Advanced Image Segmentation Techniques for Ambiguous Foreground Images

Tzu-Chieh Lin[a], Yu-Yao Huang[b], I-Fan Lu[c], Hsuan-Yi Ko[d], Jian-Jiun Ding[e,*], Jin-Yu Huang[f], and Ping-Hung Chen[g]

Graduate Institute of Communication Engineering, National Taiwan University

E-mail address: r04942101@ntu.edu.tw [a]; b05901174@ntu.edu.tw [b]; r02942042@ntu.edu.tw [c]; r039420564@ntu.edu.tw [d]; jjding@ntu.edu.tw [e]; r07942085@ntu.edu.tw [f]; r07942055@ntu.edu.tw [g]

Abstract

Image segmentation has been a popular topic in computer vision and image processing for decades. There are lots of applications that rely on the performance of image segmentation such as object detection and image compression, medical image processing. Different kinds of image segmentation algorithm have been developed over the years. In this paper, we propose an image segmentation algorithm based on superpixels, texture, foreground significance estimation and contour/edge information. First, we use mean shift to generate the initial superpixels, then under the constraint of intensity of edge with the color and texture information, the superpixels go through a growing process. Following, a foreground significance estimation is applied to determine whether the edge/texture enhancement should be used or not. Second, with the proposed dissimilarity measure metrics based on color, texture, edge, superpixel size, one can use them as the rules to determine whether two superpixels should be merged. Moreover, contact rate of two adjacent superpixels is applied to prevent over-merging from happening. The algorithm is first designed to segment image with number of regions assigned by the user in prior. However, one can implement it into a fully automatic segmentation algorithm. We report simulation results on the Berkeley segmentation dataset with both implementations. The results show that the proposed method is much better than that of many state-of-the-art.

Keyword: Image segmentation, superpixel, contour detection, saliency detection, computer vision.

1. Introduction

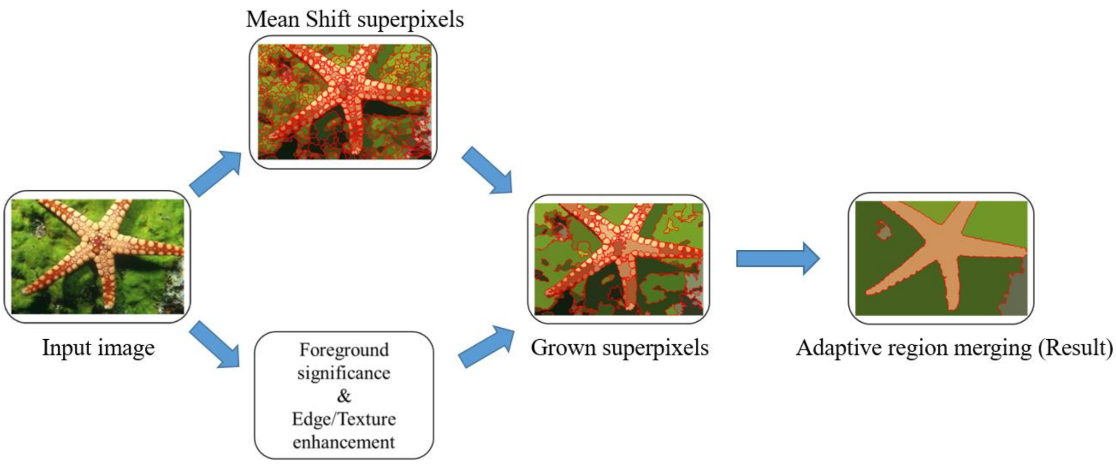


Fig. 1 An overview of our framework

Image segmentation is to segment image into several regions of objects for the purpose that the image can be simplified and analyzed later on. Therefore, image segmentation plays an important role in image processing and computer vision, for example, image segmentation can generate shading mask to improve object detection and recognition.

Recently, lots of different image segmentation algorithms has been proposed. Traditional features such as color, edge and texture have been proved having great performance in image processing, so that they are still widely used during segmentation process. For efficiency, superpixel algorithms like mean shift (MS) (Comaniciu, 2002), simple linear iterative clustering (SLIC) (Achanta, 2012) and entropy rate superpixels (ERS) (Liu, 2011) group pixels with high similarity into several perceptually meaningful small regions. That is, instead of directly utilizing single pixel, using superpixel as the underlying representation in the computation has greatly improved the efficiency, since the number of superpixels is usually smaller than that of total pixel in an image. Furthermore, superpixel segmentation can produce better segmentation result.

In this paper, we propose a superpixel-based image segmentation method that utilize color, edge, texture, and saliency information as feature to merge superpixels. Fig. 1 shows the overview of our framework. We apply adaptive edge and texture enhancement according to the foreground significance estimation. From the experimental results, the proposed method outperforms other state-of-art image segmentation methods in all evaluation metrics.

This paper is organized as follows. A brief introduction of related work about superpixel and some state-of-art image segmentation approaches are presented in Section 2. In Section 3, the adopted techniques. In Section 4, the implementation and techniques of our proposed image segmentation algorithm are described. In Section 5, the experimental results are presented. In Section 6, a conclusion will be given.

2. Related Work

Superpixel algorithm is usually considered to be an over-segmented process, as the segmentation result is focus on grouping pixels with same relatively low-level features, like color and texture. Therefore, superpixel algorithm is often a pre-processing in many image segmentation algorithms. When the image has been processed by a superpixel algorithm, its color, texture and other information will be simplified from pixel-based to region-based which is helpful for reducing the complexity of computation. Mean Shift (Comaniciu, 2002) is an iterative clustering technique that shifts each data point to the local peak on the kernel density estimating function. It is considered to be a nonparametric clustering technique while the number of clustering is no need to be set manually. The most frequently used superpixel algorithm like simple linear iterative clustering (SLIC) (Achanta, 2012) proposed by Achanta *et al*. is a gradient ascent based clustering algorithm that groups pixels using multi-dimensional feature.

In the last few years, spectral segmentation has become a popular way in image segmentation field. It uses the global information embedded in the spectrum of a given image's affinity matrix from multiple over-segmentation. Integrating local grouping cues, In (Kim, 2014) the method of multi-layer spectral segmentation (MLSS) is proposed. It uses a full range affinity model in the spectral segmentation framework to obtain high-quality segmentation results efficiently by using the proposed affinity model. A novel graph-based segmentation framework, called segmentation by aggregating superpixels (SAS), was proposed by (Li, 2012). It uses information from superpixels in multi-layers to segment an image. The basic concept for the grouping rules is based on two simple observations. First, pixels within the same superpixels tend to form a homogenous region. Second, if a pixel and its adjacent pixels are close in feature space, these two pixels are likely to form a homogenous region. Therefore, they use multiple over-segmentation obtained from different superpixel generation method and combine these superpixels to construct the segmentation output.

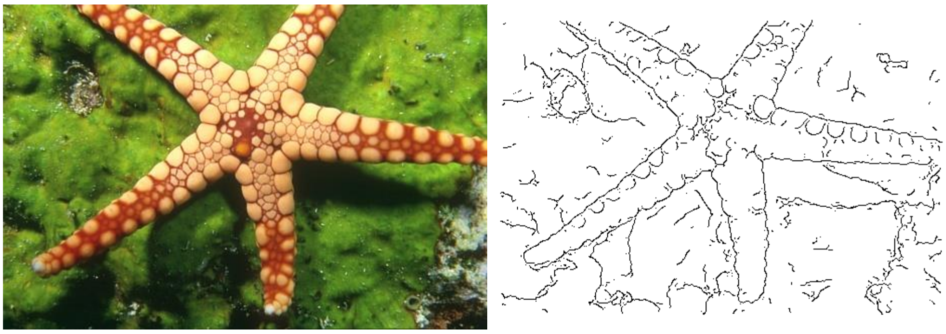
3. Adopted Techniques:

3.1 Mean Shift Superpixel:

The superpixels we adopt are generated by the mean shift method (Comaniciu, 2002). It has the properties for good boundaries maintenance and its segmented result is between over-segmentation and over-merging.

3.2 Saliency Detection:

Saliency detection is to capture the informative regions in an image. It generates an intensity map with saliency value which represent the importance of each pixel. The regions with large saliency value are more likely to be the human interested regions.



Original Image Contour Map

Fig. 2 An example of a contour map generated by structured edge detector.

In our framework, we adopt two kinds of saliency detection proposed in (Zhu, 2014) and (Kim, 2014), respectively. Kim *et al*. propose a saliency detection that utilize high-dimensional color transform (HDCT) to combine global and local salient region detection by random forest algorithm. Instead of one single color space, the HDCT applies multiple color space representation like RGB, CIELAB and HSV to solve the color ambiguities within regions. We can derive the difference of saliency values between two regions R_1 and R_2 from the computed saliency map.

$$dSV(R_1, R_2) = |SV(R_1) - SV(R_2)| \quad (1)$$

where $SV(R_1)$ represents the mean saliency value of region R_1.

3.3 Edge Detection:

In the proposed method, we adopt the edge detection algorithm proposed by (Dollar, 2013) and apply the structured edge (SE) detection to the input image to generate an edge map. This edge map will be further performed with binarization and forms the binary contour map. An example of a generated contour map is shown in Fig. 2. Here we define the *ContourRate* as:

$$ContourRate(i, j) = \frac{\#\,of\ \ pixels\ of\ \ long\ contours\ on\ Bnd(i, j)}{\#\,of\ \ pixels\ of\ \ Bnd(i, j)} \quad (2)$$

where $Bnd(i, j)$ is the boundary between two adjacent superpixel i and j. We further define edge strength to compensate the lack of long contours by:

$$ES(i, j) = \frac{\sum_{p \in Bnd(i,j)} edge\ value\ of\ \ p}{\#\,of\ \ pixels\ of\ \ Bnd(i, j)} \quad (3)$$

where edge value of p denotes the edge response value calculated by structured edge detector of pixel p on the boundary between superpixel i and j.

3.4 Lab Color Histogram:

We uniformly quantize each color channel into 32 levels and then the histogram of each superpixel is calculated in the feature space of 32×32×32 = 32768 bins. We define the similarity measure $\rho(R,Q)$ between two superpixels R and Q as

$$\rho(R,Q) = \sum_{u=1}^{U} \sqrt{\mathrm{Hist}_{\mathrm{R}}^{\mathrm{u}} \cdot \mathrm{Hist}_{\mathrm{Q}}^{\mathrm{u}}} \tag{4}$$

where $Hist_R$ is the normalized histogram of superpixel R, U is the number of bins of quantization level, and u means the u^{th} element in normalized historgram. Therefore, the Bhattacharyya coefficient ρ is defined as the cosine of the unit vectors:

$$\left(\sqrt{\mathrm{Hist}_R^1}....\sqrt{\mathrm{Hist}_R^U}\right)^T \text{ and } \left(\sqrt{\mathrm{Hist}_Q^1}....\sqrt{\mathrm{Hist}_Q^U}\right)^T . \tag{5}$$

3.5 CIEDE2000 Color Difference:

We choose CIELAB color space and CIEDE2000 color differences (Sharma, 2005) to match human perception. Given two superpixels R_1 and R_2, we calculate the mean Lab values $M_{R1} = (L_1, a_1, b_1)$ and $M_{R2} = (L_2, a_2, b_2)$ and determine their difference:

$$de00(R_1, R_2) = \Delta E_{00}(M_{R1}, M_{R2}) \tag{6}$$

where ΔE_{00} is color difference from CIEDE2000.

3.6 Texture Features:

We choose Log-Gabor filter (Field, 1987) to extract the texture features of each superpixel. The Log-Gabor filter has a null DC component and can be constructed with arbitrary bandwidth and the bandwidth can be optimized to produce a filter with minimal spatial extent. In this paper, we utilize the Log-Gabor filter with 2 scales and 4 orientations to produce texture maps. Then, the difference of textures between two adjacent superpixel i and j is determined from:

$$dTex(i,j) = \sqrt{\sum_{k=1}^{8} (T_k(i) - T_k(j))^2} \tag{7}$$

where k denotes the k^{th} combination of scale and orientation.

3.7 Texture Information Enforcement:

We apply the 2D-DCT to each superpixel in grayscale image to extract more texture information. Since the range of 2D-DCT is within a rectangular, we must adjust the shape of superpixels for computation to cope with the irregular shape of superpixels. Therefore, for each superpixel, we pad the minimum bounding box with mean intensity value of superpixel.

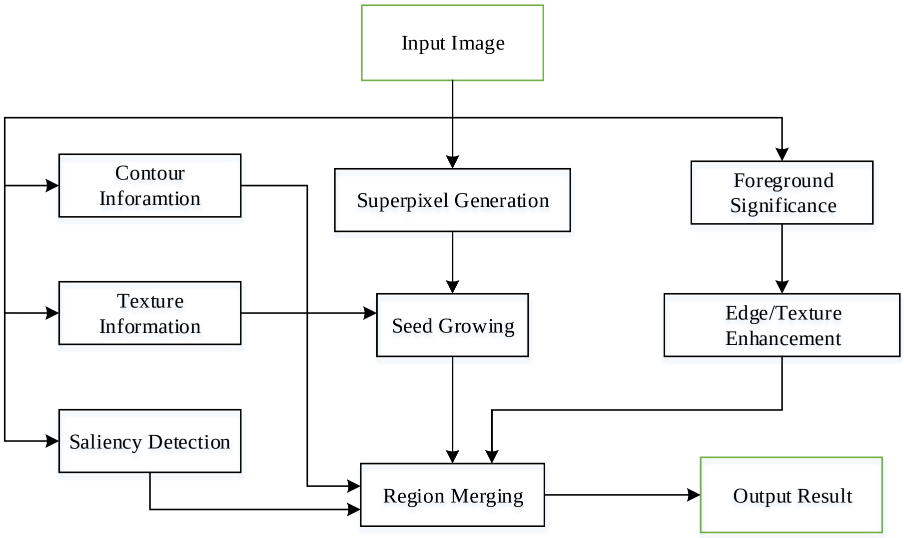


Fig. 3. The block diagram of the proposed method.

The stronger response will occur in middle frequency band with more repeating pattern within regions. Hence, we define the texture strength by measuring the middle band response after we filter the DC term.

$$stTex(R) = \frac{\#\,of\ \ pixels \in midband}{\#\,of\ \ pixels\ of\ \ X_{p,q}} \tag{8}$$

where the *midband* is area on $X_{p,q}$ with DC term filtered. We further capture the texture information using gradient information by measuring the histogram of gradient within each superpixel. Then, we compute the texture from gradient histogram by:

$$hisTex(R) = \frac{\#\,of\ \ pixels \in histogramLevel(6 \sim 30)}{\#\,of\ \ pixels\ in\ R} \tag{9}$$

4. Approach

In this section, we cover the main superpixel merging procedure of the proposed method. Based on various similarity measurement in Section 3, we use them to calculate the score of two superpixels to determine whether they should be merged. The block diagram of the proposed method is shown in Fig. 3.

4.1. Superpixel Growing:

With the similarity measure defined in (4), we use such constraint to determine whether to merge superpixel R with its adjacent superpixel Q if they are the closet in the histogram measurement space to each other. That is, we can represent the neighbor of superpixel Q as $\{S_i^Q\}_{i=1,2,\ldots,q}$. Apparently, R is within $\{S_i^Q\}_{i=1,2,\ldots,q}$ since they are adjacent, therefore, Q is also a member of $\{S_i^R\}_{i=1,2,...,r}$. Finally, the merging between superpixel R and Q will happen if the following conditions are fulfilled:

$$\rho(R,Q) = \max_{i=1,2,...r} \rho(R, S_i^R) \quad \text{and} \quad \rho(R,Q) = \max_{i=1,2,...q} \rho(Q, S_i^Q) \tag{10}$$

Although we define a strict constraint to merge two superpixels, there are some situations that could cause over-merging. Therefore, we set up some rules to further prevent such situation from happening. That is, we use $dTex(R,Q)$ and $ContourRate(R,Q)$ as threshold to reinforce the constraint.

After the initial merging of superpixels, we consider the superpixels that have merging times higher than or equal to three as the initial seeds and perform seed growing process using the distance defined as:

$$Dist(i,j) = a \cdot dTex(i,j) + b \cdot (1 - \rho(i,j)) \tag{11}$$

where i is the initial seeds, and j is its adjacent superpixels. Therefore, the seed growing process will merge the seeds with its neighbor that is closest to it in the $Dist(i,j)$ measurement. In this paper, we use (a,b) = (3,0.5) to perform the seed growing. However, we use $ContourRate(i,j)$ as the additional limitation of the growing process.

4.2. Adaptive Region Merging:

In this stage, we consider the information from the whole image instead of local regions and its adjacent regions. Therefore, we construct an N-by-N dissimilarity matrix for the images with N regions and merge regions according to the dissimilarity matrix.

We generate multi-scale contour maps (*CR25, CR80, and CR30*) using different combination of (*contourLength, threshold)* = (25,0.1), (80,0.1) and (30, 0.15) respectively. Then utilize multi-scale contour maps to produce *ContourRate* in different degree.

Also, we take the size of regions into account to preserve the main object. Therefore, the tendency to merge regions is adaptive to the size of two adjacent regions. Given two adjacent regions R_1 and R_2, the constraint is:

$$minArea = \min(area(R_1), area(R_2)) \tag{12}$$

We propose three ways of calculating dissimilarity:

$$score1(R_1,R_2) = \left(a \cdot ES(R_1,R_2) + b \cdot CR25(R_1,R_2) + c \cdot CR80(R_1,R_2)\right) \times \min\left(\frac{\text{minArea}}{W}, K\right) - \rho(R_1,R_2) + 2dTex(R_1,R_2)$$

(13)

$$\begin{aligned} score2(R_1,R_2) = {} & (ES(R_1,R_2) + CR25(R_1,R_2)) \times \frac{\text{minArea}}{W} + CR30(R_1,R_2) \\ & -(1-\beta) \cdot \rho(R_1,R_2) + \beta \cdot de00(R_1,R_2) + \alpha \cdot dTex(R_1,R_2) \end{aligned}$$

(1

4)

$$score3(R_1,R_2) = a \cdot ES(R_1,R_2) + b \cdot CR25(R_1,R_2) + c \cdot CR80(R_1,R_2) + d \cdot 30(R_1,R_2)$$
$$- \max\left(\frac{W}{\text{minArea}}, K\right) - \rho(R_1,R_2) + 2dTex(R_1,R_2) \tag{15}$$

where $\alpha = \max(2, 2\times W/minArea)$, (a, b, c, d) is the weighting parameter set, $dTex(R_1, R_2)$, $1-\rho(R_1,R_2)$, and de00(R_1, R_2) are defined in Section 3, whereas W and K are constants parameter to control the small region weighting.

Due to the irregular shape of superpixels, two superpixel could connected to each other by only a few pixels and then considered as neighbors. Therefore, the information from $Bnd(R_1,R_2)$ is insufficient to perform segmentation wells. In that case, we define the percentage of contacted boundary over the length of their own borders.

$$ContactRate(R_1,R_2) = \frac{\#\ of\ \ pixels\ of\ \ Bnd(R_1,R_2)}{\min(BL(R_1), BL(R_2))} \tag{16}$$

where $BL(R_1)$ indicates the length of border of region R_1. Two regions will not be merged if the *ContactRate* is below threshold. More on that, the threshold is adaptive to the number of remaining segments N, for example, large threshold for large N. In addition to maintain main objects, we add dSV to the constraint when the number of remaining regions is smaller than 44. Hence, we merge regions with small dSV.

5. Simulations

5.1 Parameter Setting:

We use Mean Shift to generate the initial over-segmentation to obtain the superpixels with the parameter $(h_s, h_r, M) = (5, 7, 100)$ where (h_s, h_r) is the spatial and range bandwidth respectively, and M is the minimum size of resulting regions.

We also develop another automatic version of segmentation that does not require user input the number of segments. The original version stops the merging process while the N reach the defined number of segment. Therefore, we remove stopping criterion of defined number of segment, the merging process will now stop when all the remaining superpixels fail to satisfy the constraint of merging.

5.2 Evaluation:

We perform our experiments on the Berkeley Segmentation Data Set 300 (BSDS300), which consist of 300 color images with at least 4 annotated ground truth images for each image. Furthermore, we use COCO dataset to test the automatic version of our proposed method. As for evaluation metrics in BSDS 300, we adopt Probabilistic Rand Index (PRI), Variation of Information (VoI), Global Consistency Error (GCE), and Boundary Displacement Error (BDE) to perform scoring of our segmentation results. Among all the evaluation metrics, the score is better if PRI is larger while other three are smaller.

Table 1: Performance of the proposed method compares to other methods over the BSDS 300.

Method	PRI	VoI	GCE	BDE
Ncut	0.7242	2.9061	0.2232	17.15
JSEG	0.7756	2.3217	0.1989	14.40
MNcut	0.7559	2.4701	0.1925	15.10
NTP	0.7521	2.4954	0.2373	16.30
SDTV	0.7758	1.8165	0.1768	16.24
TBES	0.80	1.76	N/A	N/A
UCM	0.81	1.68	N/A	N/A
MLSS	0.8146	1.8545	0.1809	12.21
Proposed (auto)	0.8178	1.5895	0.1647	11.22
SAS	0.8319	1.6849	0.1779	11.29
(Hsu, 2013)	0.8474	1.5542	0.1834	11.15
TBES (using W_{HP})	0.8495	1.6260	0.1785	12.3034
GL-graph	0.8384	1.8012	0.1934	10.6633
SCS	0.8414	1.6573	0.1795	10.8783
Proposed	**0.8712**	**1.3833**	**0.1427**	**8.9312**

We compare our method with the other algorithms like Ncut (Shi 2000), JSEG (Deng, 2001), NTP (Wang, 2008), MNcut (Cour, 2009), saliency driven total variation (SDTV) (Donoser, 2009), TBES (Rao, 2009), UCM (Arbelaez, 2011), MLSS (Kim, 2010), and SAS (Li, 2012), efficient image segmentation algorithm using SLIC superpixels and boundary-focused region merging (Hsu, 2013), superpixel-based image segmentation by incorporating color covariance matrix manifolds (Gu, 2014), global/local affinity graph for image segmentation (GL-graph) (Wang, 2015), and SCS (Yang, 2016). Table 1 shows the scores. Furthermore, the score of a fully automatic version of proposed(auto) algorithm that do not need user input the number of regions is also listed. Apparently our algorithm outperforms all the other methods. Even the automatic version has outperform many methods.

The visual comparisons are given in Figs. 4 and 5.

6. Conclusion

In this paper, we proposed a novel superpixel-based image segmentation that consider a variety of features like edge/contour, texture, color. Instead of using local features directly, we proposed different kind of similarity measure that can greatly illustrate the relation between two adjacent superpixels. Experimental results on the Berkeley Segmentation Data Set have shown that our proposed algorithm outperforms the state-of-the-art methods in all evaluation metrics.

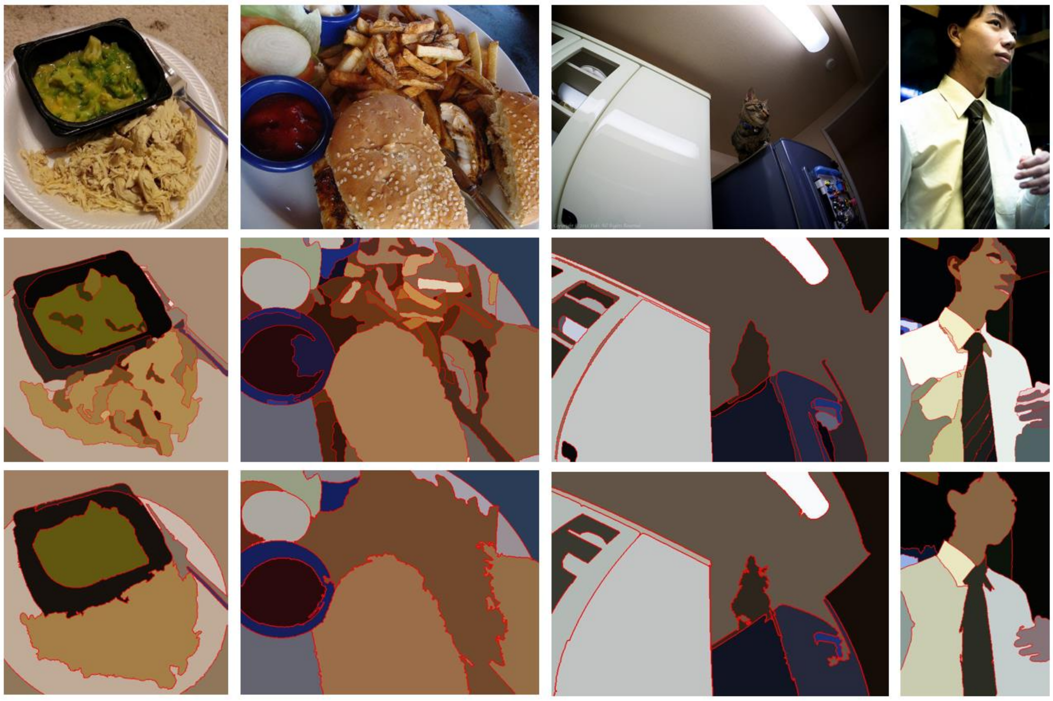


Fig 4. Visual comparison from COCO dataset. Top row: original images. Second row: results from UCM. Third row: results from the proposed method.

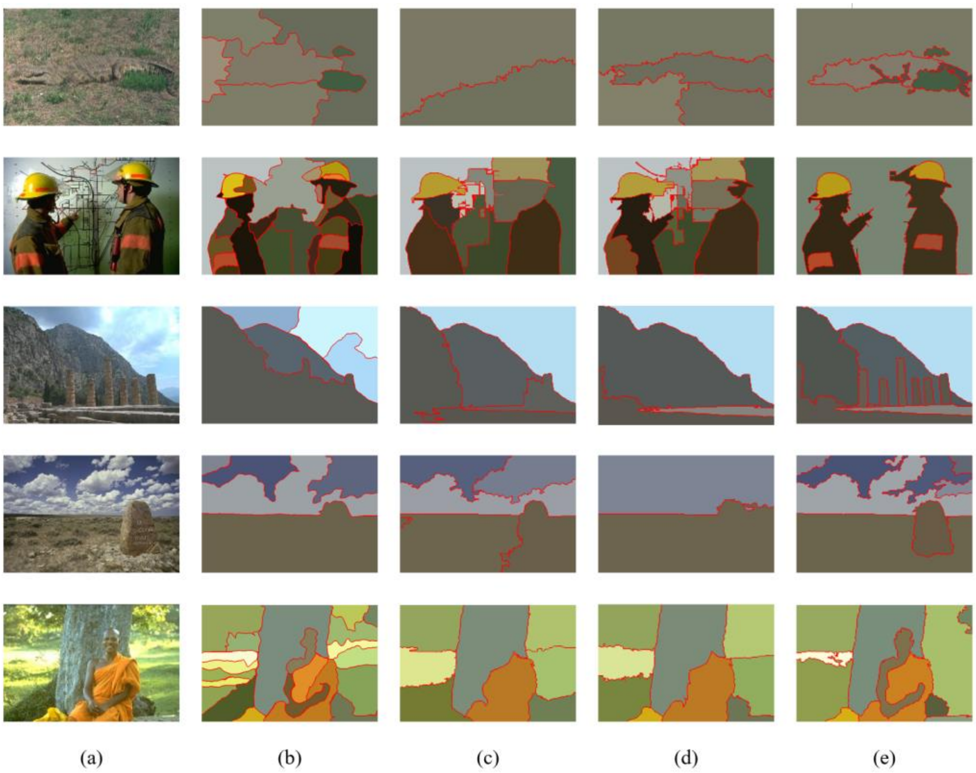


Fig 5. Visual comparison on BSDS300 (a) Input images. (b) TBES. (c) MLSS (d) SAS. (e) Ours

6. Reference

Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., & Süsstrunk, S.(2012) SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. Pattern Analysis and Machine Intelligence, 34*(11), 2274 - 2282.

Arbelaez, P., Maire, M., Fowlkes, C., & Malik, J.(2011) Contour detection and hierarchical image segmentation, *IEEE Trans. Pattern Analysis and Machine Intelligence, 33*(5), pp. 898-916.

Comaniciu, D., & Meer, P.(2002) Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Pattern Analysis and Machine Intelligence*, *24*(5), 603-619.

Cour, T., Benezit, F., & Shi, J.(2005) Spectral segmentation with multiscale graph decomposition, in *IEEE Conf. Computer Vision and Pattern Recognition*, *2*, pp. 1124-1131.

Deng, Y., & Manjunath, B.S.(2001) Unsupervised segmentation of color-texture regions in images and video, *IEEE Trans. Pattern Analysis and Machine Intelligence, 23*(8), pp. 800-810.

Dollar, P., & Zitnick, C.L.(2013) Structured forests for fast edge detection, in *ICCV*, pp. 1841–1848.

Donoser, M., Urschler, M., Hirzer M., & Bischof, H.(2009) Saliency driven total variation segmentation, in *IEEE Int. Conf. Computer Vision*, pp. 817-824.

Field, D.J.(1987) Relations between the statistics of natural images and the response properties of cortical cells, *J. Opt. Soc. Am. A*, *4*(12), 2379-2394.

Gu,, X., Deng, J.D., & Purvis, M.K.(2014) Improving superpixel-based image segmentation by incorporating color covariance matrix manifolds, in *ICIP*, pp. 4403-4406.

Hsu, C.Y., & Ding, J.J.(2013) Efficient image segmentation algorithm using SLIC superpixels and boundary-focused region merging, in *ICICS*, pp. 1-5,.

Kim, J., Han, D., Tai Y.W., & Kim, J.(2014) Salient region detection via high-dimensional color transform, in *IEEE Conf. Computer Vision and Pattern Recognition*, pp. 883-890.

Kim, T., & Lee, K.(2010) Learning full pairwise affinities for spectral segmentation, in *IEEE Conf. Computer Vision and Pattern Recognition*, pp. 2101-2108.

Li, Z., Wu, X.M., & Chang, S.F.(2012) Segmentation using superpixels: A bipartite graph partitioning approach, in *IEEE Conf. Computer Vision and Pattern Recognition*, pp. 789-796.

Liu, Y.M., Tuzel, O., Ramalingam, S., & Chellappa, R.(2011) Entropy rate superpixel segmentation, in *IEEE Conf. Computer Vision and Pattern Recognition*, pp. 2097-2104.

Rao, S. R., Mobahi, H., Yang, A.Y., Sastry, S.S., & Ma, Y.(2009) Natural image segmentation with adaptive texture and boundary encoding, in *ACCV*, pp. 135-146.

Sharma, G., Wu, W., & Dalal, E.(2005) The CIEDE2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations, *Color Research & Application*, *30*(1), 21-30.

Shi, J., & Malik, J.(2000) Normalized cuts and image segmentation, *IEEE Trans. Pattern Analysis and Machine Intelligence, 22*(8), 888-905.

Wang, J., Jia, Y., Hua, X.S., Zhang, C., & Quan, L.(2008) Normalized tree partitioning for image segmentation," in *IEEE Conf. Computer Vision and Pattern Recognition*, pp. 1-8.

Wang, X., Tang, Y., Masnou, S., & Chen, L.(2015) A global/local affinity graph for image segmentation, *IEEE Trans. Image Processing, 24*(4), pp. 1399-1411.

Yang, Y., Wang, Y., & Xue, X.(2016) A novel spectral clustering method with superpixels for image segmentation, *Optik-International Journal for Light and Electron Optics, 127*(1), pp. 161-167.

Zhu, W., Liang, S., Wei, Y., & Sun, J.(2014) Saliency optimization from robust background detection, in *IEEE Conf. Computer Vision and Pattern Recognition*, pp. 2814-2821.