSeg'17 algorithm and its main limitations. Section III describes our proposed improved cuboid segmentation algorithm. The performance studies are presented in Section IV. Section V provides the conclusion.

II. Background

In this section, the cuboid segmentation algorithm in [8] is described followed by a discussion of its main limitations.

A. *Cuboid Segmentation Algorithm*

The CSeg algorithm in [8] (CSeg'17) uses the first and second orders of colour moment (mean and standard deviation) features to distinguish one part of an image from another part. As the HCL (Hue-Chroma-Luminance) colour space $(0° \le H < 360°\ and\ 0 \le C \le K \le 1)$ is perceptually uniform, moments in RGB cube space are transformed first to HCL bi-cone space and again to a cube space $(-1 \le X', Y', Z' \le 1)$. Let $\Psi_k^I = ({X'}_K^I,\ {Y'}_K^I,\ {Z'}_K^I)$ be the k-th order colour moment of image I in the transformed cube space. The colour-contrast distance between two images I and J can be estimated as

$$d(I,J) = \frac{1}{2}\sum_{k=1}^{2} \omega_k \left\|\Psi_k^I - \Psi_k^J\right\|, \tag{1}$$

where $0 \le \omega_k \le 1$ is the normalized weight used for the k-th colour moment such that $\sum_k \omega_k = 1$. As each of the norms above can have a maximum value of 2, the factor ½ is used to normalize the distance in the range [0, 1]. The HCL bi cone colour space is considered as a coarse colour space of 65 quantized colours. The coarse colour of an image is catalogued by determining its Hl (Hue-luninance) pair in the coarse colour space.

CSeg'17 uses the abovementioned distance metric and colour quantization to partition an image I of width X and height Y recursively by splitting into horizontal or vertical rectangular halves I_{s*}^1 and I_{s*}^2 by maximizing their distance. For a user-defined target number of cuboids (regions) n_s, an area threshold A_{X,Y,n_s} is used to restrict a split s by $\min(|I_{s,1}|, |I_{s,2}|) < A_{X,Y,n_s}$. A split is considered valid only if the area of each of the two sub-cuboids meet the area threshold and they belonged to different coarse colours.

B. *Limitations of CSeg'17*

The use of colour quantization in CSeg'17 limits the algorithm from producing better segment quality. In quantization, a large set of input values are mapped into a much smaller number of bins. In colour quantization, values at the boundaries of neighbouring bins are perceptually very similar and this causes issues when matching since matching is done based on the occurrences between bins. For example, pixels A and B in Fig. 1 are perceptually very similar. However, pixel A belongs to hue 'yellow' bin whereas pixel B belongs to hue 'lime' bin. In CSeg'17, since the criteria for partitioning are based on quantized colours, it is more likely to lead to unnecessary partitioning of a cuboid which contains perceptually similar colours but the colours belong to different quantized bins.

The manner by which the area threshold is used in CSeg'17 to determine if a cuboid should be partitioned is another limitation. Since the area threshold is set as a function of the user-defined target number of cuboids, i.e. usually by dividing the image area with the targeted number of cuboids, the area threshold might still be too high for each half-cuboid to meet. This might force a cuboid to be partitioned into two sub-cuboids of lower colour uniformity in each, even though this cuboid could be partitioned with another split line to get two sub-cuboids with better lower colour uniformity in each. For example, the red arrow in Fig. 6(a) shows a vertical split line of CSeg'17 which produces two lower uniformity sub-cuboids. The red arrow in Fig. 6(b) shows the split line which will produce two sub-cuboids of optimal uniformity in each.

III. Proposed Improved CSeg Algorithm

In this section, we propose our improved CSeg algorithm that uses infinity norm distance in the RGB colour space to form the distance metric. We also describe the greedy optimization technique for image segmentation, we have adopted to avoid the poor segmentation quality of CSeg'17. For the rest of this paper, we will refer to our proposed improved CSeg algorithm as *CSeg'18*.

A. *Infinity Norm Distance*

For two points $(x_1, \dots, x_n)$ and $(y_1, \dots, y_n)$ in the Euclidean space R^n, the infinity-norm distance is defined as follows:

$$D = \lim_{l \to \infty}\left(\sum_{i=1}^{n} |x_i - y_i|^l\right)^{1/l} = \max_{1 \le i \le n} |x_i - y_i|, \tag{2}$$

which is the maximum of the absolute values of the coordinates. As the RGB colour space is not perceptually uniform, the Euclidean distance between two colours may provide a difference which is mathematically correct but would not conform to humans' perception. To address this issue, we propose to use infinity norm distance because the RGB colour space is orthogonal and human eyes perceive the difference of two colour by the maximum difference in the RGB value. In [2], instead of the Euclidean distance, infinity norm distance is used to measure the neighbourhood between two pixels thus reducing the calculation time.

If $\mu_I^{(R,G,B)}$ is the first order raw RGB colour moment of Image I, then the colour-contrast distance between two Images I and J can be estimated using the infinity-norm distance as

$$D_{I,J} = \max\left(\left|\mu_I^R - \mu_J^R\right|, \left|\mu_I^G - \mu_J^G\right|, \left|\mu_I^B - \mu_J^B\right|\right) \quad (3)$$

Thus by using the infinity norm distance in the RGB model, the colour quantization limitation in CSeg'17 is avoided in CSeg'18. The proposed distance in the RGB model still well simulate how human eyes perceived colour differences like using the Euclidean distance in the HCL model.

B. Proposed Algorithm

Given a user-defined number of cuboids n_s, we need to first determine the required area threshold A_{X,Y,n_s} and contrast threshold C_{X,Y,n_s}. It is obvious that n_s monotonically decreases with the increased values of both A_{X,Y,n_s} and C_{X,Y,n_s}, but for a certain value of A_{X,Y,n_s}, an image could be partitioned into n_s segments in many different ways with different values of C_{X,Y,n_s} but all those ways do not guarantee the compactness of segmentation results. That is why we need to maximize the value of contrast threshold among all those possible values.

Image I can be split into two sub-cuboids I_i^1 and I_i^2 of sizes $i \times Y$ and $(X - i) \times Y$ pixels respectively, using a vertical line $x = i + 0.5$ in $X - 1$ ways with $i \in \{1, 2, \dots X - 1\}$. Similarly, it can be split into two sub-cuboids I_{X-1+j}^1 and I_{X-1+j}^2 of sizes $X \times j$ and $X \times (Y - j)$ pixels respectively, using a horizontal line $y = j + 0.5$ in $Y - 1$ ways with $j \in \{1, 2, \dots, Y - 1\}$.

As discussed in Section II.B, in CSeg'17, both the two sub-cuboids needs to satisfy the area threshold for a cuboid to be partitioned. To address this limitation in our proposed algorithm, the area threshold, A_{X,Y,n_s} is applied on the whole cuboid (line 3 in **Algorithm 1**). This allows a cuboid to be partitioned even though one of the two half-cuboid is much smaller than the other and better colour uniformity in each half-cuboid is maintained.

A split is considered valid if the colour contrast distance of the two half-cuboids meets the contrast threshold i.e.,

$$v(s|I, n_s) = \begin{cases} D_{I_i^1, I_i^2} \geq C_{X,Y,n_s}, & s < X; \\ D_{I_{X-1+j}^1, I_{X-1+j}^2} \geq C_{X,Y,n_s}, & s \geq X. \end{cases} \quad (4)$$

The colour contrast distance of the half-cuboids is the objective function,

$$f(s|I) = D_{I_s^1, I_s^2}. \quad (5)$$

Then the greedy optimization heuristic to find the best split of I from the possible $X + Y - 2$ ways as:

$$\begin{aligned} &\underset{1 \leq s \leq X+Y-2}{\text{maximize}} \quad f(s|I) \\ &\text{subject to} \quad v(s|I, n_s). \end{aligned} \quad (6)$$

A hierarchical partitioning algorithm may be designed by recursively splitting the two half-cuboids using the optimal split s^*. The algorithm terminates when all possible ways of splitting are found to be obsolete that means if one or both of the area threshold and contrast threshold is not satisfied. The algorithm is now formally presented below as **Algorithm 1**.

```
Algorithm 1: CSeg'18 $(I, X, Y, n_s)$
1:  $d_{\max} = 0$;
2:  $s^* = 0$;
3:  if $\text{area}(I) \geq A_{X,Y,n_s}$ then
4:      for $s = 1,2, \dots, X + Y - 2$ do
5:          if $v(s|I, n_s) \wedge f(s|I) > d_{\max}$ then
6:              $d_{\max} = f(s|I)$;
7:              $s^* = s$;
8:          end if
9:      end for
10: end if
11: if $d_{\max} > 0$ then
12:     CSeg'18$(I_{s^*}^1, X, Y, n_s)$;
13:     CSeg'18$(I_{s^*}^2, X, Y, n_s)$;
14: end if
```

C. Optimization

In CSeg'17, it is possible to partition a cuboid into two sub-cuboids of the same colour. This happens when partitioning is done iteratively in either horizontal or vertical direction. For example, a cuboid is partitioned vertically into two sub-cuboids. Then in the next iteration, the left sub-cuboid is considered to be partitioned into two sub-cuboids based on only their colour difference. However, there is a possibility that it can result in a sub-cuboid in the right side that is of the same colour as the sub-cuboid to the right side found in the iteration prior to that. To resolve this issue, we have considered the distances of these hazardous cuboids from its neighbour of prior depth that means partition is secured if contrast threshold for both prior and immediate neighbour is met.

IV. Perfomance Studies

In this section, we will first propose a new metric which is more suitable for evaluating the quality of regions partitioned using Cuboid-based segmentation algorithms. We will then present our experimental results to compare the performances of CSeg'17 and CSeg'18.

A. Proposed new segmentation quality metric

As cuboid segmentation is different from other types of segmentation, conventional evaluation metrics like Sensitivity, Specificity, Accuracy, Dice and Jacard indexes, Haussdorf distance or the maximum cardinality similarity measure, and ROC curves are not appropriate to measure the performance of quality of resultant regions. The reason is these metrics do not correlate the quality of segmentation with the number of segments. Our proposed metric is able to measure the performance of cuboid segmentation as well as traditional segmentation algorithms.

Let $\mathcal{S}_I = (\mathcal{S}_I^1, \dots, \mathcal{S}_I^{|\mathcal{S}_I|})$ be a segmentation outcome of image I with $|\mathcal{S}_I|$ segments such that $\sum_{k=1}^{|\mathcal{S}_I|} |\mathcal{S}_I^k| = |I|$ where $|\cdot|$ denotes the number of pixels in an image or image segment. With the assistance of a manually-segmented ground-truth binary image G_I, where

$$G_I(x, y) = \begin{cases} 1, & \text{if pixel } (x, y) \in \text{foreground;} \\ 0, & \text{otherwise,} \end{cases} \quad (7)$$

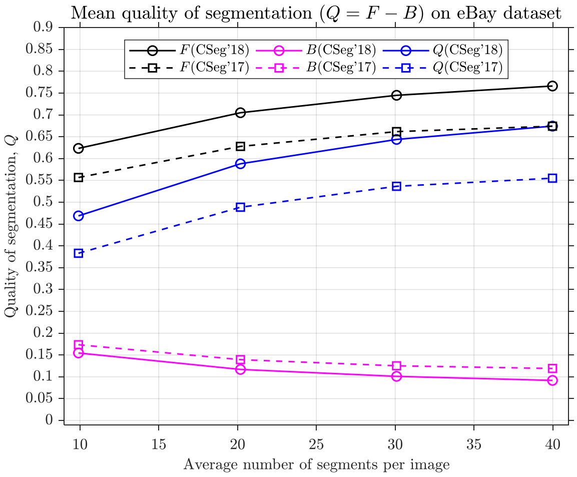


Fig. 2. Quality comparison of CSeg'17 and CSeg'18 in terms of number of segments.

we can define a segmentation-quality metric by considering the number of foreground pixels in each segment.

Let $f_{\mathcal{S}_I}(k) = \sum_{(x,y)\in\mathcal{S}_I^k} G_I(x,y)\,/|\mathcal{S}_I^k|$ and $b_{\mathcal{S}_I}(k) = 1 - f_{\mathcal{S}_I}(k)$ denote the proportion of foreground and background pixels respectively in segment $\mathcal{S}_I^k$, for all $1 \le k \le |\mathcal{S}_I|$. With imperfect segmentation $\mathcal{S}_I$, the likelihood of segment $\mathcal{S}_I^k$ being foreground can be estimated from the proportion of foreground pixels in the segment as $p_{f,\mathcal{S}_I}(k) = \sum_{(x,y)\in\mathcal{S}_I^k} G_I(x,y)\,/\sum_{(x,y)\in I} G_I(x,y)$for all $1 \le k \le |\mathcal{S}_I|$.

Similarly, the likelihood of that segment being background can be estimated from the proportion of background pixels in the segment as $p_{b,\mathcal{S}_I}(k) = (|\mathcal{S}_I^k| - \sum_{(x,y)\in\mathcal{S}_I^k} G_I(x,y))/(|I| - \sum_{(x,y)\in I} G_I(x,y))$ for all $1 \le k \le |\mathcal{S}_I|$. Note that both $p_{f,I}$ and $p_{b,I}$ are probability density functions as $\sum_{k=1}^{|\mathcal{S}_I|} p_{f,\mathcal{S}_I}(k) = 1$ and $\sum_{k=1}^{|\mathcal{S}_I|} p_{b,\mathcal{S}_I}(k) = 1$. Hence, we can use them to estimate the expected proportion of foreground pixels in foreground and background segments as

$$F_{\mathcal{S}_I} = E_f[f_{\mathcal{S}_I}(k)] = \sum_{k=1}^{|\mathcal{S}_I|} f_{\mathcal{S}_I}(k) p_{f,\mathcal{S}_I}(k) \tag{8}$$

and

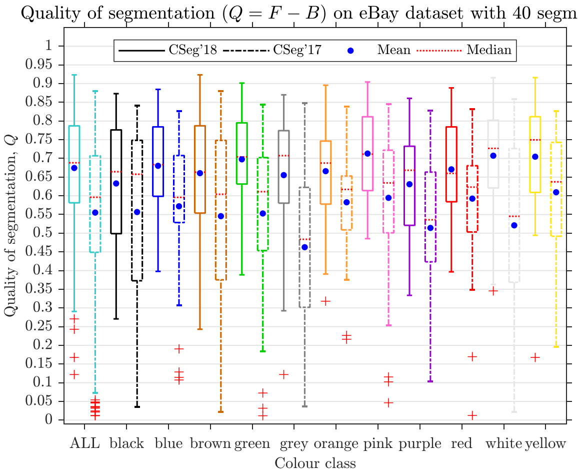


Fig. 3. Statistical quality comparison of CSeg'17 and CSeg'18 in terms of colour class.

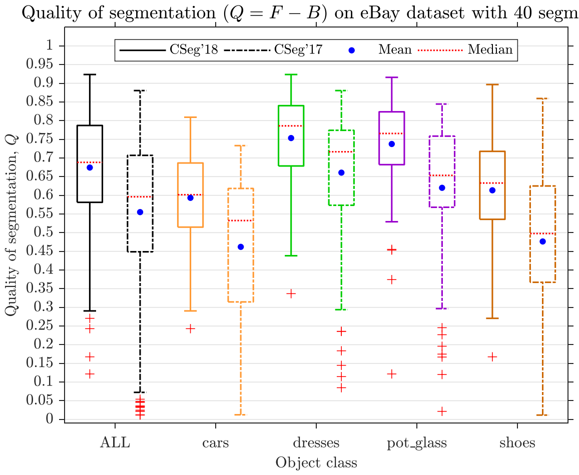


Fig. 4. Statistical quality comparison of CSeg'17 and CSeg'18 in terms of object class.

$$B_{\mathcal{S}_I} = E_b[f_{\mathcal{S}_I}(k)] = \sum_{k=1}^{|\mathcal{S}_I|} f_{\mathcal{S}_I}(k) p_{b,\mathcal{S}_I}(k). \tag{9}$$

A segmentation-quality metric can now be defined as $Q_{\mathcal{S}_I} = F_{\mathcal{S}_I} - B_{\mathcal{S}_I}$. The range of $Q_{\mathcal{S}_I}$ is [0,1]. Without any segmentation, since, $|\mathcal{S}_I| = 1$, $p_{f,\mathcal{S}_I}(1) = p_{b,\mathcal{S}_I}(k) = 1$ and then $F_{\mathcal{S}_I} = B_{\mathcal{S}_I} = 1$ .Therefore, $Q_{\mathcal{S}_I} = 0$. With perfect segmentation, we have $Q_{\mathcal{S}_I} = 1$ as $f_{\mathcal{S}_I}(k) = 1$ when $p_{f,\mathcal{S}_I}(k) > 0$ and if $p_{b,\mathcal{S}_I}(k) > 0$, $f_{\mathcal{S}_I}(k) = 0$. This implies that $F_{\mathcal{S}_I} = 1$ and $B_{\mathcal{S}_I} = 0$. Therefore to compare segmentation-quality of different segmentation algorithms, we need to keep the number of segments produced by all algorithms the same or as close as possible. Therefore, the higher the quality of segmentation, the higher the value of Q will be. Note that Q is expected to improve with an increased number of segments, irrespective of segmentation algorithm. This is because the more the number of segments, the more accurate separation of foreground and background pixels will be, eventually enhancing the performance of retrieval.

B. Experimental results

In this section, we present our experimental results for evaluating the performances of CSeg'18 to that of CSeg'17 using our proposed evaluation metric. We used the eBay test dataset [9] for our experiments. This dataset consists of four

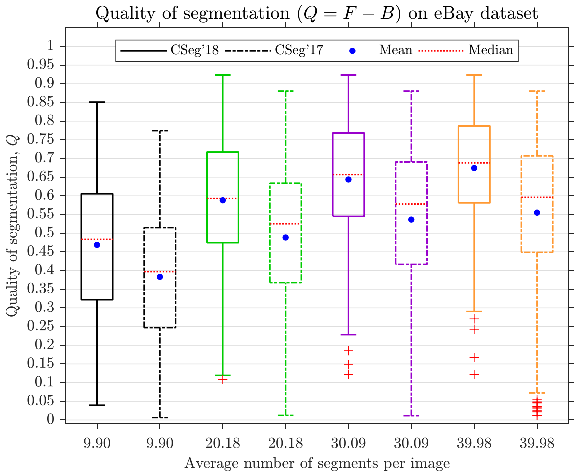
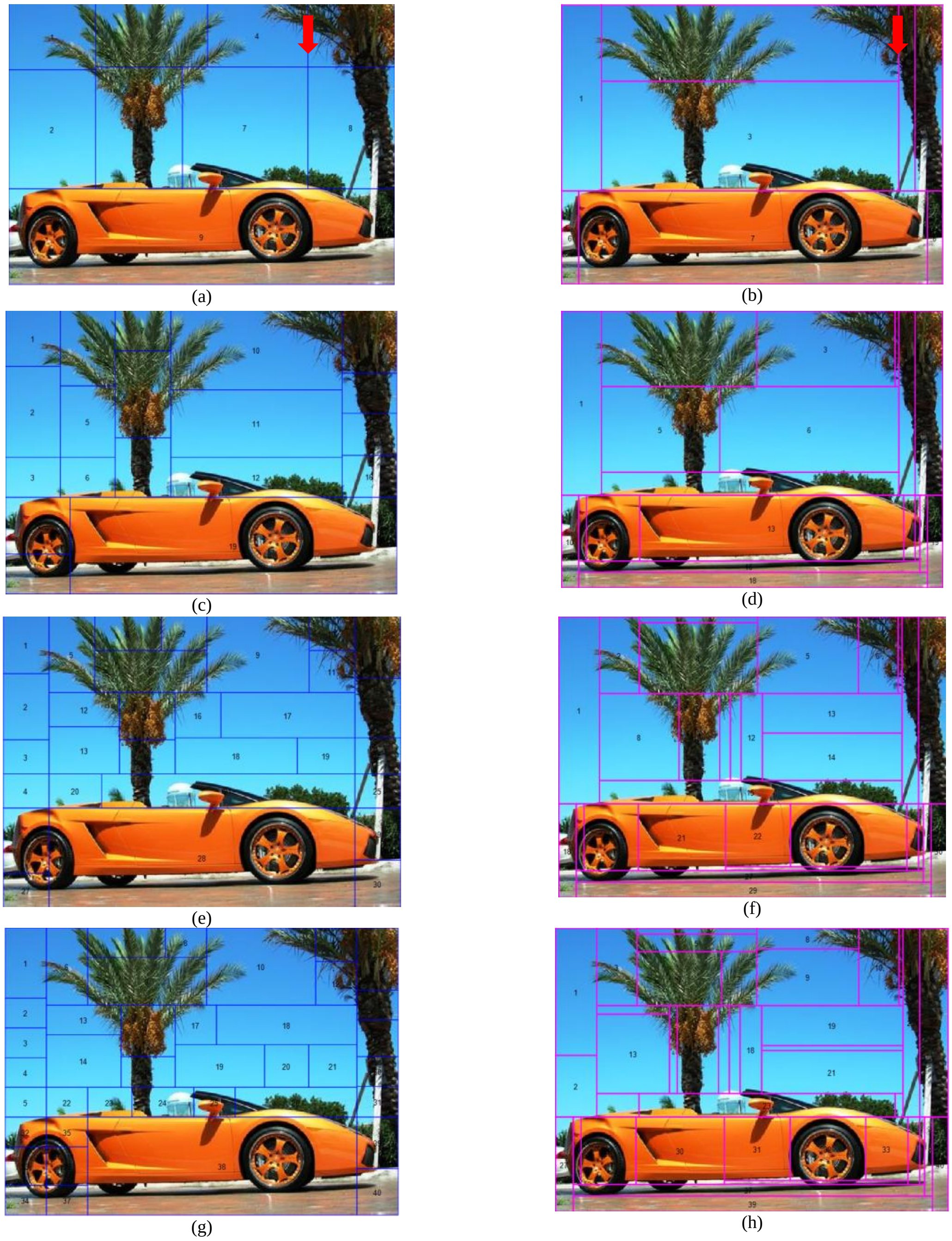


Fig. 5. Statistical quality comparison of CSeg'17 and CSeg'18 in terms of average number of cuboids per image.



(a) (b) (c) (d) (e) (f) (g) (h)

Fig. 6. Visual comparison of CSeg’17 and CSeg’18 for different numbers of cuboids. Output of CSeg’17 for (a) $n_s = 10$, (c) $n_s = 20$, (e) $n_s = 30$, (g) $n_s = 40$. Output of CSeg’18 for (b) $n_s = 10$, (d) $n_s = 20$, (f) $n_s = 30$, (h) $n_s = 40$.

object-classes: cars, dresses, pot glasses and shoes. Each of these classes is again divided into 11 colour-classes. In each colour class of an object, there are 12 images. Therefore, the dataset has in total $N = 4 \times 11 \times 12 = 528$ images. The foreground object of an image determines its object-class and colour of that object determines its colour-class.

Each image in the dataset is also accompanied with a binary mask image identifying the pixels belonging to the foreground object. To compare the performance of the two algorithms, there is a pre-requisite that the value of n_s needs to be same for both approaches. Since the contrast threshold plays the primary role in CSeg’18, it was not possible to ensure that condition for all the images in the dataset. However, we generated segmentation results for four different values of n_s (10, 20, 30, and 40). Out of the 2112 (528 × 4) images used with all four target, only 166 (7.9%) images have an unmatched number of cuboids. All the results are based on the matched images.

A comparison of CSeg’17 and CSeg’18 is presented in Table I and Fig. 2, for all four number of cuboids. It is observed that quality increases gradually with the increased number of cuboids for both algorithms. The values of both *F* and *B* improve as the number of cuboids increases but the improvement of *B* is sharper than *F*. The average improvement of quality is 21.06% considering all the number of cuboids.

We also use the box plot to compare the quality of segmentation algorithms statistically. In Fig. 3, the statistical quality comparison of segmentation ($n_s = 40$) is presented in terms of colour and it is obvious that the mean and median quality of CSeg’18 is greater than that of CSeg’17 for all the

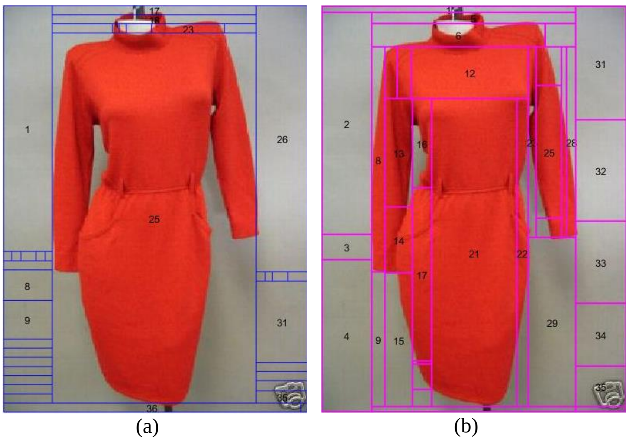


Fig. 7. Visual comparison of (a) CSeg'17 and (b) CSeg'18 for 'dress'.

images together as well as for each colour class. For CSeg'18, the maximum average quality is noticed for the pink coloured object while the minimum is noticed for black. On the contrary, for CSeg'17 the maximum average quality is noticed for the yellow coloured object while the minimum is noticed for grey. The interquartile range (IQR) is smaller for the proposed algorithm in maximum (8 out of 11) colour classes which reveals the quality of the new algorithm varies less from image to image.

The statistical quality comparison of CSeg'17 and CSeg'18 in terms of the object class and the number of cuboids is presented in Figs. 4 and 5, respectively. In all cases, CSeg'18 outperforms CSeg'17. CSeg'18 expresses maximum average quality of segmentation for object class 'dress' while CSeg'17 expresses for 'pottery'. The minimum average quality of segmentation is expressed for 'car' by both of the algorithms.

Fig. 6 gives the segmentation outcome of a car image for both the algorithms for all different number of cuboids for observing the improvements visually. In Fig. 6(a), the split line between cuboids 7 and 8 is unnatural as there is a large amount of sky portion in cuboid 8 and the line needs to be far right for perfect segmentation. The reason is the area threshold was not satisfied by the right half cuboid. But in Fig. 6(b), a vertical line separates the sky part from the tree thus resulting into higher quality segmentation.

Figs. 7 and 8 present the segmented images of 'dress' and 'pottery' and both images present better segmentation produced by CSeg'18 than that of CSeg'17. In all segmentation results of CSeg'18, it is clearly noticed that the highest proportion of foreground is captured by the cuboids on objects as well as the highest proportion of background is captured in non-object cuboids which is not exhibited in the results of CSeg'17 with many redundant partitions. For example, in Fig. 7(a), the cuboid 25 contains the maximum part of the object with some part of the background but in Fig. 7(b), that cuboid is divided into 'foreground' as well as 'background' cuboids, thus preserving the motive of segmentation. Moreover, the part of the image labeled as cuboid 4 in Fig. 7(b) is segmented into 11 cuboids in Fig. 7(a). This poor segmentation output of CSeg'17 shown in Fig. 7(a) is caused by colour quantization as these cuboids contain colours belonging to different coarse colour bins.

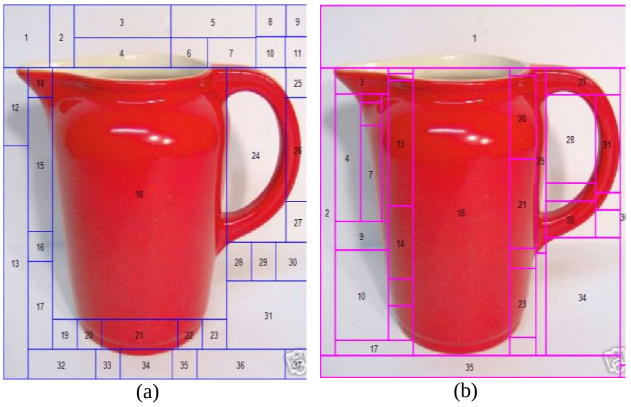


Fig. 8. Visual comparison of (a) CSeg'17 and (b) CSeg'18 for 'pottery'.

V. Conclusion

We have proposed an improved cuboid segmentation algorithm by addressing the limitations of the algorithm in [8]. At the same time an innovative evaluation metric for measuring performance of both traditional and cuboid segmentation is presented. Experimental results show our proposed algorithm significantly outperforms the algorithm in [8].

References

[1] R. C. Gonzalez and R. E. Woods, "Digital Image Processing," 2nd ed., Pearson Education, ISBN: 81-7808-629-8.

[2] D. Domínguez and R. R. Morales, "Convergence of the mean shift using the infinity norm in image segmentation," *J. Patt. Recog. Research*, vol. 6, no. 1, 2011.

[3] Y. Li and Y. Shen, "Fuzzy C-means clustering based on spatial neighborhood information for image segmentation," *J. Systems Engineering and Electronics*, vol. 21, no. 2, pp. 323- 328, 2010.

[4] M. Murugavel and J. M. Sullivan Jr, "Automatic cropping of MRI rat brain volumes using pulse coupled neural networks," *NeuroImage*, vol. 45, no. 3, pp. 845–854, 2009.

[5] B. Han and L. S. Davis, "Density-based multifeature background subtraction with support vector machine," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 5, pp. 1017–1023, 2012.

[6] F. Forbes and N. Peyrard, "Hidden Markov random field model selection criteria based on mean field-like approximations", *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 9, pp. 1089–1101, 2003.

[7] Z. Yu, O. C. Au, R. Zou, W. Yu, and J. Tian, "An adaptative unsupervised approach toward pixel clustering and color image segmentation". *Pattern Recognition*, vol. 43, no. 5, pp. 1889–1906, 2010.

[8] M. Murshed, S. W. Teng, and G. Lu, "Cuboid segmentation for effective image retrieval", Proc. Int. Conf. Digital Image Computing: Techniques and Applications (DICTA), 2017.

[9] J. v.d. Weijer, C. Schmid, and J. Verbeek, "Learning color names from real-world images," Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition, 2007.

Table I. Performance comparison of CSeg'17 and CSeg'18

	# of segs	***F***	***B***	$Q = F - B$
CSeg'17	10	0.557	0.174	0.383
	20	0.628	0.139	0.489
	30	0.662	0.125	0.536
	40	0.674	0.119	0.555
CSeg'18	10	0.623	0.155	0.469
	20	0.705	0.117	0.588
	30	0.745	0.101	0.644
	40	0.766	0.092	0.674
Improvement (%)	10	11.99	-10.90	22.35
	20	12.29	-16.07	20.37
	30	12.56	-19.31	20.01
	40	13.65	-22.93	21.50