Automatic Image Thumbnailing Based on Fast Visual Saliency Detection

Maiko M. I. Lie[1], Hugo Vieira Neto[1],
Gustavo B. Borba[2], Humberto R. Gamba[1]
Graduate Program in Electrical and Computer Engineering[1]
Graduate Program in Biomedical Engineering[2]
Federal University of Technology – Paraná
Curitiba, Brazil
minian.lie@gmail.com, {hvieir, gustavobborba, humberto}@utfpr.edu.br

ABSTRACT

Image retargeting has seen many applications in areas such as content adaptation for small displays and thumbnailing for image database browsing. Most retargeting methods, however, are too expensive computationally to achieve fast performance on common desktop systems. This work addresses the problem of fast automatic thumbnailing for image browsing. A simple approach of automatic thresholding saliency maps and cropping using bounding box extraction is presented. Eight of the fastest saliency detectors in the literature and three automatic thresholding methods are assessed using precision, recall, F-score and execution time on the MSRA1K dataset. The results show that the approach is computationally efficient and adequate for fast automatic image thumbnailing. In particular, saliency detection with difference to random color samples (RS) thresholded by Rosin's method achieved the best trade-off between execution time and F-score.

Keywords

Thumbnailing; Visual attention; Saliency detection

1. INTRODUCTION

Adaptation of image content to fit certain size restrictions, also known as *image retargeting*, has been used in applications such as image/video viewing on small screens [4], thumbnailing for image database browsing [17] and selective focus of operator attention in surveillance videos [7]. Although being simple and fast, retargetting using uniform resizing is usually not effective, as it does not consider the different importance of the content in each and every region of the image – a very significant aspect, as information loss or distortion is inevitable in this process and, in most cases, integrity preservation of important content is desirable. For this reason, image retargeting algorithms compute an *importance map*, which indicates the importance of each location of the image, in order to preserve their contents accordingly.

Several strategies have been adopted in importance map computation, most notably visual saliency, face detection and text detection [4]. In this work, only visual saliency is considered, as it is based exclusively on low-level characteristics and consequently adequate for general images – unlike face and text detection which are application specific. From the importance map, many spatial manipulations may be performed for retargeting. Methods such as local warping [11] and seam carving [2] resize by preserving important regions while distorting or completely removing the remaining regions. Because unimportant regions can occur in any location, these approaches might alter the relationship between objects in the image and compromise scene comprehension. In applications in which these distortions are undesirable, a combination of rescaling and cropping can be employed.

Many, if not most, image retargetting methods are computationally expensive – some take several seconds [11] to process a single image. Considering this, this work addresses the problem of fast automatic image cropping for thumbnailing. Since image browsers must show several thumbnails at a time, fast mechanisms for their computation are needed. In this work, importance maps are computed using fast saliency detection, followed by automatic thresholding. The importance maps are used for retargeting based on cropping, for simplicity and speed. Eight saliency detectors among the fastest in the literature, as well as three automatic thresholding methods are assessed for this task. Quantitative assessment is made in terms of precision, recall, F-score and execution time on the MSRA1K dataset.

2. RELATED WORK

Chen and colleagues [4] proposed an image retargeting method for visualization in small displays. Their method integrates both bottom-up (i.e. color, intensity and orientation contrasts) and top-down (i.e. face and text detection) visual attention. Suh and colleagues [16] assessed the effectiveness of automatic thumbnail cropping through user interaction experiments on recognition and visual search tasks, finding strong evidence supporting the effectiveness of thumbnails based on visually salient regions.

Image retargetting was also explored in the context of video content, for instance, in surveillance applications [7], in which multiple cropping windows are desired, as well as their smooth trajectory.

WebMedia '16, November 08-11, 2016, Teresina, PI, Brazil
 ISBN 978-1-4503-4512-5/16/11…$15.00
DOI: http://dx.doi.org/10.1145/2976796.2988190

Marchesotti and colleagues [12] proposed a saliency detection framework based of visual similarity applied to the problem of image thumbnailing. Their method has two stages: saliency detection and thumbnail extraction. The former is formulated as a co-saliency model based on visually similar images from a dataset, and is shown to outperform other three state-of-the-art methods in precision, recall and F-score. The latter is formulated as a segmentation method (Grab-Cut) initialized with the saliency map from the first stage. The authors state that this stage overcomes a drawback of the saliency detectors assessed, i.e. these do not account for the contours of the salient objects. We show that this is not entirely true for more recent saliency detectors.

3. THUMBNAIL CROPPING BASED ON VISUAL SALIENCY

3.1 Fast Saliency Detection

Eight saliency detectors among the fastest in the literature were assessed for importance map computation. They are briefly described next, maintaining the notation of the original papers when possible. The criterion for selection was to be listed among the fastest in the extensive benchmark by Borji and colleagues [3] or having comparable execution time.

Luminance Contrast (LC). This method computes the saliency of a pixel as its luminance contrast to the rest of the image. To accelerate computation, the contrast between each luminance value is computed instead and attributed to pixels with correspondent luminance [18]. Given an image $I \in R^{m \times n}$, the saliency map output by LC is defined as:

$$S_{\mathrm{LC}}(p) = \sum_{i=0}^{255} f_i D(p,i), \qquad \forall p \in I, \tag{1}$$

where l_p is the luminance of the pixel p, f_i is the frequency of the luminance level i and $D(p,i)$ is the map of luminance contrasts $||l_p - l_i||$, which can be computed in constant time.

Spectral Residual (SR). This method differs from most of the others due to its frequency-domain formulation. Given an image $I \in R^{m \times n}$, its log-spectrum representation $L(f)$ is the log of the magnitude of its Fourier Transform:

$$L(f) = log(Re(F[I])). \tag{2}$$

Saliency is then estimated as the spectral residual $R(f)$, that is, the difference between the input image and its average filtered version, both in their log-spectrum representation [9]:

$$R(f) = L(f) - h_n(f) * L(f), \tag{3}$$

where h_n is simply an averaging filter. The saliency map in the spatial domain is obtained by the Inverse Fourier Transform, which is squared to indicate the estimation error and smoothed by a Gaussian filter G_σ for better visual quality:

$$S_{\mathrm{RS}}(I) = G_\sigma * F^{-1}[exp(R(f) + P(f))]^2, \tag{4}$$

where $P(f) = Im(F[I])$ indicates the phase spectrum of the image. Before saliency map computation, the input image is downsampled to 64 pixels in width or height, to approximate the limited spatial scale of pre-attentive human vision.

Frequency-tuned (FT). This method operates on a simple premise: the average color of the image is more similar to pixels from the background than to salient pixels. Thus, the saliency of a pixel can be estimated from its color distance to the average color of the image. Given an image $I \in R^{m \times n}$, the saliency map output by FT is defined as [1]:

$$S_{\mathrm{FT}}(p) = ||I_\mu - I_G(p)||, \qquad \forall p \in I, \tag{5}$$

where I_μ is the average color of the image and $I_G(p)$ is the color of the pixel p on the Gaussian blurred version of I.

Histogram-based Contrast (HC). This method is basically an improvement of LC, which is extended to consider color difference instead of luminance contrast. This is made computationally viable through color quantization and removal of less frequent colors. Given an image $I \in R^{m \times n}$, the saliency map output by HC is defined as [5]:

$$S_{\mathrm{HC}}(p) = \sum_{i=1}^{N} f_i D(c_p, c_i), \qquad \forall p \in I, \tag{6}$$

where c_p is the color of the pixel p, N is the number of colors, f_i the bin of color c_i in the color histogram of I and $D(c_p, c_i)$ is the map of color distances.

Sparse Sampling and Kernel Density Estimation (FES). Given an image $I \in R^{m \times n}$, the saliency map output by FES is defined as [14]:

$$S_{\mathrm{FES}}(p,r,N) = A_C * [P_r^N(1|f,\bar{p})]^\alpha, \qquad \forall p \in I, \tag{7}$$

where r is the radius of the circular sampling area around the pixel p, N is the number of samples in this area, A_C is a circular averaging filter, $P_r^n(1|f,\bar{p})$ is the probability of the pixel belonging to center (as opposed to surround) given a feature vector f and that p is located at $\bar{p}$, α is an adjustable attenuation factor. Before saliency map computation, I is rescaled to 171×128 pixels.

Image Signature (IS). This method is based on the Discrete Cosine Transform (DCT). Given an image $I \in R^{m \times n}$, the saliency map output by IS is defined as [8]:

$$S_{\mathrm{IS}}(I) = g * \sum_i (\bar{I}_i \circ \bar{I}_i), \tag{8}$$

where g is a Gaussian kernel, i is the ith color channel of I, $\circ$ is the Haddamard product operator and $\bar{I}$ is the inverse DCT of the *image signature*, which is defined as the sign component of the DCT of the input I. The input image is rescaled to 64×48 pixels before saliency map computation.

Soft Image Abstraction (SIA). Proposed by Cheng and colleagues [6], this method decomposes the input image into perceptually homogeneous components using a Gaussian Mixture Model and determines the salient regions by integrating its color contrast to the other components and the spatial distribution of colors.

Difference to Random Color Samples (RS). This method describes the saliency of each pixel as its color difference to a random sample of other pixels. Given an image $I \in R^{m \times n}$, the saliency map output by RS is defined as [10]:

$$S_{\mathrm{RS}}(p) = \sum_{\forall p_r \in I_R} ||I(p) - I(p_r)||, \qquad \forall p \in I, \tag{9}$$

where I_R is a set of random pixels from I. The size of I_R is set to three pixels and the input is resized to 25% of its original size to accelerate computation. As image thumbnailing does not require as much accuracy as salient region segmentation, the joint upsampling step in the original method was replaced by Gaussian filtering to further improve execution time.

3.2 Adaptive Saliency Map Thresholding and Thumbnail Extraction

Importance maps computed using saliency detection are thresholded so that the connected components of the salient regions can be extracted. Three simple automatic thresholding methods were considered, *Achanta* for being common in salient region segmentation [1, 3], *Otsu* and *Rosin* for being well-known and having complementary characteristics:

- Achanta: Proposed by Achanta and colleagues [1] for saliency map thresholding, this method defines the threshold as twice the average saliency of the image.
- Otsu: Considering that the image is bimodal (two classes), this method determines the threshold that minimizes their intra-class variance [13].
- Rosin: Considering that the image is unimodal, this method considers a line from the peak of the image histogram to its first empty bin. The threshold is selected as the value for which the perpendicular distance between this line and the histogram is maximum [15].

Once the saliency map is computed and thresholded, the bounding box of the largest connected component is selected as cropping window.

4. EXPERIMENTS AND DISCUSSION

The assessment of the saliency detectors and automatic threshold algorithms was based on precision, recall, F-score and execution time using the MSRA1K dataset [1], which contains 1000 images with diverse unambiguously salient objects in a variety of scenes. For each image there is a corresponding ground-truth image with the salient regions labeled (by human subjects) with bounding boxes – considered as the ideal cropping based on visual saliency. The experiments were run on an Intel Core i7-860 2.80 GHz CPU with 4 GB RAM, using MATLAB.

A qualitative assessment can be made analyzing Figure 1, whereas the quantitative assessment of the methods is summarized in Table 1. The three top performances are indicated in bold. FES is the most accurate of the assessed methods, achieving the first (0.7230) and third (0.6690) highest F-scores when using Achanta's and Rosin's thresholding methods, respectively. The second highest F-score (0.6742) results from RS thresholded by Rosin's method. No thresholding method performed consistently better than the others.

Although FES has the best accuracy performance, it has the slowest execution time (97.7 ms per image), as indicated in Table 2. On the other hand, the fastest saliency detectors, SR and LC, have the worst F-scores, 0.51 and 0.47 respectively. This suggests that what is desirable is a trade-off between execution time and F-score, as can be seen more clearly in Figure 2, where points closer to the bottom right have the best combination of short execution time and high F-score. The method with the best trade-off is RS, which takes on average 20.7 ms per image, with an F-score of 0.67.

Table 1: Assessment of the saliency detectors on the MSRA1K dataset for each of the automatic thresholding algorithms considered. The images of the dataset have a typical size of 400 × 300 pixels.

	Threshold	Precision	Recall	F-score
LC	**Achanta**	0.3585	0.3278	0.3512
	Otsu	0.3712	0.3741	0.3656
	Rosin	0.4532	0.8776	0.4672
SR	**Achanta**	0.3958	0.3604	0.3875
	Otsu	0.4859	0.5268	0.4820
	Rosin	0.5014	0.7867	0.5113
FT	**Achanta**	0.4029	0.3314	0.3927
	Otsu	0.4955	0.4500	0.4864
	Rosin	0.5787	0.6562	0.5769
HC	**Achanta**	0.6170	0.5828	0.6084
	Otsu	0.6362	0.6859	0.6327
	Rosin	0.4938	0.8166	0.5042
FES	**Achanta**	0.7509	0.5685	**0.7230**
	Otsu	0.5215	0.3863	0.4990
	Rosin	0.6628	0.8632	**0.6690**
IS	**Achanta**	0.4088	0.3367	0.3966
	Otsu	0.6288	0.7200	0.6277
	Rosin	0.5555	0.8252	0.5650
SIA	**Achanta**	0.6736	0.6186	0.6648
	Otsu	0.6157	0.5764	0.6072
	Rosin	0.6374	0.8130	0.6421
RS	**Achanta**	0.5528	0.4037	0.5318
	Otsu	0.5378	0.4458	0.5226
	Rosin	0.6767	0.7492	**0.6742**

Note: saliency detector implementations from the original authors or Borji and colleagues' benchmark [3] were used.

Figure 1: Thumbnails cropped using saliency detection and automatic thresholding. For the selected examples, LC and SR tend to undercrop, while FES and IS tend to overcrop. FT, SIA and RS output the most visually pleasing thumbnails.

Table 2: Average execution time and F-score on the MSRA1K dataset. Only the best performing thresholding method for each detector was considered.

	LC[3]	SR[3]	FT[3]	HC[2]	FES[1]	IS[2]	SIA[1]	RS[3]
Time (ms)	11.9	10.5	61.6	16.9	97.7	20.1	61.8	20.7
F-score	0.47	0.51	0.58	0.63	0.72	0.63	0.66	0.67

[1]Achanta, [2]Otsu, [3]Rosin.

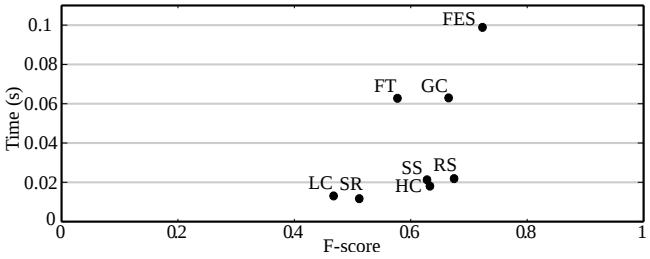

Figure 2: Trade-off between F-score and execution time of the assessed saliency detectors. The closer to the bottom right, the better the resulting trade-off.

5. CONCLUSIONS

This paper presented an assessment of fast saliency detectors for importance map computation in terms of precision, recall, F-score and execution time – with promising results for automatic image thumbnailing. In particular, saliency detection based on difference to random color samples (RS) thresholded by Rosin's method presented the best trade-off between execution time (20.7 ms/image) and F-score (0.67).

The main contributions of this paper are: (i) showing that, unlike suggested by previous work [12], saliency-based importance maps can be used for thumbnailing without additional segmentation algorithms besides thresholding, due to the accuracy of recent saliency detectors, and (ii) providing an assessment of fast saliency detectors in image thumbnailing, considering their bounding box accuracy and execution time. Future work includes strategies for grouping multiple salient regions into a cropping window and assessing the feasibility of the chosen saliency detectors to parallelization.

6. ACKNOWLEDGMENTS

The authors would like to acknowledge the Brazilian Coordination for the Improvement of Higher Education Personnel (CAPES) for the financial support of this work.

7. REFERENCES

[1] R. Achanta, S. Hemami, F. Estrada, and S. Süsstrunk. Frequency-tuned Salient Region Detection. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, pages 1597 – 1604, 2009.

[2] S. Avidan and A. Shamir. Seam Carving for Content-aware Image Resizing. *ACM Transactions on Graphics*, 26(3), July 2007.

[3] A. Borji, M. M. Cheng, H. Jiang, and J. Li. Salient Object Detection: A Benchmark. *IEEE Transactions on Image Processing*, 24(12):5706–5722, 2015.

[4] L.-Q. Chen, X. Xie, X. Fan, W.-Y. Ma, H.-J. Zhang, and H.-Q. Zhou. A Visual Attention Model for Adapting Images on Small Displays. *Multimedia Systems*, 9(4):353–364, 2003.

[5] M. M. Cheng, N. J. Mitra, X. Huang, P. H. S. Torr, and S. M. Hu. Global Contrast Based Salient Region Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(3):569–582, March 2015.

[6] M.-M. Cheng, J. Warrell, W.-Y. Lin, S. Zheng, V. Vineet, and N. Crook. Efficient Salient Region Detection with Soft Image Abstraction. In *Proceedings of the IEEE International Conference on Computer Vision*, December 2013.

[7] H. El-Alfy, D. Jacobs, and L. Davis. Multi-scale Video Cropping. In *Proceedings of the 15th ACM International Conference on Multimedia*, pages 97–106, NY, USA, 2007. ACM.

[8] X. Hou, J. Harel, and C. Koch. Image Signature: Highlighting Sparse Salient Regions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(1):194–201, Jan 2012.

[9] X. Hou and L. Zhang. Saliency Detection: A Spectral Residual Approach. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[10] M. M. I. Lie, G. B. Borba, H. Vieira Neto, and H. R. Gamba. Fast Saliency Detection Using Sparse Random Color Samples and Joint Upsampling. In *Proceedings of the 29th SIBGRAPI Conference on Graphics, Patterns and Images*, São José dos Campos, SP, Brazil, October 2016. (Forthcoming).

[11] F. Liu and M. Gleicher. Automatic Image Retargeting with Fisheye-view Warping. In *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology*, pages 153–162, NY, USA, 2005. ACM.

[12] L. Marchesotti, C. Cifarelli, and G. Csurka. A Framework for Visual Saliency Detection with Applications to Image Thumbnailing. In *Proceedings of the IEEE 12th International Conference on Computer Vision*, pages 2232–2239, Sept 2009.

[13] N. Otsu. A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man and Cybernetics*, 9(1):62–66, 1979.

[14] H. Rezazadegan Tavakoli, E. Rahtu, and J. Heikkilä. Fast and Efficient Saliency Detection Using Sparse Sampling and Kernel Density Estimation. In *Proceedings of the 17th Scandinavian Conference on Image Analysis*, pages 666–675, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[15] P. L. Rosin. Unimodal Thresholding. *Pattern Recognition*, 34(11):2083 – 2096, 2001.

[16] B. Suh, H. Ling, B. B. Bederson, and D. W. Jacobs. Automatic Thumbnail Cropping and its Effectiveness. In *Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology*, pages 95–104, NY, USA, 2003. ACM.

[17] J. Sun and H. Ling. Scale and Object Aware Image Retargeting for Thumbnail Browsing. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1511–1518, Nov 2011.

[18] Y. Zhai and M. Shah. Visual Attention Detection in Video Sequences Using Spatiotemporal Cues. In *Proceedings of the 14th ACM International Conference on Multimedia*, pages 815–824, NY, USA, 2006. ACM.