ADAPTIVE BACKGROUND SEARCH AND FOREGROUND ESTIMATION FOR SALIENCY DETECTION VIA COMPREHENSIVE AUTOENCODER

Ke Yan, Changyang Li, Xiuying Wang, Ang Li, Yuchen Yuan, Jinman Kim, Dagan Feng

School of Information Technologies, University of Sydney, Australia

ABSTRACT

In saliency object detection, inappropriate boundary-background priors is known to degrade performance in challenging image datasets, and even may lead to 'inverse' results when saliency regions are attached to the image boundaries. This is an active field where many works have proposed various techniques to lessen such degradation by inappropriate boundary-background priors. Although the use of boundary-background priors has shown to be capable of improving the detection, inherently, these techniques confront serious challenges in background suppression. To overcome this limitation, we propose an adaptive background extractor to search background seeds without the need of boundary-background priors. With the adaptive background seeds, the saliency objects can be then extracted via our proposed hierarchical foreground estimation model. We evaluate our adaptive Background Search and Foreground Estimation (BSFE) algorithm in comparison with six state-of-the-art methods on four well-recognized public datasets. The experimental results demonstrate that our BSFE algorithm outperforms compared methods in majority of the datasets and in particular achieves double-winners in terms of F-measure and mean absolute error on two challenging datasets.

Index Terms— saliency detection, autoencoder, deep learning

1. INTRODUCTION

Saliency detection aims to predict the most informative regions of the images and serves as a fundamental process for a large variety of multimedia tasks, such as in image montage [6], action recognition [27], and image segmentation [14]. As a sub-field of saliency detection, saliency object detection has gained intensive attention since it tends to extract whole meaningful objects compared to saliency fixation prediction which focuses on the human fixation locations.

A common approach for saliency object detection is to select several background seeds as the first step and then to apply various strategies to form the saliency map, such as cellular automata [16], manifold ranking [13, 25], bootstrap learning [21], Markov chain [12, 15], normalized cut [8], and

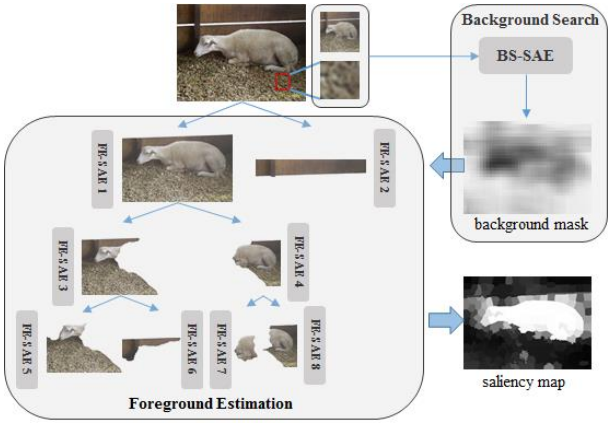

Figure 1. Overview of our proposed BSFE method for saliency detection.

foreground connectivity [20]. The background seeds selection thus is an essential step and directly affects the accuracy of the saliency detection. However, most existing methods [10, 12, 16, 21] simply use image boundaries as the background seeds. Such boundary-background seed selections are technically sound for simple image sets (e.g. MSRA-10K [7]), but are at risk of failing to produce saliency map for complex image sets (e.g. ECSSD [24] and PASCAL-S [14]) when the candidate objects are attached to the image boundaries. Although some works [13] have improved the boundary-background priors, it is still insufficient for precise saliency detection.

To deal with the above limitations, high-level features are extracted by deep neural networks for saliency region/object prediction. Compared to the conventional methods, recent works [22, 26, 28] with high-level features, generated saliency maps directly from the deep neural networks, and therefore did not rely on boundary-background priors; these high-level features proved more effective than low-level features. Hence, we exploit the high-level features that are extracted by a comprehensive autoencoder (AE), which has the advantage of exploring intrinsic structures of the input data [17].

The works of [10, 11] have studied the AE in saliency detection. However, [11] focused on saliency fixation prediction and cannot be directly applied in saliency object detection. In [10], they only utilized AE for classification and

still heavily relied on boundary-background priors. As shown in Figure 1, in this work, we propose an AE-based approach to first search the background seeds with no need of boundary-background priors and then hierarchically form the final saliency map via data reconstruction capability inherent in AE. Our work has two major contributions. Firstly, our proposed adaptive background extractor can approximate background regions semantically and cognitively, and thus improves the accuracy of saliency detection. Secondly, with the image segmentation algorithm, we hierarchically utilize the favorable capacity of data reconstruction of AE to tune the saliency map.

2. METHODOLOGY

In this section, we propose two individual stacked autoencoder (SAE) models for adaptive background search and foreground estimation respectively. As shown in Figure 2, SAE is one type of deep neural network tending to learn feature representation and data reconstruction. With a classifier (e.g. softmax regression) on the top of the feature representation layer, SAE can serve as a powerful supervised learning model for classification of unlabeled data. To better understand our algorithm, we refer readers to [10, 11] for the theory of SAE.

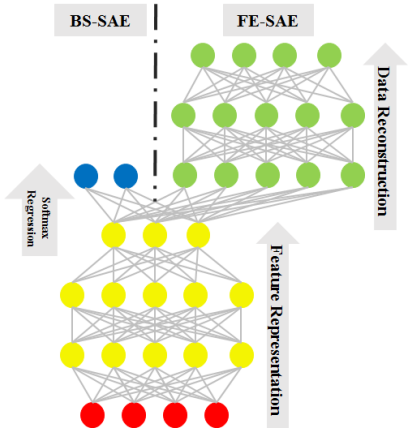

Figure 2. Illustration of SAE. The red nodes are original input data; the yellow nodes are feature representation; the green nodes are data reconstruction; and the blue nodes are outputs of softmax regression for binary classification.

2.1. Adaptive Background Search

In this sub-section, the rough background region of an image can be adaptively extracted by our proposed background search SAE model (BS-SAE), which has favorable capability of feature representation. Specifically, for a three-channel image patch p_{bs} with the size of $m \times m$ pixels from the training image I, the input vector $f(p_{bs})$ of BS-SAE is obtained by

$$f(p_{bs}) = \begin{bmatrix} g(p_{bs}) \\ g(\check{I}) \end{bmatrix} \quad (1)$$

where $\check{I} \in \mathbb{R}^{m \times m \times 3}$ is the resized image of I, and following [22], m is set to 51 in this work; $g(\cdot)$ is the vectorization operation, and thus we have $f(p_{bs}) \in \mathbb{R}^{15606 \times 1}$. As $f(p_{bs})$ is the concatenation of local context (p_{bs}) and global context ($\check{I}$), the trained BS-SAE model can infer background region from holistic view, rather than strictly local view [22] or regional view [11].

With the feature representations of each image patch by the trained BS-SAE model, we use softmax regression to measure the probability of each image patch being background. This generates a background mask M_{bs} of I, which can be utilized for further foreground estimation (Section 2.2). As shown in Figure 3, compared to the conventional boundary-background priors [9, 10, 12, 13, 21, 23, 25], such background mask can capture the background region semantically and cognitively, thus it is adaptive for background search.

Figure 3. Examples of background mask by BS-SAE.

2.2. Foreground Estimation

In this sub-section, we propose our estimation of finer object saliency with the guidance of the background mask. To improve the efficiency of our algorithm, we transform M_{bs} to a superpixel-wise background mask and use superpixel as the atomic unit in further operation. This can be easily implemented by calculating the mean value of pixels belonging to one superpixel as the probability of the superpixel being background. For brevity, we use M_{bs} defined in Section 2.1 to denote the superpixel-wise background mask unless otherwise specified. In this work, we partition each image into 250 superpixels using the SLIC algorithm [2].

With the testing image I and the corresponding background mask M_{bs}, we construct the foreground estimation SAE model (FE-SAE) to extract the foreground of I. Different from the BS-SAE model, the RGB histogram of the superpixel, with 20 bins in each color channel, are exploited as the input vector of the FE-SAE; and there is no

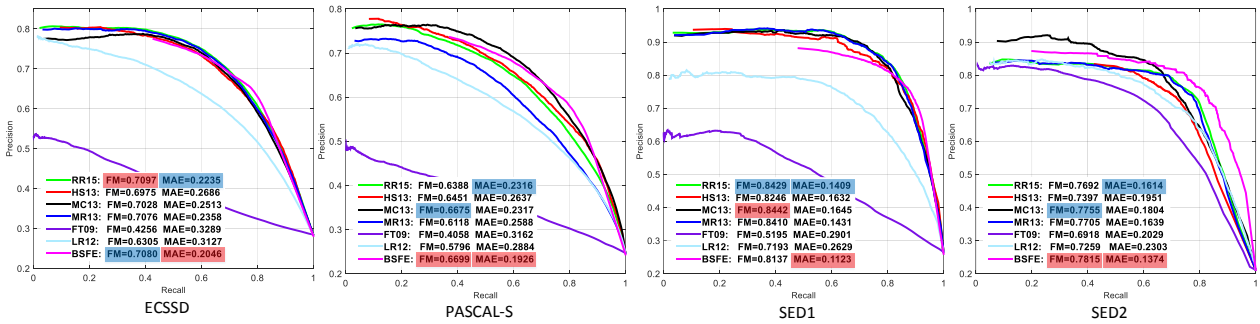

Figure 4. The PR curve, FM and MAE of benchmarking methods on four public datasets. The best and second best results are padded with red and blue rectangle respectively.

softmax regression in FE-SAE, thus it is totally an unsupervised learning model. Only those superpixels whose values on M_{bs} are more than 0.7 are selected as the training set for the FE-SAE model.

After the training of FE-SAE, we calculate the reconstruction residual $r_{p_{fe}}$ for each superpixel p_{fe} of I by

$$r_{p_{fe}} = \left\| h(p_{fe}) - \bar{h}(p_{fe}) \right\| \tag{2}$$

where $h(p_{fe})$ is the original input vector corresponding to p_{fe} and $\bar{h}(p_{fe})$ is the data reconstruction of $h(p_{fe})$ by FE-SAE. Inspired by [10], our idea is that as the FE-SAE is constructed by the background superpixels, the superpixels belonging to background have low reconstruction residual, while those belonging to foreground have high reconstruction residual. Hence, we use the reconstruction residual to measure the saliency value of p_{fe} with the following formula:

$$\begin{cases} s_{p_{fe}} = \dfrac{1}{1 + e^{\frac{\xi(u - r_{p_{fe}})}{u - v}}} \\ u = \max\{r_p : p \in \mathcal{D}\} \\ v = \dfrac{1}{|\mathcal{D}|} \sum_{p \in \mathcal{D}} r_p \end{cases} \tag{3}$$

where ξ is the smooth factor and set to 6 empirically; r_p is the reconstruction residual of superpixel p by (2); and $\mathcal{D}$ is the training set of FE-SAE.

Considering the complex background which may impede the precise foreground estimation, we hierarchically conduct foreground estimation algorithm in regional scales for better performance. Specifically, the testing image I is first segmented into two regions by Ncut algorithm [19]. Two individual FE-SAEs are then constructed respectively under the two regions and each superpixel of I is assigned to the saliency value by (3) with the corresponding FE-SAE. In the next hierarchy, we segment the two regions respectively to generate four smaller regions and construct four individual FE-SAEs corresponding to these regions. Each superpixel of I is assigned to the new saliency value by (3) in this hierarchy. Note that in each segmentation operation, only two sub-regions are generated and the region is no longer segmented when $|\mathcal{D}'| \le 0.3 \times |\mathcal{A}|$ or $|\mathcal{D}'| \ge 0.7 \times |\mathcal{A}|$, where $\mathcal{D}'$ and $\mathcal{A}$ are the training set and superpixel set respectively corresponding to the region. This process is repeated until there regions to be segmented are exhausted. Finally, the saliency value of the superpixel is obtained by linearly combining the saliency values of each hierarchy. The constructed binary segmentation tree is shown in Figure 1 and the hierarchical foreground estimation algorithm is summarized in Algorithm 1.

Algorithm 1: Hierarchical Foreground Estimation

Input: testing image I, background mask M_{bs}
Output: saliency map $S = \{s_p\}$

```
S ← 1 − M_bs
segment I into two regions I_1 and I_2 by Ncut algorithm [19]
O ← {I_1, I_2}
while O ≠ ∅:
    for each R ∈ O:
        select training set D'_R according to M_bs
        train FE-SAE
        for each superpixel p ∈ R:
            calculate saliency value s'_p by (3)
            s_p ← (s_p + s'_p)/2
        end for
        remove R from O
        if 0.3 × |R| ≤ |D'_R| ≤ 0.7 × |R| then:
            segment R into two regions R_1 and R_2 by
            Ncut algorithm
            O ← O∪{R_1, R_2}
        end if
    end for
end while
```

3. EXPERIMENT

3.1. Setup

For BS-SAE model, we stack three AEs to extract feature representation in high-level manners, with 7000, 3500 and 2000 hidden nodes in each AE, respectively. As the MSRA-10K [7] dataset provides a large variety of natural images and

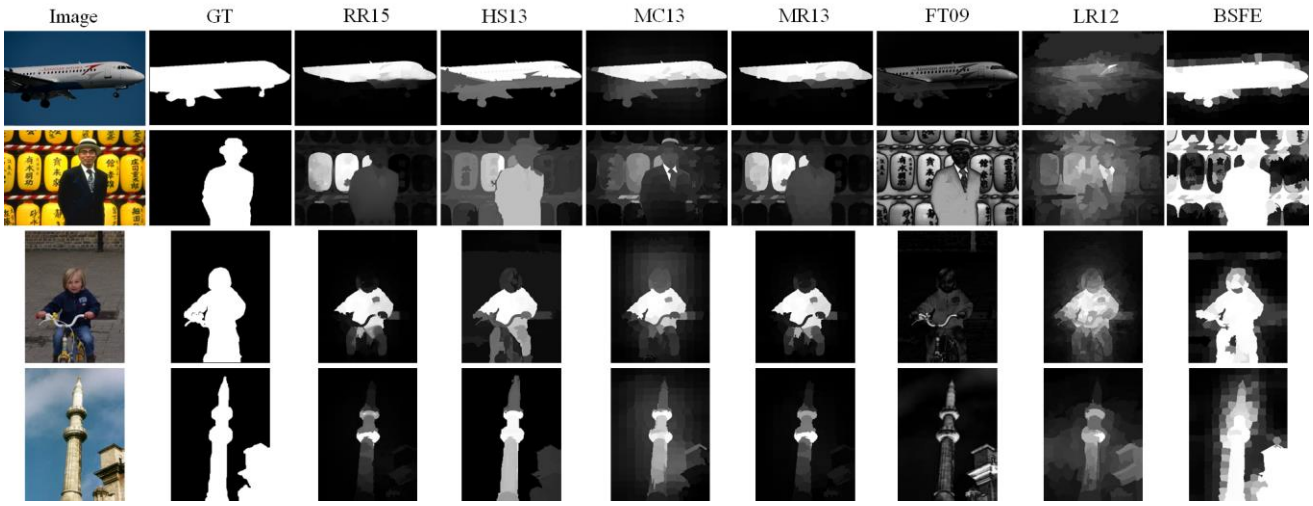

Figure 5. Example images and the saliency maps by our proposed approach (BSFE) and state-of-the-art methods.

the corresponding pixel-wise saliency annotations, we randomly selected 9000 images from the dataset to train the BS-SAE and left out 1000 images for use in the validation. As suggested in [10, 11], before input to BS-SAE, $f(p_{bs})$ is corrupted to enhance the robustness across a large training set, in which some of the units are set to be zero randomly. For FE-SAE model, we stacked two AEs to boost the performance of data reconstruction, with 60 hidden nodes in each of the AE. As the number of training samples is small (generally less than 250), we did not corrupt the original input vector in FE-SAE to make the trained model more specific to the small training set. The two models were both implemented with Theano frame [4, 5], which enabled the use of GPU to boost the speed in the training phase. The hyperparameters in the training of BS-SAE and FE-SAE are listed in Table 1.

Table 1. The hyperparameters in the training of two models.

	BS-SAE		FE-SAE	
	Pre-training	Fine-tuning	Pre-training	Fine-tuning
Training epoch	15	60	15	100
Learning rate	1e-2	1e-6 in first 20 epochs; 8e-8 in last 40 epochs.	1e-2	1e-3

3.2. Evaluation

We evaluated our proposed algorithm on four public benchmark datasets, i.e. ECSSD [24], PASCAL-S [14], SED1 [3] and SED2 [3]. Six popular state-of-the-art algorithms were chosen as comparison methods, including RR15 [13], HS13 [24], MC13 [12], MR13 [25], FT09 [1] and LR12 [18]. Following [13, 21, 22], we adopt F-measure (FM), precision-recall (PR) curve and mean absolute error (MAE) [13] to evaluate the performances. The experimental results shown in Figure 4 quantitatively demonstrate the superiority of our method on most datasets. Note that our BSFE method even achieved double-best results in terms of FM and MAE on PASCAL-S and SED2 datasets which contain more challenging scenarios with complex structures and double-salient-objects.

Figure 5 visually depicts that BSFE achieves best qualitative performance against comparison methods. For example, as shown in the first row, BSFE successfully recognized the whole saliency object while most of the other methods only recognized the main body of the airplane but failed to capture the wing and the landing gears. Such favorable performance is largely attributed to the BS-SDAE, as it can semantically infer the whole structure of the airplane from the learned features. Similarly in the third row, contrary to our method which accurately recognized the bicycle and the child as the salient objects, even the boundary-background priors based comparison methods (e.g. RR15 and MC13) failed to capture the bicycle which covers and in contact with the bottom of the image.

4. CONCLUSION

In this study, we proposed a novel AE-based method for saliency object detection. Compared to most existing algorithms which simply treat image boundaries as background query seeds, our method self-adaptively searches background via the proposed BS-SAE model. The saliency map is produced by the proposed FE-SAE model, which hierarchically utilizes the capacity of data reconstruction of AE. Our method is compared against six popular state-of-the-art methods on four datasets, demonstrating favorable superiority of our method quantitatively and qualitatively.

5. REFERENCES

[1] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in *Computer vision and*

pattern recognition, 2009. cvpr 2009. ieee conference on, 2009, pp. 1597-1604.
[2] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 34, pp. 2274-2282, 2012.
[3] S. Alpert, M. Galun, A. Brandt, and R. Basri, "Image segmentation by probabilistic bottom-up aggregation and cue integration," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 34, pp. 315-327, 2012.
[4] F. Bastien, P. Lamblin, R. Pascanu, J. Bergstra, I. Goodfellow, A. Bergeron, *et al.*, "Theano: new features and speed improvements," *arXiv preprint arXiv:1211.5590,* 2012.
[5] J. Bergstra, O. Breuleux, F. Bastien, P. Lamblin, R. Pascanu, G. Desjardins, *et al.*, "Theano: A CPU and GPU math compiler in Python," in *Proc. 9th Python in Science Conf*, 2010, pp. 1-7.
[6] T. Chen, M.-M. Cheng, P. Tan, A. Shamir, and S.-M. Hu, "Sketch2Photo: internet image montage," *ACM Transactions on Graphics (TOG),* vol. 28, p. 124, 2009.
[7] M. Cheng, N. J. Mitra, X. Huang, P. H. Torr, and S. Hu, "Global contrast based salient region detection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 37, pp. 569-582, 2015.
[8] K. Fu, C. Gong, I. Y. Gu, J. Yang, and P. Shi, "Salient object detection using normalized cut and geodesics," in *Image Processing (ICIP), 2015 IEEE International Conference on*, 2015, pp. 1100-1104.
[9] V. Gopalakrishnan, Y. Hu, and D. Rajan, "Random walks on graphs for salient object detection in images," *Image Processing, IEEE Transactions on,* vol. 19, pp. 3232-3242, 2010.
[10] J. Han, D. Zhang, X. Hu, L. Guo, J. Ren, and F. Wu, "Background prior based salient object detection via deep reconstruction residual," 2014.
[11] J. Han, D. Zhang, S. Wen, L. Guo, T. Liu, and X. Li, "Two-Stage Learning to Predict Human Eye Fixations via SDAEs," 2015.
[12] B. Jiang, L. Zhang, H. Lu, C. Yang, and M.-H. Yang, "Saliency detection via absorbing markov chain," in *Computer Vision (ICCV), 2013 IEEE International Conference on*, 2013, pp. 1665-1672.
[13] C. Li, Y. Yuan, W. Cai, Y. Xia, and D. D. Feng, "Robust Saliency Detection via Regularized Random Walks Ranking," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* vol. 34, p. 2274, 2015.
[14] Y. Li, X. Hou, C. Koch, J. M. Rehg, and A. L. Yuille, "The secrets of salient object segmentation," in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, 2014, pp. 280-287.
[15] Y. Liu, Q. Cai, X. Zhu, J. Cao, and H. Li, "Saliency detection using two-stage scoring," in *Image Processing (ICIP), 2015 IEEE International Conference on*, 2015, pp. 4062-4066.
[16] Y. Qin, H. Lu, Y. Xu, and H. Wang, "Saliency Detection via Cellular Automata," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 110-119.
[17] T. N. Sainath, B. Kingsbury, and B. Ramabhadran, "Auto-encoder bottleneck features using deep belief networks," in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, 2012, pp. 4153-4156.
[18] X. Shen and Y. Wu, "A unified approach to salient object detection via low rank matrix recovery," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, 2012, pp. 853-860.
[19] J. Shi and J. Malik, "Normalized cuts and image segmentation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 22, pp. 888-905, 2000.
[20] R. S. Srivatsa and R. V. Babu, "Salient object detection via objectness measure," in *Image Processing (ICIP), 2015 IEEE International Conference on*, 2015, pp. 4481-4485.
[21] N. Tong, H. Lu, X. Ruan, and M.-H. Yang, "Salient Object Detection via Bootstrap Learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1884-1892.
[22] L. Wang, H. Lu, X. Ruan, and M.-H. Yang, "Deep Networks for Saliency Detection via Local Estimation and Global Search," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3183-3192.
[23] W. Wang, Y. Wang, Q. Huang, and W. Gao, "Measuring visual saliency by site entropy rate," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, 2010, pp. 2368-2375.
[24] Q. Yan, L. Xu, J. Shi, and J. Jia, "Hierarchical saliency detection," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, 2013, pp. 1155-1162.
[25] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang, "Saliency detection via graph-based manifold ranking," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, 2013, pp. 3166-3173.
[26] R. Zhao, W. Ouyang, H. Li, and X. Wang, "Saliency detection by multi-context deep learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1265-1274.
[27] X. Zhen, L. Shao, and X. Li, "Action recognition by spatio-temporal oriented energies," *Information Sciences,* vol. 281, pp. 295-309, 2014.
[28] W. Zou and N. Komodakis, "HARF: Hierarchy-Associated Rich Features for Salient Object Detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 406-414.