A Fast Algorithm for Content-aware Saliency Detection and Stylized Rendering

Yongwei Miao , Ruifeng Han
College of Computer Science and Technology
Zhejiang University of Technology
Hangzhou, China

Huahao Shou
College of Computer Science and Technology
Zhejiang University of Technology
Hangzhou, China

Abstract—**Image saliency plays an important role in image processing and recognizing tasks. Owing to color difference computation between the patch centered at the current pixel and the patches centered at its neighboring pixels, a fast algorithm for multi-scale content-aware saliency detection is presented in this paper. Furthermore, the histogram information of the image is also incorporated into our saliency detection to eliminate the loss of accuracy introduced in our local dissimilarity comparison. Due to our content-aware saliency estimation, a saliency guided stylized rendering scheme is given by combing the saliency guided bilateral filtering and saliency guided contour detection techniques.**

Keywords- Content-aware, Image saliency, Bilateral filtering, Edge detection, Stylized rendering

I. INTRODUCTION

Human possess a strong ability of visual perception and pattern recognition. According to theory of biological vision, the content-aware visual saliency can effectively guided the viewer's visual attention to the salient regions of 2D images [1]. The saliency map of an image is a scalar field defined for each image pixel and is usually represented as a gray image. The saliency measure of image pixel can reflect its relative saliency triggered by its difference compared with its surrounding pixels. The saliency map can benefit for many applications of image processing, such as image resizing and retargeting, summarization and stylized rendering [2, 3, 4, 5].

Recently, combing the bottom-up and top-down scheme, Goferman et al. [6] proposed a context-aware saliency computation approach by employing the color and position information of each image pixel, which can extract the salient objects and also reserve their surrounding regions. However, they calculate the saliency of each pixel by considering the patch dissimilarity of K most similar patches, which leads to the computation of patch dissimilarity between each pixel patch and all other patches in the whole image. So, their scheme for saliency detection always leads to high time complexity.

Owing to the computation of the dissimilarity degree between the patch centered at the current pixel and the patches centered at its local neighboring pixels, a fast algorithm for content-aware saliency detection is presented in this paper. To alleviate the error of saliency detection, we introduce the histogram information of the image into our saliency detection scheme to recover the lost global information. Furthermore, by incorporating the saliency map into the traditional image processing tasks, we also proposed a saliency guided stylized rendering scheme by combing the saliency guided bilateral filtering and saliency guided contour detection techniques.

II. RELATED WORK

Image saliency can reflect the visual important of each pixel. Due to different visual attention mechanisms, the saliency detection can be classified as the data-driven bottom-up scheme and the task-driven top-down scheme [7, 8, 9]. Owing to the theoretical framework of controlling the visual attention proposed by Koch and Ullman [10], Itti et al. [7, 8] proposed a bottom-up scheme for computing saliency of 2D images. It firstly extracts the major features from the input image in terms of brightness, color, etc and then calculates the attention map of various feature domains by using their center-surround operation. Finally, the saliency map of the underlying image can obtain by combining these attention maps.

Meanwhile, with the development of understanding inherent mechanism of visual information processing, the task-driven top-down scheme for saliency detection also attractive the attention of researchers. Sun and Fisher [11] presented a target-based attention computational model, which combine the visual attention theory with Itti's visual saliency computational model. Peters and Itti [12] improved the data-driven bottom-up visual attention model by employing the task-driven top-down attention mechanism on spatial information.

III. A FAST ALGORITHM FOR CONTENT-AWARE SALIENCY DETECTION

In general, the image pixels in the significant regions should get high saliency values, whilst those pixels in the unimportant areas should get lower saliency values. Various image features may contribute to the saliency measure of pixels [7, 8], such as brightness, color, etc. Whether a pixel is salient or not depends on the degree of dissimilarity between the pixel's patch and other patches, that is, the greater dissimilarity generates higher saliency value and vice versa. This dissimilarity degree can be defined by the "patch distance", and the saliency value of pixel can be determined by the dissimilarity degree. Our fast algorithm for content-aware saliency detection consists of the following steps.

A. Patch distance computation in a local scheme

For two image pixels i and j, we first define the "patch distance" between patch p_i centered at i and patch p_j centered at j as follows:

$$d(p_i, p_j) = \frac{d_{color}(p_i, p_j)}{1 + c \cdot d_{position}(p_i, p_j)} \quad (1)$$

Where c is a constant and set to $c = 3.0$ in our implementation. The term $d_{color}(p_i, p_j)$ means the normalized Euclidean distance between two patches p_i and p_j in CIE L*a*b color space, which is calculated by a quadratic sum of the color differences between the corresponding pixels of two patches. The term $d_{position}(p_i, p_j)$ is the Euclidean distance between the positions of patches p_i and p_j, normalized by the larger image dimension.

B. Single-scale saliency estimation by employing the histogram information

For a patch p_i of single-scale $r(7*7)$, we only search for the K smallest distance patches $\{q_k\}_{k=1}^{K}$ (due to the distance computation by Eq. (1)) from its local neighboring pixels (for example 15*15 rectangular neighborhood centered at pixel i). We set $K = 60$ in our implementation. So, we can calculate the single-scale saliency of pixel i at scale r as follows:

$$S_i^r = 1 - \exp\{-\frac{1}{K}\sum_{k=1}^{K} d(p_i^r, q_k^r)\} \quad (2)$$

In [6], the K most similar patches $\{q_k\}_{k=1}^{K}$ are selected from the whole image, which will lead to the distance computation $d(p_i, p_j)$ between patch p_i and all of other patches p_j in the whole image. It will lead to large memory requirement and high time complexity. Here, for the sake of computing the patch dissimilarity effectively, we limit the search range of K most similar patches into the local neighboring of patch p_i. However, the saliency measure of image pixels by this confining scheme will only consider its relative importance by comparing with its local neighboring pixels, which will obviously lead to the loss of accuracy of saliency estimation. This issue will be overcome in the next step by employing the histogram information of the image.

In the field of image processing, the image histogram can record the number of occurrences of each color. Unfortunately, the color statistics is difficult to describe in the CIE L*a*b space, so we use the RGB color space to calculate the image histogram information. That is, a three-dimensional array $RGB[256][256][256]$ is adopted to record the number of occurrences of each color. Such as, if the red pixels occur in the image for 100 times, we set the value $RGB[256][0][0] = 100$. Let RGB_i be the number of occurrences of the color of image pixel i. Then, we can improve the saliency estimation as follow:

$$S_i^r = \begin{cases} \frac{S_i^r}{RGB_i}, & RGB_i > 0 \\ S_i^r, & RGB_i = 0 \end{cases} \quad (3)$$

Here, the value of RGB_i is normalized [6, 8] thus restraining the saliency value of those background pixels and reducing the inaccuracy of saliency map.

C. Saliency map by combing multi-scale saliency estimation

For each image pixel i, we employ the multi-scale patches of $r1(7*7)$, $r2(5*5)$ and $r3(3*3)$ for the similarity comparison with the same scale patches centered in the neighborhood pixel j. So, we get the multi-scale distance computation:

$$\{d(p_i^{r1}, p_j^{r1}), d(p_i^{r2}, p_j^{r2}), d(p_i^{r3}, p_j^{r3}), \quad for all\ j\}$$

Then we can pick the K smallest distance $\{d(p_i^r, p_k^r), k = 1, 2, \cdots, K\}$ from all of these distance values, and the saliency of pixel i can be determined by combining the multi-scale saliency:

$$S_i = 1 - \exp\{-\frac{1}{K}\sum_{k=1}^{K} d(p_i^r, p_k^r)\} \quad (4)$$

Finally, the normalized saliency value of pixel i can then be computed as:

$$S_i = 255 \times \frac{S_i - v_{\min}}{v_{\max} - v_{\min}} \quad (5)$$

Here, the $v_{\max}$, $v_{\min}$ mean the maximum and minimum value of the saliency map respectively.

IV. EXPERIMANTAL RESULTS AND DISCUSSION

A. Our experimental results of content-aware saliency detection

We have implemented all of the proposed algorithms for multi-scale saliency detection. For the sake of convenient, we call the multi-scale scheme for globally searching K most similar patches from the whole image as "global comparison", whist denote our multi-scale scheme for locally searching K most similar patches from the local neighborhood as "local comparison".

Figure1 shows the multi-scale (7*7, 5*5, 3*3) saliency map comparison between "global comparison" and our "local comparison". Here, our "local comparison" scheme has employed the histogram information, and the neighborhood for each pixel is a 15*15 rectangular area. Figure 2 compares the multi-scale (7*7, 5*5, 3*3) saliency map by our "local comparison" scheme with or without employing the histogram information.

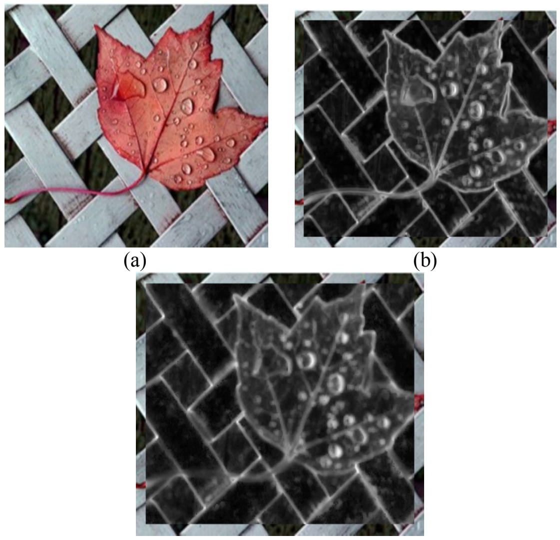

Figure 1. Multi-scale (7*7,5*5,3*3) saliency map comparison between "global comparison" and our "local comparison". (a): Input image; (b): The multi-scale saliency map computed by "global comparison" scheme; (c): The multi-scale saliency map computed by our "local comparison" scheme (15*15 rectangular neighborhood for each pixel).

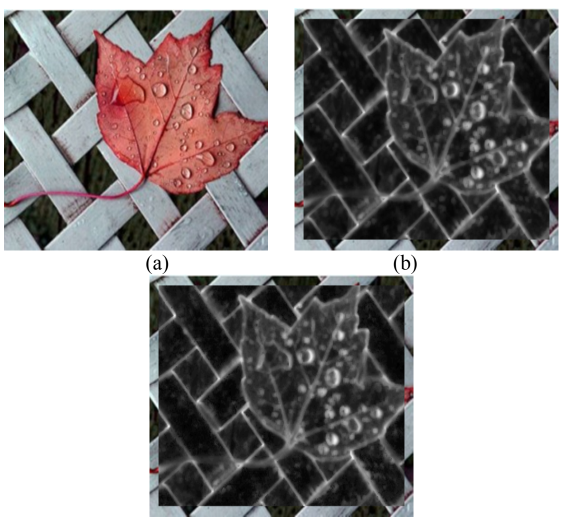

Figure 2. Multi-scale (7*7,5*5,3*3)saliency map comparison by our "local comparison" scheme (15*15 rectangular neighborhood for each pixel) with or without employing the histogram information. (a): Input image; (b): The multi-scale saliency map computed by our "local comparison" scheme without employing the histogram information; (c) The multi-scale saliency map computed by our "local comparison" scheme with employing the histogram information.

In Figure 3, we list some saliency detection results to compare between the "global comparison" scheme and our "local comparison" scheme with employing the histogram information. It shows the effectiveness of the saliency estimation by our "local comparison" with employing the histogram information. For example, the saliency of the clothes region in the 2nd Row should be low due to the high occurrence frequency of the color, which can be found in the saliency map by our "local comparison" scheme.

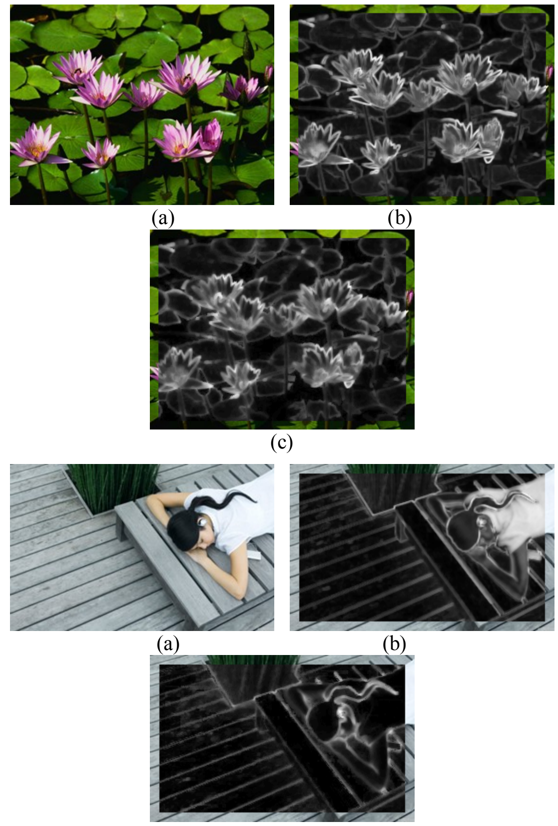

Figure 3. Multi-scale (7*7,5*5,3*3) saliency map comparison between "global comparison" and our "local comparison". Column (a): Input image; Column (b): The multi-scale saliency map computed by "global comparison" scheme; Column (c): The multi-scale saliency map computed by our "local comparison" scheme (15*15 rectangular neighborhood for each pixel) with employing the histogram information.

B. Computational efficiency of our local comparison scheme

The proposed algorithms for multi-scale saliency detection have been implemented on a PC with Dual-Core CPU of 2.1 GHz, and 1024M memory. We select 3 pictures with different size and compare the multi-scale saliency estimation speed by two schemes --- the "global comparison" and our "local comparison". In our implementation, we set $K = 60$ uniformly. For the "global comparison" scheme, we don't use all the other patches when computing the saliency of pixel i, but only use the first pixel in every three pixels in every row and column (It is similar as the "patches with 50% overlap" way which is adopted in [6]). For our "local comparison" scheme, we take the neighboring pixels as the 15*15 rectangular area centered at each pixel.

TABLE I. RUN TIMES (IN SECOND) COMPARISON FOR SALIENCY DETECTION

Input Image Size	292*222 (Figure1)	310*234 (Figure3, Row 1)	278*175 (Figure3, Row 2)
The Global Comparison	2100 s	2375 s	840 s
Our Local Comparison	96 s	104 s	85 s

V. SALIENCY GUIDED STYLIZED RENDERING

Here, due to our content-aware saliency estimation, a saliency guided stylized rendering scheme is given by combing the saliency guided bilateral filtering and the saliency guided contour detection techniques. In our saliency guided techniques, owing to different saliency measure of image pixels, we treat the salient regions and non-salient regions differently, which can achieve better processing results if comparing with the traditional methods.

In our saliency guided bilateral filtering, we incorporate the estimated saliency measure into the traditional bilateral filtering [13] to make the non-significant regions blurred whist keep the important regions of the input image. That is, the following range weight function is employed in our bilateral filtering operation:

$$w(\hat{x},x,\sigma_r)=(1-u(\hat{x}))e^{-\frac{1}{2}(\frac{\|f(\hat{x})-f(x)\|}{\sigma_r})^2}$$

Here $u(\hat{x})$ denotes the normalized saliency value of image pixel $\hat{x}$. The bilateral filtering can filter the input image by weighted average of combining the positional and color difference information of neighboring pixels as following:

$$H(\hat{x},\sigma_d,\sigma_r)=\frac{\int e^{-\frac{1}{2}(\frac{\|\hat{x}-x\|}{\sigma_d})^2} w(\hat{x},x,\sigma_r)f(x)dx}{\int e^{-\frac{1}{2}(\frac{\|\hat{x}-x\|}{\sigma_d})^2} w(\hat{x},x,\sigma_r)dx}$$

Here $f(x)$ means the color of neighboring pixel x. The parameter σ_d and σ_r mean the "blur radius" of pixel distance and pixel color respectively.

In our saliency guided contour detection, we incorporate the estimated saliency measure into the traditional DoG contour detection [5] to improve the contour extraction results. By eliminating some unimportant contours in contour detection, we take the following DoG function.

$$D(\hat{x},\sigma_e,\tau,\varphi_e)=\begin{cases}1, & if\,(S_{\sigma_e}-\tau\cdot S_{\sigma_r})>0\ or\ u(\hat{x})<X\\ 1+\tanh(\varphi_e\cdot(S_{\sigma_e}-\tau\cdot S_{\sigma_r})) & else\end{cases}$$

Here, $S(\hat{x},\sigma_e)=\frac{1}{2\pi\sigma_e^2}\int f(x)e^{-\frac{1}{2}(\frac{\|\hat{x}-x\|}{\sigma_e})^2}dx$, and $S_{\sigma_e}=S(\hat{x},\sigma_e)$, $S_{\sigma_r}=S(\hat{x},\sqrt{1.6}\cdot\sigma_e)$. Meanwhile, $u(\hat{x})$ denotes the normalized saliency value of image pixel $\hat{x}$, and threshold level is selected as $X=0.5$. That is, we eliminate the non-salient edges which the saliency value of its pixel is lower than 0.5. In the Gauss filter function $S(\hat{x},\sigma_e)$, the parameter σ_e decides the special scale for edge detection. The larger the value σ_e is, the coarser the edges that are detected. The threshold level τ determines the sensitivity of the edge detector. For small values of τ, less noise is detected, but real edges become less prominent. It is selected as $\tau=0.98$ in our experiments. The fall off parameter φ_e determines the sharpness of edge representation, we set $\varphi_e=0.75$ in our implementation.

Finally, the stylized rendering result of an input image can be easily obtained by adding our saliency guided contour extraction result on the saliency guided bilateral filtering result. In our implementation, we always perform 2 bilateral filtering iterations before our saliency guided edge detection procedure. Figure4 shows the saliency guided stylized rendering result for the input image.

VI. CONCLUSION AND FUTURE WORK

By comparing the dissimilarity degree between the patches centered at the current pixel and centered at its local neighboring pixels, a fast algorithm for content-aware saliency detection is presented in this paper. To alleviate the error of saliency detection, the histogram information of the image is combined into our saliency detection scheme to recover the lost global information. A saliency guided stylized rendering scheme is also presented by combing the saliency guided bilateral filtering and saliency guided contour detection techniques.

In the future, we will consider how other image processing tasks can benefit from our content-aware saliency detection, such as image retargeting, image summarization and image thumbnailing, etc.

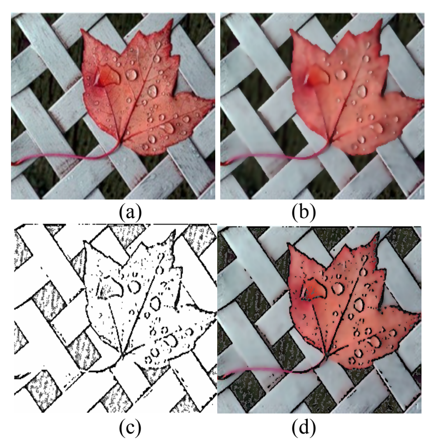

Figure 4. The stylized rendering result for the input image. (a) The input image; (b)The filtering result by our saliency guided bilateral filtering technique; (c)The extracted contour by our saliency guided contour detection technique; (d)Final stylized rendering result.

REFERENCES

[1] Shen, L. Zhang, J., Li, X. Image retrieval and compressed domain processing [M]. Beijing: Posts & Telecom Press, 2008, pp. 102-114.

[2] Avidan, S., Shamir, A. Seam carving for content-aware image resizing [J]. ACM Trans. on Graphics, 2007, 26(3): 20-32.

[3] Achanta, R., Susstrunk, S. Saliency detection for content-aware image resizing [C]. IEEE Intl. Conf. on Image Processing, 2009, pp. 1005-1008.

[4] Liu, Z., Yan, H., Shen, L., Ngan, K.N., Zhang, Z. Adaptive image retargeting using saliency-based continuous seam carving [J]. Optical Engineering, 2010, 49(1): 017002.1-10.

[5] Winnemoller, H., Olsen, S.C., Gooch, B. Real-time video abstraction [J]. ACM Trans. on Graphics, 2006, 25(3):1221-1226.

[6] Goferman,S., Zelnik-Manor,L., Tal, A. Context-aware saliency detection [C]. Proc. IEEE Conf. Computer Vison and Pattern Recognition (CVPR), 2010, pp. 2376-2383.

[7] Itti,L., Koch,C., Niebur,E. A model of saliency-based visual attention for rapid scene analysis [J]. IEEE Trans. on Pattern Analysis and Machine Intelligence, 1998, 20(11): 1254–1259.

[8] Itti,L.,Koch,C. Computational modeling of visual attention [J]. Nature Reviews Neuroscience, 2001, 2(3): 194-203.

[9] Luo, S., et al. The Perception Computing of Visual Information [M]. Beijing: Science Press, 2010, pp. 181-201.