

Saliency Propagation from Simple to Difficult (Supplementary Material)

Chen Gong^{1,2}, Dacheng Tao², Wei Liu³, S.J. Maybank⁴, Meng Fang², Keren Fu¹, and Jie Yang¹

¹Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University

²The Centre for Quantum Computation & Intelligent Systems, University of Technology, Sydney

³IBM T. J. Watson Research Center

⁴Birkbeck College, London

Please contact: jieyang@sjtu.edu.cn; dacheng.tao@gmail.com

1. Updating $\mathbf{K}_{\mathcal{L},\mathcal{L}}^{-1}$

When the iteration proceeds, the size l of labeled set \mathcal{L} will grow larger and larger, so inverting the $l \times l$ matrix $\mathbf{K}_{\mathcal{L},\mathcal{L}}$ ($\mathbf{K}_{\mathcal{L},\mathcal{L}}$ is the sub-matrix of \mathbf{K} corresponding to \mathcal{L}) at scratch under each iteration is very inefficient. Here we tackle this computational issue by incrementally updating $\mathbf{K}_{\mathcal{L},\mathcal{L}}^{-1}$ based on the previous inverting result.

As our submission, \mathcal{T} is used to denote the curriculum set with size q , and $\mathbf{K}_{\mathcal{T},\mathcal{L}}$, $\mathbf{K}_{\mathcal{L},\mathcal{T}}$, $\mathbf{K}_{\mathcal{L},\mathcal{L}}$ are sub-matrices of the kernel matrix \mathbf{K} indexed by the associated subscripts. After one iteration, the kernel matrix on the labeled set is updated by

$$\mathbf{K}_{\mathcal{L},\mathcal{L}} \leftarrow \begin{pmatrix} \mathbf{K}_{\mathcal{L},\mathcal{L}} & \mathbf{K}_{\mathcal{L},\mathcal{T}} \\ \mathbf{K}_{\mathcal{T},\mathcal{L}} & \mathbf{K}_{\mathcal{T},\mathcal{T}} \end{pmatrix}. \quad (1)$$

As a result, its inverse can be updated by using the blockwise inversion equation [5] as

$$\mathbf{K}_{\mathcal{L},\mathcal{L}}^{-1} \leftarrow \begin{pmatrix} \mathbf{K}_{\mathcal{L},\mathcal{L}}^{-1} + \mathbf{K}_{\mathcal{L},\mathcal{L}}^{-1} \mathbf{K}_{\mathcal{L},\mathcal{T}} (\mathbf{K}_{\mathcal{T},\mathcal{T}} - \mathbf{K}_{\mathcal{T},\mathcal{L}} \mathbf{K}_{\mathcal{L},\mathcal{L}}^{-1} \mathbf{K}_{\mathcal{L},\mathcal{T}})^{-1} \mathbf{K}_{\mathcal{T},\mathcal{L}} \mathbf{K}_{\mathcal{L},\mathcal{L}}^{-1} & -\mathbf{K}_{\mathcal{L},\mathcal{L}}^{-1} \mathbf{K}_{\mathcal{L},\mathcal{T}} (\mathbf{K}_{\mathcal{T},\mathcal{T}} - \mathbf{K}_{\mathcal{T},\mathcal{L}} \mathbf{K}_{\mathcal{L},\mathcal{L}}^{-1} \mathbf{K}_{\mathcal{L},\mathcal{T}})^{-1} \\ -(\mathbf{K}_{\mathcal{T},\mathcal{T}} - \mathbf{K}_{\mathcal{T},\mathcal{L}} \mathbf{K}_{\mathcal{L},\mathcal{L}}^{-1} \mathbf{K}_{\mathcal{L},\mathcal{T}})^{-1} \mathbf{K}_{\mathcal{T},\mathcal{L}} \mathbf{K}_{\mathcal{L},\mathcal{L}}^{-1} & (\mathbf{K}_{\mathcal{T},\mathcal{T}} - \mathbf{K}_{\mathcal{T},\mathcal{L}} \mathbf{K}_{\mathcal{L},\mathcal{L}}^{-1} \mathbf{K}_{\mathcal{L},\mathcal{T}})^{-1} \end{pmatrix}. \quad (2)$$

From (2) we observe that only a $q \times q$ matrix, instead of the original $l \times l$ ($l \gg q$ in later iterations) matrix, should be inverted in each iteration. Moreover, q will not be excessively large since only a small proportion of unlabeled superpixels are included into the curriculum per iteration (see Figs. 3 and 5(d) in the submission). Therefore, the proposed algorithm runs efficiently.

2. Physical Interpretation and Justification

A key factor to the effectiveness of our method is the well-ordered learning sequence from simple to difficult, which is also considered by curriculum learning [1] and self-paced learning [8]. Our paper introduces this strategy to graph-based saliency propagation. More interestingly, we provide a physical interpretation of this strategy, by relating the curriculum guided propagation to the practical fluid diffusion.

In physics, *Fick's Law of Diffusion* [2] is well-known for understanding the mass transfer of solids, liquids, and gases through diffusive means. It postulates that the flux diffuses from regions of high concentration to regions of low concentration, with a magnitude that is proportional to the concentration gradient (see Fig. 1(a)). Along one diffusive direction, the law is formulated as

$$J = -\gamma \frac{\partial h}{\partial \delta}, \quad (3)$$

where γ is the diffusion coefficient, δ is the diffusion distance, h is the concentration that evaluates the density of molecules of fluid, and J is the diffusion flux that measures the quantity of molecules flowing through the unit area per unit time.

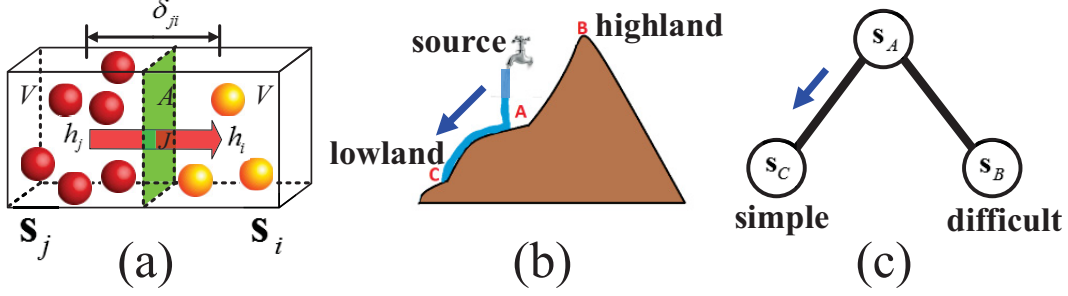


Figure 1. The physical interpretation of our saliency propagation algorithm. (a) analogies the propagation between two regions with equal difficulty to the fluid diffusion between two cubes with same altitude. The left cube with more balls is compared to the region with larger saliency value. The right cube with fewer balls is compared to the region with less saliency cues. The red arrow indicates the diffusion direction. (b) and (c) draw the parallel between fluid diffusion with different altitudes and saliency propagation guided by curriculums. The lowland “C”, highland “B”, and source “A” in (b) correspond to the simple node s_C , difficult node s_B , and labeled node s_A in (c), respectively. Like the fluid can only flow from “A” to the lowland “C” in (b), s_A in (c) also tends to transfer the saliency value to the simple node s_C .

We regard the seed superpixels as sources to emit the fluid, and the remaining unlabeled superpixels are to be diffused, among which the simple and difficult superpixels are compared to lowlands and highlands, respectively (see Figs. 1(b)(c)). There are two obvious facts here: 1) the lowlands will be propagated prior to the highlands, and 2) fluid cannot be transmitted from lowlands to highlands. Therefore, by treating γ as the propagation coefficient, h as the saliency value (equivalent to f in the submission), and δ as the propagation distance defined by $\delta_{ji} = 1/\sqrt{\omega_{ji}}$, (3) explains the process of saliency propagation from s_j to s_i as

$$J_{ji} = -m_i \gamma \frac{f_i^{(t)} - f_j^{(t)}}{\delta_{ji}} = -m_i \gamma \sqrt{\omega_{ji}} (f_i^{(t)} - f_j^{(t)}). \quad (4)$$

The parameter m_i in (4), which plays the same role as M_{ii} in Eq. (14) of our submission, denotes the “altitude” of s_i . It equals to 1 if s_i corresponds to a lowland, and 0 if s_i represents a highland. Note that if s_i is higher than s_j , the flux $J_{ji} = 0$ because the fluid cannot transfer from lowland to highland. Given (4), we have the following theorem:

Theorem 1: Suppose all the superpixels s_1, \dots, s_N in an image are modelled as cubes with volume V , and the area of their interface is A . By using m_i to indicate the altitude of s_i and setting the propagation coefficient $\gamma = 1$, the proposed saliency propagation can be derived from the fluid transmission modelled by Fick’s Law of Diffusion.

Proof. The propagation process from superpixels s_j to s_i is illustrated in Fig. 1(a). Since both superpixels are regarded as cubes with volume V , and the area of their interface is A , so during an unit time from t to $t + 1$, the amount of saliency information (similar to the number of molecules in fluids) received by s_i satisfies the following equation

$$(f_i^{(t+1)} - f_i^{(t)})V = J_{ji}A, \quad (5)$$

where J_{ji} is diffusion flux. By replacing J_{ji} with the Eq. (4) and considering $V = A/\sqrt{\omega_{ji}}$, we have the basic propagation model between two superpixels expressed as

$$f_i^{(t+1)} - f_i^{(t)} = -\gamma m_i \omega_{ji} (f_i^{(t)} - f_j^{(t)}). \quad (6)$$

Practically, a superpixel receives the saliency values from all its neighbors rather than only one as modelled by (6), so the saliency value propagated to s_i should be summed over multiple superpixels. Therefore, by treating $\omega_{ji} = 0$ if s_i and s_j are not directly linked by an edge on \mathcal{G} , (6) is extended to

$$f_i^{(t+1)} - f_i^{(t)} = -\gamma m_i \sum_{j=1 \sim N, j \neq i} \omega_{ji} (f_i^{(t)} - f_j^{(t)}), \quad (7)$$

where N is the total amount of superpixels in the image. After re-arranging (7), we obtain the following model explaining the diffusions among multiple superpixels:

$$f_i^{(t+1)} = \left(1 - \gamma m_i \sum_{j=1 \sim N, j \neq i} \omega_{ji} \right) f_i^{(t)} + \gamma m_i \sum_{j=1 \sim N, j \neq i} \omega_{ji} f_j^{(t)}. \quad (8)$$

By applying (8) to all the N superpixels $\{s_i\}_{i=1}^N$ in the image, the saliency propagation on graph \mathcal{G} can be reformulated into a compact formation

$$\mathbf{f}^{(t+1)} = \Psi \mathbf{f}^{(t)}, \quad (9)$$

where $\mathbf{f}^{(t)} = (f_1^{(t)}, f_2^{(t)}, \dots, f_N^{(t)})^T$ as defined in our submission, and

$$\Psi = \begin{pmatrix} 1 - \gamma m_1 \sum_{j=1 \sim N, j \neq 1} \omega_{j1} & \gamma m_1 \omega_{21} & \cdots & \gamma m_1 \omega_{N1} \\ \gamma m_2 \omega_{12} & 1 - \gamma m_2 \sum_{j=1 \sim N, j \neq 2} \omega_{j2} & \cdots & \gamma m_2 \omega_{N2} \\ \vdots & \vdots & \ddots & \vdots \\ \gamma m_N \omega_{1N} & \gamma m_N \omega_{2N} & \cdots & 1 - \gamma m_N \sum_{j=1 \sim N, j \neq N} \omega_{jN} \end{pmatrix}.$$

For propagation purpose, the diagonal elements in Ψ are set to 0 to avoid the self-loop on graph \mathcal{G} [16]. Therefore, (9) can be rewritten as

$$\mathbf{f}^{(t+1)} = \gamma \mathbf{M}^{(t)} \mathbf{W} \mathbf{f}^{(t)}. \quad (10)$$

By row-normalizing \mathbf{W} as $\mathbf{W} \leftarrow \mathbf{D}^{-1} \mathbf{W}$ and setting the propagation coefficient $\gamma = 1$, we achieve the employed propagation model that has the same formation as Eq. (14) in the submission. Consequently, our algorithm can be perfectly explained and derived from the practical fluid diffusion process. This completes the proof. \square

Theorem 1 reveals that our propagation strategy from simple superpixels to difficult superpixels has a close relationship with the practical fluid diffusion with highlands and lowlands.

3. Additional Experiment on DUT-OMRON Dataset

To further demonstrate the strength of our Teaching-to-Learn and Learning-to-Teach (abbreviated as ‘‘TLLT’’) algorithm, we test TLLT on a challenging dataset DUT-OMRON [15] that is much more difficult than the two databases in the submission. DUT-OMRON consists of 5168 high quality images, in which the images contain extremely complex background, and one or more salient objects of different sizes and locations. Furthermore, even the foregrounds may not show sufficient compactness and the backgrounds also contain broad diversity in many images, which present great difficulty for saliency detectors to obtain perfect results.

Fig. 2 shows the precision^w, recall^w, and F_β^w averaged over the 5168 images of our algorithm and the baselines appeared in the submission, including LD [9], GS [12], SS [4], PD [10], CT [7], RBD [17], SF [11], MR [15], GP [3], AM [6], GRD [14]. HS [13] is not compared because this method fails to apply to this dataset. We observe that our TLLT performs better than other baselines with a large margin in precision^w and F_β^w , and the resulting precision^w and recall^w are also more balanced than other algorithms. The reason that other baselines obtain higher recall^w than TLLT is that they tend to detect the most salient regions at the expense of low precision, therefore the background is very likely to be mistakenly detected as target. As a result, the imbalance between precision^w and recall^w will happen, which yield unsatisfactory F_β^w .

Fig. 3 visually compares the saliency maps of all the evaluated methods on a number of example images. Though it is tough to detect the salient regions in these testing images, the proposed method obtains near-perfect saliency maps. Comparatively, the saliency maps generated by other baselines have some defects, such as blurred foreground, incomplete foreground, and unsatisfactory background suppression, etc.

The average CPU seconds of all comparators for processing one image are reported in Tab. 1. With unoptimized matlab code, our method takes 2.90 seconds per detection, which is more efficient than LD, SS, PD, CT, and GP.

The parametric robustness of N and θ is also investigated on DUT-OMRON dataset. Fig. 4 shows the results. By fixing θ to 0.25, we notice that the F_β^w is not sensitive to the choice of N . In contrast, the parameter θ has a large influence on the final performance, and the peak value of F_β^w is achieved when $\theta = 0.25$. Above experimental results are consistent with our findings in the submission.

References

- [1] Y. Bengio, J. Louradour, R. Collobert, and J. Weston. Curriculum learning. In *Proc. International Conference on Machine Learning*, pages 41–48. ACM, 2009. 1
- [2] A. Fick. On liquid diffusion. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 10(63):30–39, 1855. 1

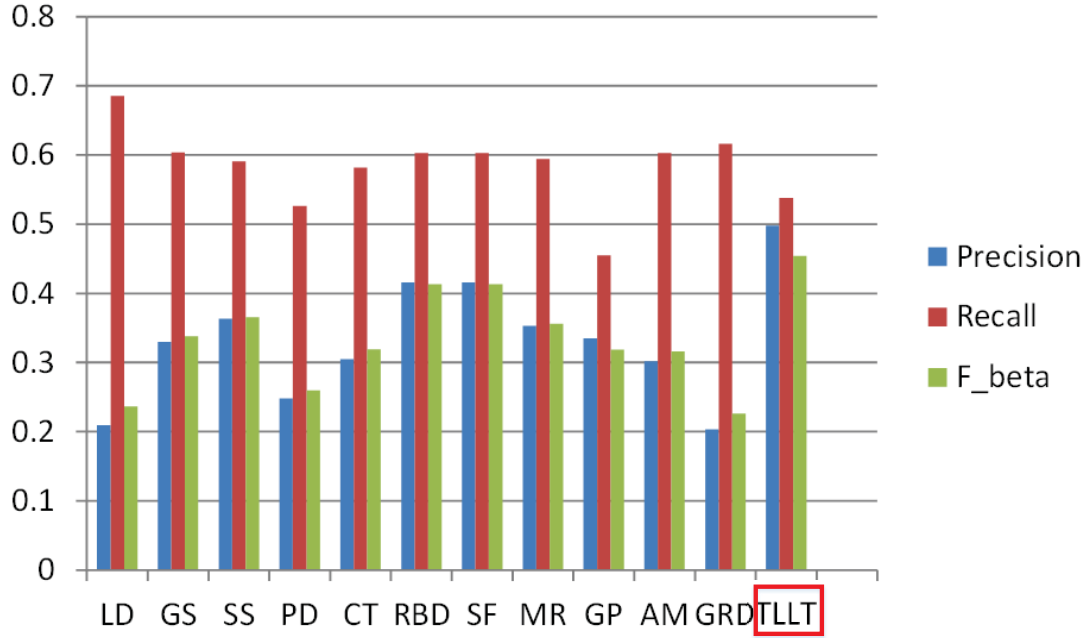


Figure 2. Comparison of different methods on DUT-OMRON dataset. TLLT generates balanced precision^w and recall^w, leading to better F_{β}^w than all the baselines.

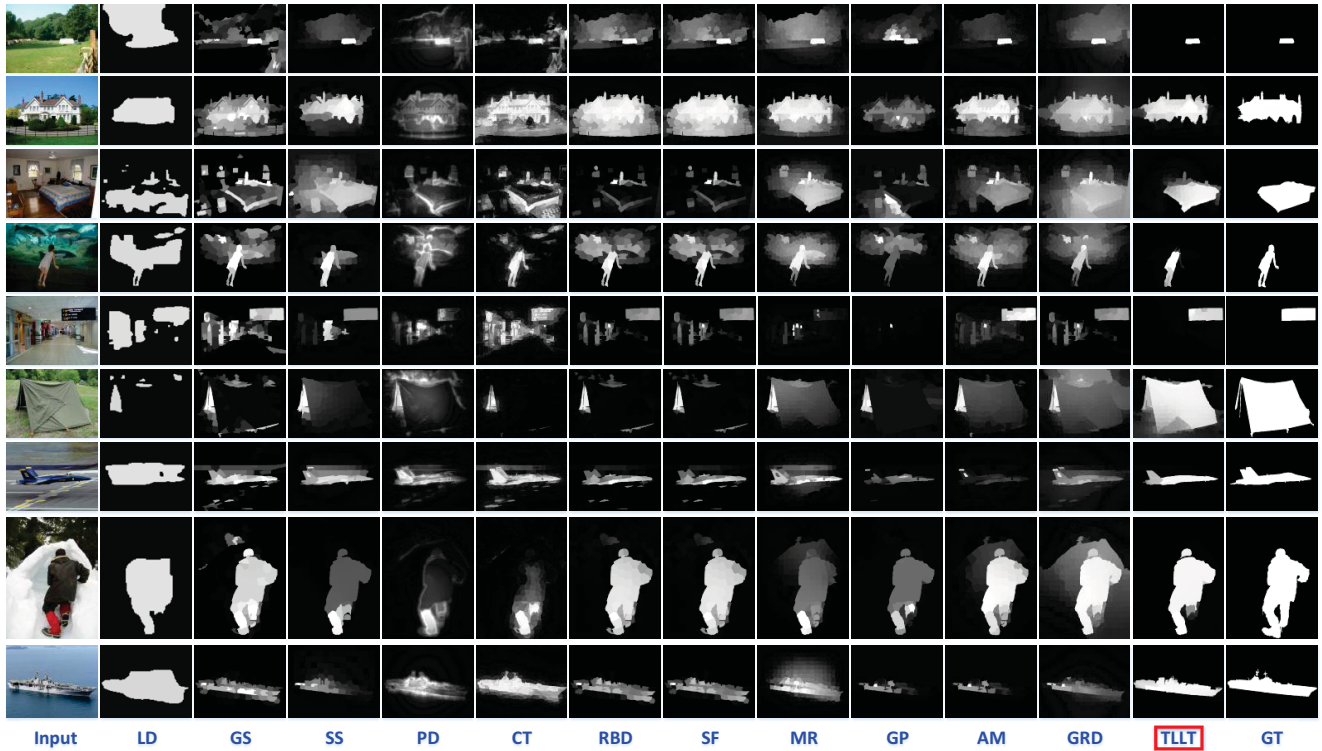


Figure 3. Visual comparison of our method with other baselines.

- [3] K. Fu, C. Gong, I. Gu, and J. Yang. Geodesic saliency propagation for image salient region detection. In *Image Processing (ICIP), IEEE Conference on*, pages 3278–3282, 2013. 3

Table 1. Average CPU seconds of all the approaches on DUT-OMRON dataset

Method	LD	GS	SS	PD	CT	RBD	SF	MR	GP	AM	GRD	TLLT
Duration (s)	6.96	0.19	3.59	2.99	3.43	0.20	0.19	1.26	2.92	0.14	0.92	2.90
Code	matlab	matlab	matlab	matlab	matlab	matlab	matlab	matlab	matlab	matlab	matlab	matlab

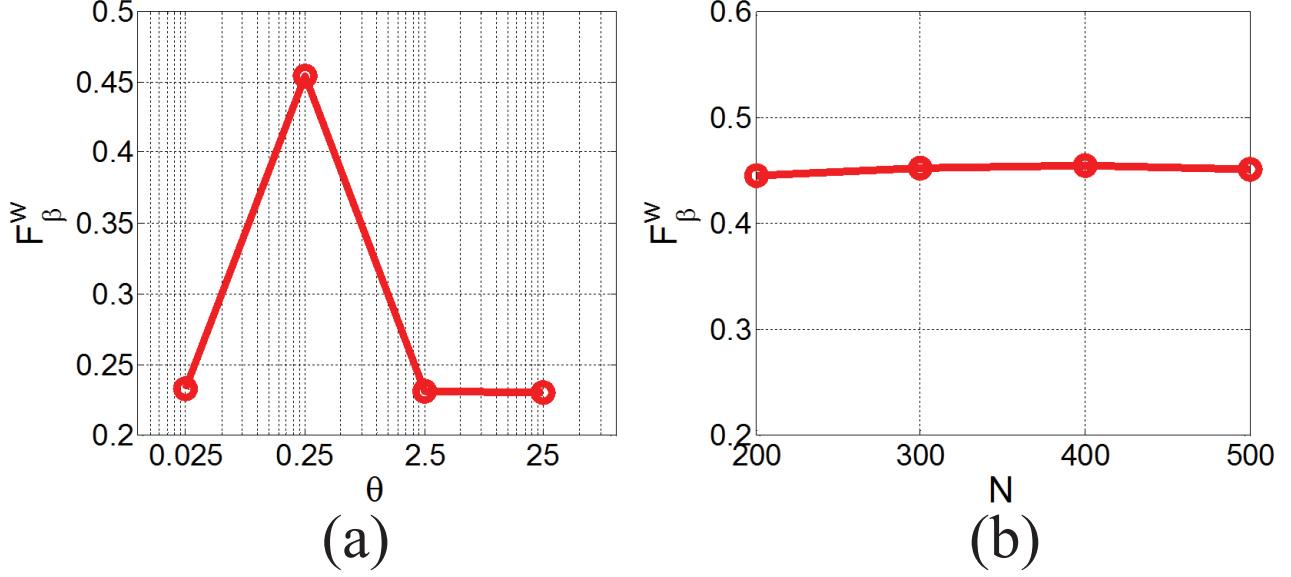


Figure 4. Parametric sensitivity on DUT-OMRON dataset. (a) shows the variation of F_{β}^w w.r.t. θ by fixing $N = 400$; (b) presents the change of F_{β}^w w.r.t. N by keeping $\theta = 0.25$.

- [4] K. Fu, C. Gong, I. Gu, J. Yang, and X. He. Spectral salient object detection. In *Multimedia and Expo (ICME), IEEE International Conference on*, 2014. 3
- [5] W. Hager. Updating the inverse of a matrix. *SIAM review*, 31(2):221–239, 1989. 1
- [6] B. Jiang, L. Zhang, H. Lu, C. Yang, and M. Yang. Saliency detection via absorbing markov chain. In *Computer Vision (ICCV), IEEE International Conference on*, pages 1665–1672. IEEE, 2013. 3
- [7] J. Kim, D. Han, Y. Tai, and J. Kim. Salient region detection via high-dimensional color transform. In *Computer Vision and Pattern Recognition (CVPR), IEEE Conference on*, pages 883–890. IEEE, 2014. 3
- [8] M. Kumar, B. Packer, and D. Koller. Self-paced learning for latent variable models. In *Advances in Neural Information Processing Systems*, pages 1189–1197, 2010. 1
- [9] T. Liu, J. Sun, N. Zheng, X. Tang, and H. Shum. Learning to detect a salient object. In *Computer Vision and Pattern Recognition (CVPR), IEEE Conference on*, pages 1–8. IEEE, 2007. 3
- [10] R. Margolin, A. Tal, and L. Zelnik-Manor. What makes a patch distinct? In *Computer Vision and Pattern Recognition (CVPR), IEEE Conference on*, pages 1139–1146. IEEE, 2013. 3
- [11] F. Perazzi, P. Krahenbuhl, Y. Pritch, and A. Hornung. Saliency filters: Contrast based filtering for salient region detection. In *Computer Vision and Pattern Recognition (CVPR), IEEE Conference on*, pages 733–740. IEEE, 2012. 3
- [12] Y. Wei, F. Wen, W. Zhu, and J. Sun. Geodesic saliency using background priors. In *European Conference on Computer Vision (ECCV)*, pages 29–42. Springer, 2012. 3
- [13] W. Yan, L. Xu, J. Shi, and J. Jia. Hierarchical saliency detection. In *Computer Vision and Pattern Recognition (CVPR), IEEE Conference on*, pages 1155–1162. IEEE, 2013. 3
- [14] C. Yang, L. Zhang, and H. Lu. Graph-regularized saliency detection with convex-hull-based center prior. *Signal Processing Letters, IEEE*, 20(7):637–640, 2013. 3
- [15] C. Yang, L. Zhang, H. Lu, X. Ruan, and M. Yang. Saliency detection via graph-based manifold ranking. In *Computer Vision and Pattern Recognition (CVPR), IEEE Conference on*, pages 3166–3173. IEEE, 2013. 3
- [16] D. Zhou and O. Bousquet. Learning with local and global consistency. In *Advances in Neural Information Processing Systems*, 2003. 3

- [17] W. Zhu, S. Liang, Y. Wei, and J. Sun. Saliency optimization from robust background detection. In *Computer Vision and Pattern Recognition (CVPR), IEEE Conference on*, pages 2814–2821. IEEE, 2014. 3