Unsupervised Saliency Detection in 3-D-Video Based on Multiscale Segmentation and Refinement

Ping Zhang, Pengyu Yan, Jiang Wu, Jingwen Liu, and Fengcan Shen

***Abstract*—In this letter, we propose an unsupervised salient object detection method in three-dimensional videos. Both temporal and depth information are efficiently considered, and multiscale architecture and graph-based refinement are built to improve accuracy and robustness. First, the input video frame is segmented into nonoverlapping superpixels by combining both appearance and depth information at the input. A multiscale architecture is also deployed after the segmentation with different segmentation parameters. Second, the initial saliency score of each segmented superpixel in each scale is calculated via global contrast, which is defined by appearance, depth, and motion cues from two consecutive frames. Third, the initial saliency in each scale is refined by smoothing over graphs built by three spatial–temporal feature priors—color, depth, and motion. Finally, the result is obtained by fusing three refined saliency maps in three scales. The experiments on two widely used datasets illustrate that our method outperforms state-of-the-art algorithms in terms of accuracy, robustness, and reliability.**

***Index Terms*—Appearance, depth, graphical model, motion, multiscale, saliency detection, segmentation.**

I. Introduction

HUMAN visual attention is most significant in human visual system, as it ensures us to pop out the most important object in a complex scene.

In 1998, Itti [1] defined "saliency" as the area that is much different from the surrounding. This definition was broadly accepted by the research community. Later, based on such definition, lots of method, such as [2]–[7], were developed to improve the performance of saliency detection in static color image. In 2015, Gong *et al.* [8] designed a saliency model based on novel propagation algorithm employing the teaching-to-learn and learning-to-teach strategies to process the complex regions. Similarly, emphasizing the entire object, Fu *et al.* [9] utilized normalized graph cut to extract the saliency area.

With the development of the three-dimensional (3-D) measuring device, depth information is naturally incorporated in the saliency-detection model. Zhang *et al.* [10] used appearance, depth, motion, illumination, and direction to calculate bottom-up saliency in an image. In [11], depth is used to weigh the 2-D saliency result. Based on that, Niu [12] constructed a depth–weight curve by assuming that the comfort zone and popping out object are more salient than any other region or object. In [13] and [14], instead of using depth as a weight bias, the authors considered it as another feature and calculated the depth-only saliency map via global contrast. In [15], based on appearance and depth information from images, bootstrap learning and multiscale fusion are designed to obtain comprehensive saliency result.

Since human visual attention is easily drawn by the fast moving objects, the spatial–temporal information between video frames could help in improving the accuracy of saliency detection. In [10] and [16], a saliency-detection method was proposed to fuse spatial, motion, and depth saliency maps to yield the final saliency map. In [17], motion and color histograms are extracted at superpixel level as local features, and at frame level as global features to obtain the saliency result. In [18] and [19], a graph considering spatial–temporal features between video frames was built to generate saliency maps. In 2018, Zhou *et al.* [20] used localized estimation and spatial–temporal refinement to pop out the salient object in videos.

The success of convolutional neural networks (CNNs) [21] in ImageNet2012 competition [22] showed that deep neural nets have advantages on feature utilizing. Li *et al.* [23] used CNN to extract multiscale features from image to obtain saliency result. In 2015, Zhao *et al.* [24] calculated the saliency via high-level feature, which is obtained through CNN networks. Later, considering the temporal information from videos, Wang *et al.* [25] used two consecutive frames as the input of the fully convolutional networks to obtain spatial–temporal saliency.

Although there are many existed methods about saliency detection, very few of them efficiently exploited both depth and temporal information from video sequences. Besides, they are limited in preserving clear edges of the foreground object when considering for a wide range of 3-D video. As for deep-learning methods, they need quantities of data and supervised training to achieve good performance.

In order to overcome aforementioned challenges, as shown in Fig. 1, we propose a 3-D video saliency-detection method considering appearance, depth, and temporal information from sequences. Multiscale architecture, initial saliency calculation via global contrast, and refinement based on graphical model are deployed. The contributions are concluded as follows.

1) We upgrade the simple linear iterative cluster algorithm (SLIC) by adding depth information and build a multiscale architecture after segmentation with different parameters.

Manuscript received May 17, 2018; revised July 2, 2018; accepted July 11, 2018. Date of publication July 18, 2018; date of current version August 3, 2018. This was supported in part by the Science and Technology Planning Project of Sichuan Province, China under Grant 2018GZ0166, and in part by the National Natural Science Foundation of China under Grant 61308102. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Wei Li. *(Corresponding author: Ping Zhang.)*

The authors are with the School of Optoelectronic Science and Engineering, University of Electronic Science and Technology of China, Chengdu 610054, China (e-mail: pingzh@uestc.edu.cn; py.yan965@gmail.com; 1625523461@qq.com; 18702505013@163.com; xiaoshenrobert@outlook.com).

Color versions of one or more of the figures in this letter are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LSP.2018.2857401

Adding depth information could make segmentation result with a well-preserved shape and edge, while multiscales improve the robustness.

2) We build a graphical model to refine the initial saliency map, which is obtained via appearance, depth, and motion from frames. Leveraging each feature prior to refining the initial saliency map improves the features' utilizing efficiency and prompts a clear background and a well-preserved foreground.

II. Proposed Method

A. Segmentation With Multiscale

1) Upgraded SLIC Model: In this part, our aim is to decompose the input image into nonoverlapping superpixels and preserve accurate object's shape and edge. While the traditional K-means algorithm in SLIC only used appearance and distance to cluster pixels, we further add depth information with appropriate weight. We set the depth feature distance as $D_d = \sqrt{(d_j - d_i)^2}$, where d_j and d_i represent the jth and ith superpixels' depth value, respectively. By giving a weight to depth feature distance as β, the complete feature distance evaluation becomes

$$D' = (1-\beta)D_c + \beta D_d \tag{1}$$

$$D = \sqrt{\left(\frac{D'}{m}\right)^2 + \left(\frac{D_S}{S}\right)^2}. \tag{2}$$

Here, D' is the weighted mean of color and depth feature distance. D_c and D_S are the Euclidean distance of color and space. Also, m is a parameter related to the largest color distance in one image and S is the expected number of superpixel. Throughout the experiment, β and m are set to 0.5 and 15, as they give the best segmentation performance.

2) Multiscale Architecture: In segmentation, a small number of superpixels reduce the ability to preserve accurate edges of objects, whereas a large number of superpixels misses objects' important corners. Furthermore, the number of optimal superpixels depends on each input frame.

In order to increase the robustness of the proposed algorithm, a multiscale approach is integrated with the SLIC algorithm. The first part of Fig. 1 shows the modified SLIC algorithm's output for an image with appearance and depth information and segmented to 200, 600, and 1000 superpixels. The feature value in each superpixel is the mean of all pixel's feature value in the superpixel area.

The remaining part of the model is calculated upon superpixel and scale level. In the last step, the results of all scales are fused using the designed way described in Section II-D.

B. Initial Saliency

In this part, within superpixel level, two steps are proposed to calculate the initial saliency map.

First, since L-a-b color correlates better to the perception of human eyes, we use CIE-Lab color space to represent the appearance of input image. Then, motion is calculated by the TV-L_1 Optical Flow [26], [27], and the two parameters of motion, v_x and v_y, can be obtained by solving the equation $I(x,y,t) = I(x+\mathrm{d}x, y+\mathrm{d}y, t+\mathrm{d}t)$, constraint by $I_x v_x + I_y v_y + I_t = 0$. I is the luminance.

Second, we use global contrast with color, depth, motion, and spatial distance to generate the initial saliency maps. We use the arithmetic mean of color, motion, and depth of all pixels in each superpixel area to represent the feature value of whole superpixel. The spatial distance is calculated among the barycenters of superpixels, and each feature distance is calculated by Euclidean distance as follows:

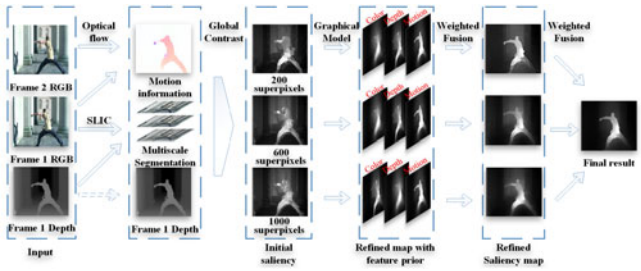

Fig. 1. Flowchart of the proposed method

$$d_C(R_j, R_i) = \sqrt{(L_{R_j} - L_{R_i})^2 + (a_{R_j} - a_{R_i})^2 + (b_{R_j} - b_{R_i})^2} \tag{3}$$

$$d_M(R_j, R_i) = \sqrt{(v_{x_{R_j}} - v_{x_{R_i}})^2 + (v_{y_{R_j}} - v_{y_{R_i}})^2} \tag{4}$$

$$d_D(R_j, R_i) = \sqrt{(D_{R_j} - D_{R_i})^2} \tag{5}$$

where R_i denotes the ith superpixel, and $L, a,$ and b are the color values under CIE-Lab. Also, d_C, d_M, and d_D are the feature distance of color, motion, and depth, respectively.

After the feature-distance calculation, according to the visual mechanism of human where the closer the two superpixels are, the more influence they had for each other, we designed the weight coefficient based on spatial distance as follows:

$$\omega(R_j, R_i) = \exp\left(-\frac{d_S(R_j, R_i)}{\sigma}\right) \tag{6}$$

where d_S represents the spatial distance among the barycenter of the superpixels. The value of the parameter σ is 0.6.

Using weight coefficient, we can obtain the score of each superpixel under each feature as follows:

$$\begin{aligned} S_F(R_i) &= \sum_{R_j \in \Omega} s_F(R_j, R_i) \\ &= \sum_{R_j \in \Omega} \omega(R_j, R_i) \cdot d_F(R_j, R_i) \end{aligned} \tag{7}$$

where the notation *F*" stands for either "*C*," "*M*," or "*D*," i.e., color, motion, and depth, correspondingly. Also, s_F is weighted "*F*" feature distance between two superpixels and $S_F(R_i)$ is the saliency score of R_i^{th} superpixel under "*F*" feature. Ω is the set of all the superpixels in one frame.

By normalizing all superpixels' score, the "F" feature saliency map is obtained. Then, we calculate the aggregation degree as the evaluation weight to fuse all feature saliency maps

$$\mu_F = \frac{\sum_{R_i \in \Omega} \sqrt{(y_{R_i} - \overline{p_{y_F}})^2 + (x_{R_i} - \overline{p_{x_F}})^2} \cdot S_F(R_i)}{\sum_{R_i \in \Omega} S_F(R_i)} \tag{8}$$

$$S = \frac{\sum_{F \in \{C,M,D\}} 1/\mu_F \times S_F}{3} \tag{9}$$

where "*F*" means this equation is used to calculate value of "*F*" feature saliency map. (x_{R_i}, y_{R_i}) is the coordinate barycenter of

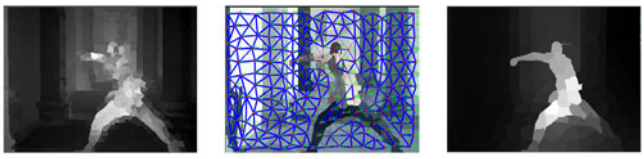

Fig. 2. Optimization based on graphical model. (a) Initial saliency. (b) Graphical model. (c) Optimized saliency.

the R_ith superpixel, $(\overline{p_{x_F}}, \overline{p_{y_F}})$ is the barycenter of the saliency area calculated by saliency scores of all superpixels, and μ_F is the aggregation degree of each feature saliency map. $S_F(R_i)$ is the "F" feature saliency score of R_i^{th} superpixel. S is the initial saliency map.

C. Saliency Refinement Based on Graphical Model

The initial saliency map, obtained from global contrast, has many artifacts at object's edge and the nearby background, as seen in Fig. 2(a). To regularize salient object's shape and clean the nearby background, a refinement based on graphical model is proposed. A probability transition function could reduce the negative impact between the salient superpixel and nearby non-salient superpixel, and balance the relationship among homogeneous salient superpixel. Thus, the saliency score of superpixels in background would be decreased, which helps to prompt a clear background and well-preserved object.

First, upon initial saliency maps, we separately use motion, color, and depth features to calculate the transition probability $P_{R_j,R_i,F}$

$$P_{R_j,R_i,F} = \frac{W_{R_i,R_j,F}}{\sum_{R_j \in \Omega_{R_i}} W_{R_i,F}} \tag{10}$$

where $F \in \{C, M, D\}$ and $W_{R_i,R_j,F} = \exp(-\frac{d_F(R_j,R_i)}{\sigma})$. We observe that the more different the two superpixels' feature are, the smaller the transition probability would be.

Based on the transition probability, we draw the lines among the superpixels' barycenters and use the thickness of the lines to represent the value of transition probability. If the feature contrast of these two superpixels are small, the line would be thick, and vice versa [see Fig. 2(b)]. In Fig. 2(b), the rough outline of the objects could be presented by the thin lines, which reflect a big difference between the objects and the background. It is also the location where the line just cross the objects' edge.

Second, we use the transition probability as the weight to process the initial saliency maps

$$U_F = \sum_{R_j,R_i \in \Omega} P_{R_j,R_i,F} \cdot s(R_j, R_i) \tag{11}$$

where $s(R_j, R_i)$ means the fused feature distance between two superpixels implied in the initial saliency maps and it can be calculated from (7) and (8) by $s(R_i, R_j) = \frac{1}{3}\sum_{F \in C,M,D} \frac{1}{\mu_F} s_F(R_j, R_i)$.

From this equation, we can gain the refined saliency map by each feature prior, for example, U_M represents the refined saliency map by motion prior.

Third, after the normalization, the three refined saliency maps by three features priors would be fused as follows:

$$U = \frac{1}{3} \sum_{F \in \{C,M,D\}} U_F. \tag{12}$$

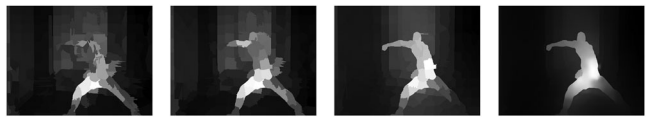

Fig. 3. Images from left to right are: initial saliency map without depth, initial saliency map with depth, refining based on graphical model, and result of multiscale approach.

The refined saliency maps are with well-preserved object's shape and edge, as shown in Fig. 2(c).

D. Fusion of Multiscale Saliency Maps

After Section II-C, there are three refined saliency maps under three scales. We name each saliency map as U_n, e.g., U_1 means the saliency map in the first scale.

$$U_A = \frac{1}{3}\sum_{n=1}^{3} U_n + \prod_{n=1}^{3} U_n. \tag{13}$$

In order to combine the advantages of each scale, we develop a fusing method by combining an average sum and a multiplication. In the sum part, we average the sum to make the saliency value of each superpixel still lying in the region $[0, 1]$. In the multiplication part, we multiple saliency maps together. The saliency value of common saliency areas in all saliency maps would be enlarged, and that value of both non-common saliency areas and common non-saliency areas in all saliency maps would be greatly diminished (because it is the multiply between 0 and 1). In the multiplication part, the saliency value of each superpixel is lying in the region $[0, 1]$ as well. The final result is obtained after the normalization of U_A.

III. Experiments and Evaluation

In this section, two steps are adapted to evaluate our experiment with two public 3-D-video-sequence datasets, which are 3D-HEVC [28] and NAMA3DS1 [29]. The first step is to vindicate every part of our proposal is rational and necessary. The second step is to compare our method with other state-of-the-art methods.

A. Vindication of Proposal

It can be seen in Fig. 3 that, by comparing the result of each step in our approach, the importance of each step in our proposal is presented. As the first two subimages in Fig. 3 show, adding depth could help in improving the performance of the segmentation and initial saliency maps. After refinement, the quality of saliency maps improves significantly, which preserved the object edge and regularized the object shape. The multiscale approach further improves the saliency result in the last subfigure as shown in Fig. 3.

B. Comparison With Other Methods

In the next experiments, we compare our proposal with 30 state-of-the-art saliency-detection methods: ITTI [1], GBVS [30], FT [31], MR [7], SPL [32], MDF* [23], RFCN* [33], RGBD [14], WANG2DV [34], FCN* [25], ZHANG3DV [10], LINO3DV [16], and FCN-D* [25], where the proposals with "*" are deep-learning-based methods. In order to conduct the experiments more convincingly, we slightly

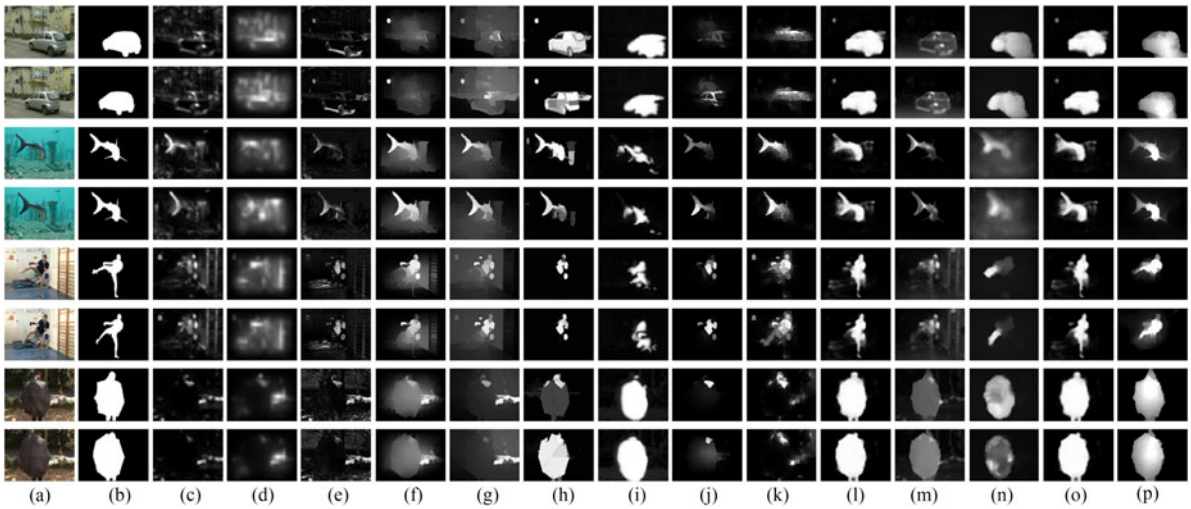

Fig. 4. Example of saliency results of differen methods. (a) Original frame. (b) Ground truth. (c)–(o) Results of ITTI [1], GBVS [30], FT [31], MR [7], SPL [32], MDF* [23], RFCN* [33], RGBD [14], WANG2DV [34], FCN* [25], ZHANG3DV [10], LINO3DV [16], and FCN-D* [25]. (p) Our proposal.

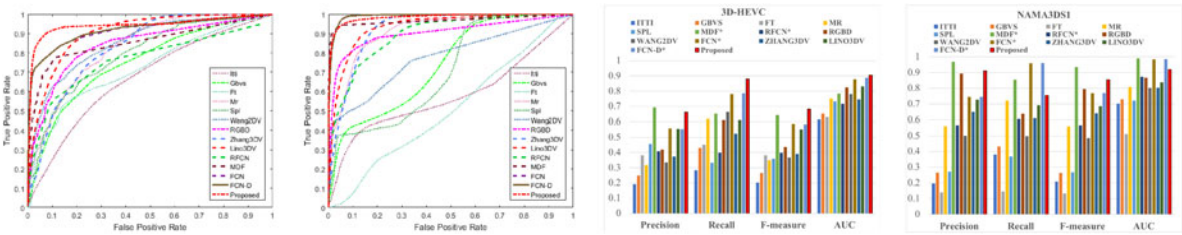

Fig. 5. Evaluation. From left to right: ROC on 3D-HEVC and NAMA3DS1, and chart on 3D-HEVC and NAMA3DS1

TABLE I
MEASUREMENT

Type		2D							3D	2DV		3DV			
Datasets	Models	ITTI	GBVS	FT	MR	SPL	MDF*	RFCN*	RGBD	WANG2DV	FCN*	ZHANG3DV	LINO3DV	FCN-D*	Proposed
3D-HEVC	Precision	0.1926	0.2496	0.3815	0.3180	0.4567	0.6938	0.4086	0.4192	0.3350	0.5595	0.3745	0.5562	0.5546	0.6626
	Recall	0.2846	0.4303	0.4521	0.6191	0.3331	0.6533	0.3993	0.6109	0.6637	0.7811	0.5249	0.6112	0.7856	0.8802
	F-measure	0.2027	0.2660	0.3825	0.3506	0.3594	0.6438	0.3968	0.4367	0.3684	0.5888	0.3908	0.5527	0.5849	0.6821
	AUC	0.6157	0.6526	0.6314	0.7521	0.7336	0.7840	0.7178	0.8251	0.7811	0.8792	0.7463	0.8318	0.8886	0.9057
NAMA3DS1	Precision	0.1981	0.2655	0.1392	0.5633	0.2720	0.9688	0.5681	0.8943	0.5017	0.7455	0.6509	0.7277	0.7454	0.9114
	Recall	0.3804	0.4336	0.1456	0.7224	0.3681	0.8548	0.6067	0.6382	0.4988	0.9604	0.6115	0.6929	0.9616	0.7556
	F-measure	0.2095	0.2643	0.1331	0.5611	0.2684	0.9353	0.5684	0.7967	0.4887	0.7696	0.6409	0.6863	0.7697	0.8553
	AUC	0.7017	0.7305	0.5129	0.8077	0.7225	0.9901	0.8727	0.8667	0.8022	0.9848	0.8033	0.8380	0.9851	0.9197

modified the FCN deep-learning method by guiding the origin saliency results with depth maps to form it into a "3-D-video-based" method, named as FCN-D.

The qualitative comparisons are presented in Fig. 4, which consistof four pairs of two consecutive frames–two pairs are from 3D-HEVC and the other two pairs are from NAMA3DS1, and the results show that our proposed method achieves the best visual effects. The ROC curves and the chart of measures' value on two datasets are draw in Fig. 5. Detailed evaluation measures, including precision, recall, F-measure and AUC, are presented in Table I. From the results, it could be concluded that our method outperforms all other methods on the first datasets and exceeds all traditional methods on the second datasets. However, due to the weak depth maps in the second datasets, our method yields almost same but slightly poor results compared with three deep-learning methods. Nevertheless, compared with deep-learning saliency detection, our method is unsupervized and no need of quantitative manufactured datasets. Our proposal is implemented by using C++ with OpenCV 3 in a desktop with i7-6700 K and 16-GB RAM and it costs 26.37 s to obtain a saliency image from video frames. The time-cost comparisons with the state-of-the-art methods are shown in Table II.

TABLE II
TIME-CONSUMING EVALUATION

Models	ITTI	GBVS	FT	MR	SPL	MDF*	RFCN*
Time Cost	0.23 s	0.17 s	0.21 s	1.53 s	3.38 s	81.23 s	28.49 s
Models	RGBD	WANG2DV	FCN*	ZHANG3DV	LINO3DV	FCN-D*	Proposed
Time Cost	1.71 s	2.34 s	0.7 s	7.82 s	5.35 s	1.3 s	26.37 s

IV. CONCLUSION

In this letter, we improved the SLIC algorithm and build a multiscale architecture based on segmentation. Additionally, a refining method based on graphical model is adopted to improve the object's shape and edge in the saliency maps. The experiments on 3D-HEVC and NAMA3DS1 datasets verify the excellent performance of our proposal. The saliency maps using the proposed method consist of objects with clear shape and edge, demonstrating robustness of the proposed algorithm.

REFERENCES

[1] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Trans. Pattern Analy. Mach. Intell.*, vol. 20, no. 11, pp. 1254–1259, Nov. 1998.

[2] D. Gao and N. Vasconcelos, "Bottom-up saliency is a discriminant process," in *Proc. IEEE 11th Int. Conf. Comput. Vis.*, Oct. 2007, pp. 1–6.

[3] D. Gao, V. Mahadevan, and N. Vasconcelos, "On the plausibility of the discriminant center-surround hypothesis for visual saliency," *J. Vis.*, vol. 8, no. 7, pp. 1–18, 2008.

[4] M. M. Cheng, N. J. Mitra, X. Huang, P. H. S. Torr, and S. M. Hu, "Global contrast based salient region detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 569–582, Mar. 2015.

[5] F. Perazzi, P. Krhenbhl, Y. Pritch, and A. Hornung, "Saliency filters: Contrast based filtering for salient region detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 733–740.

[6] B. Schlkopf, J. Platt, and T. Hofmann, *Graph-Based Visual Saliency*. Cambridge, MA, USA: MIT Press, 2007, pp. 545–552. [Online]. Available: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6287326

[7] C. Yang, L. Zhang, H. Lu, X. Ruan, and M. H. Yang, "Saliency detection via graph-based manifold ranking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 3166–3173.

[8] C. Gong *et al.*, "Saliency propagation from simple to difficult," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 2531–2539.

[9] K. Fu, C. Gong, I. Y. H. Gu, and J. Yang, "Normalized cut-based saliency detection by adaptive multi-level region merging," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 5671–5683, Dec. 2015.

[10] Y. Zhang, G. Jiang, M. Yu, and K. Chen, "Stereoscopic visual attention model for 3D video," in *Proc. Int. Conf. Advances Multimedia Model.*, 2010, pp. 314–324.

[11] C. Chamaret, S. Godeffroy, P. Lopez, and O. L. Meur, "Adaptive 3D rendering based on region-of-interest," in *Stereoscopic Displays and Applications XXI*, vol. 7524, 2010, p. 75240V.

[12] Y. Niu, Y. Geng, X. Li, and F. Liu, "Leveraging stereopsis for saliency analysis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 454–461.

[13] Y. Fang, J. Wang, M. Narwaria, P. L. Callet, and W. Lin, "Saliency detection for stereoscopic images," *IEEE Trans. Image Process.*, vol. 23, no. 6, pp. 2625–2636, Jun. 2014.

[14] H. Peng, B. Li, W. Xiong, W. Hu, and R. Ji, "RGBD salient object detection: A benchmark and algorithms," *Nat. Lab. Pattern Recognit.*, vol. 8691, pp. 92–109, 2014.

[15] H. Song, Z. Liu, H. Du, G. Sun, O. L. Meur, and T. Ren, "Depth-aware salient object detection and segmentation via multiscale discriminative saliency fusion and bootstrap learning," *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4204–4216, Sep. 2017.

[16] L. Ferreira, L. A. da Silva Cruz, and P. Assuncao, "A method to compute saliency regions in 3d video based on fusion of feature maps," in *Proc. IEEE Int. Conf. Multimedia Expo*, Jun. 2015, pp. 1–6.

[17] Z. Liu, X. Zhang, S. Luo, and O. L. Meur, "Superpixel-based spatiotemporal saliency detection," *IEEE Trans. Circ. Syst. Video Technol.*, vol. 24, no. 9, pp. 1522–1540, Sep. 2014.

[18] K. Fu, I. Y. H. Gu, Y. Yun, C. Gong, and J. Yang, "Graph construction for salient object detection in videos," in *Proc. 22nd Int. Conf. Pattern Recognit.*, Aug. 2014, pp. 2371–2376.

[19] Z. Liu, J. Li, L. Ye, G. Sun, and L. Shen, "Saliency detection for unconstrained videos using superpixel-level graph and spatiotemporal propagation," *IEEE Trans. Circ. Syst. Video Technol.*, vol. 27, no. 12, pp. 2527–2542, Dec. 2017.

[20] X. Zhou, Z. Liu, C. Gong, and W. Liu, "Improving video saliency detection via localized estimation and spatiotemporal refinement," *IEEE Trans. Multimedia*, to be published.

[21] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[22] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.

[23] G. Li and Y. Yu, "Visual saliency based on multiscale deep features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 5455–5463.

[24] R. Zhao, W. Ouyang, H. Li, and X. Wang, "Saliency detection by multi-context deep learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1265–1274.

[25] W. Wang, J. Shen, and L. Shao, "Video salient object detection via fully convolutional networks," *IEEE Trans. Image Process.*, vol. 27, no. 1, pp. 38–49, Jan. 2018.

[26] C. Zach, T. Pock, and H. Bischof, "A duality based approach for real-time tv-l 1 optical flow," *in Proc. Joint Pattern Recognit. Symp.*, 2007, pp. 214–223.

[27] J. S. Pérez, E. Meinhardt-Llopis, and G. Facciolo, "Tv-l1 optical flow estimation," *Image Process. Online*, vol. 2013, pp. 137–150, 2013.

[28] K. Mller *et al.*, "3D high-efficiency video coding for multi-view video and depth data," *IEEE Trans. Image Process.*, vol. 22, no. 9, pp. 3366–3378, Sep. 2013.

[29] M. Urvoy *et al.*, "NAMA3DS1-COSPAD1: Subjective video quality assessment database on coding conditions introducing freely available high quality 3d stereoscopic sequences," in *Proc. 4th Int. Workshop Quality Multimedia Experience*, Jul. 2012, pp. 109–114.

[30] B. Scholkopf, J. Platt, and T. Hofmann, *Graph-Based Visual Saliency*. Cambridge, MA, USA: MIT Press, 2007, pp. 545–552. [Online]. Available: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6287326

[31] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 1597–1604.

[32] C. Yang, L. Zhang, and H. Lu, "Graph-regularized saliency detection with convex-hull-based center prior," *IEEE Signal Process. Lett.*, vol. 20, no. 7, pp. 637–640, Jul. 2013.

[33] L. Wang, L. Wang, H. Lu, P. Zhang, and X. Ruan, "Saliency detection with recurrent fully convolutional networks," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 825–841.

[34] W. Wang, J. Shen, and F. Porikli, "Saliency-aware geodesic video object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 3395–3402.