Visual Saliency Detection Algorithm in Compressed HEVC Domain

Rui Bai, Wei Zhou✉, Guanwen Zhang, Henglu Wei
* Northwestern Polytechnical University, Xi’an, China
E-mail: zhouwei@nwpu.edu.cn

***Abstract*—Saliency detection has been widely used to predict human fixation. In this paper, a Visual Saliency Detection Algorithm in Compressed HEVC Domain is proposed which consists of three parts: static saliency detection, dynamic saliency detection and competitive fusion. Firstly, the Gauss model is used to filter out the background of the static features which are extracted by down-sampling and DCT. Secondly, the motion vectors are used to represent the dynamic feature. Then the dynamic saliency is calculated by filtering out the background of dynamic feature. Finally, the competitive fusion model is used to adaptively combine the characteristic of static and dynamic saliency maps. Experimental results show that the proposed method is superior to classic state-of-the-art saliency detection methods with 0.05 AUC value increasing and 0.17 KL divergence decreasing on average. The average time of one frame detection is 2.3 seconds.**

I. Introduction

Saliency detection comes from human visual system (HVS) and aims to detect areas of concern to human eyes and filter out unimportant areas [1]. Saliency detection models are widely used to automatically extracted region of interest (ROI) in image/frame. According to the size of image/frame saliency, a reasonable allocation strategy for computing resources is formulated. Visual saliency models are widely used in object detection [2], object recognition [3], image retargeting [4], image quality assessment, image/frame compression and coding. The basic principle of image/frame compression and coding is using lossless compression or lossless compression for saliency regions, and using lossy compression for background areas. Such a principle can not only guarantee the quality of image/frame, but also maintain a high compression ratio [5].

Various saliency detection models have been developed. These models are separated according to two mechanisms: bottom-up and top-down. The bottom-up refers to low level visual features and data-driven fast processing. C. Koch and S. Ullman put forward a very influential biological inspiration model [6]. Itti et al. [7] found out the low-level features of intensity, color and detected static image saliency regions. Several studies tried to detect salient regions in image and video [7], [8], [9]. The top-down refers to slow processing based on task-driven and conscious control. Existing top-down models are designed to learn prior knowledge firstly, and use prior knowledge to guide saliency detection. Hou and Zhang [10] presented a fast Fourier spectrum residual method. RS et al. [11] used Bayesian framework to calculate image saliency models. Most of top-down saliency detection models need to learn large database of images, and the computation is huge.

In image saliency detection, only static features need to be extracted. However, not only static features need to be extracted, but also dynamic features need to be extracted in the saliency detection of video. The dynamic saliency map is an important factor to attract human beings attention [12], [13]. Currently, most of the formats of video storage are MPEG2, H.264 and HEVC. Several studies tried to detect saliency regions in compressed domain such as MPEG2, H.264 [14], [15]. Only a few saliency models are designed for HEVC. The latest research on saliency of HEVC was presented by Mai Xu et al [16]. They established eye tracking data sets and detected video saliency with HEVC features. The work of [16] which follows the top-down mechanism is very complex and time cosuming.

In this paper, a saliency detection algorithm in compressed HEVC domain is proposed. The HEVC coded videos are used for saliency detection. Our method includes static detection, dynamic detection and competitive fusion. Firstly, the static features which include chroma, luminance and texture are extracted by down-sampling of color components and the DCT coefficients of Y component. Then, the background of static features is filtered out by Gauss model and a static saliency map is calculated. Next, the dynamic feature is represented by motion vector (MV) and a new picture which is calculated by Coding Unit (CU) depth and bit allocation is used to filtered out the background of dynamic saliency. Finally, the final saliency map is calculated by competitive fusion model. In this paper, the bottom-up is the focus of research.

The rest of this paper is organized as follows. Section II presents the saliency detection algorithm in compressed HEVC domain. Experimental results are presented and discussed in Section III. Section IV concludes the works in this paper.

II. The proposed algorithm

In this section, we introduce the proposed compressed HEVC domain saliency detection algorithms. The framework is shown in Fig.1. Firstly, the methods of down-sampling and DCT are used to extracted static features. Then, the center-surrounding difference of the static features is extracted by Gauss mode and the static saliency map is obtained. Secondly, the motion vector is used to represent the dynamic feature and the background of the dynamic feature is filtered out. Finally, an adaptive fusion algorithm which based on competition is proposed and the final saliency map can be obtained.

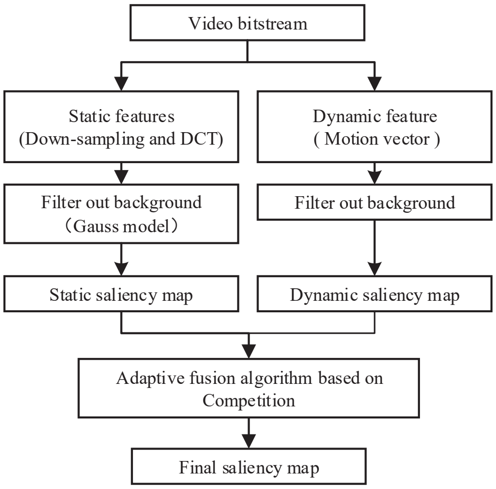


Fig. 1. Proposed framework.

A. Static Saliency Detection Model

In HEVC, the format of input video sequence is YCbCr which contains three color components Y, Cb and Cr. The static saliency maps contains three parts, namely chroma, luminance and texture.

1) Extract Static Features: Firstly, chroma features and luminance feature are calculated by down-sampling. Each frame is divided into multiple blocks of fixed size. The size of the block is 8×8. Each block contains a luminance component Y and two chroma components Cr and Cb. The luminance feature is extracted by down-sampling of the luminance components Y. The chroma features contain two parts which are extracted by down-sampling of color components Cr and Cb, respectively. The specific calculation of the down-sampling is shown in (1),

$$\left[\begin{array}{l} L^k = \sum\limits_{i=1}^{64} {Y_i}^k/64 \\ C_1^k = \sum\limits_{i=1}^{64} {Cb_i}^k/64 \\ C_2^k = \sum\limits_{i=1}^{64} {Cr_i}^k/64 \end{array}\right. \tag{1}$$

where L^k, C_1^k and C_2^k denote a luminance and two kinds of chroma static features of the k-th 8×8 block, and one luminance (L) and two chroma static feature maps (C_2 and C_1) can be obtained; ${Y_i}^k$, ${Cb_i}^k$ and ${Cr_i}^k$ represent the i-th pixel value of the Y, Cb and Cr components in the k-th 8×8 block, respectively.

Then, the texture feature is calculated by Discrete Cosine Transform (DCT). Alternate Current (AC) coefficients include the detailed frequency information. In fact, most images contain more low-frequency components and the human eye is not sensitive to the high-frequency image [15]. Therefore, part of AC coefficients are chosen to denote the texture feature in (2),

$$T^k = \{AC^k_{Y_{(0,1)}}, AC^k_{Y_{(1,0)}}, AC^k_{Y_{(2,0)}}, AC^k_{Y_{(1,1)}}, AC^k_{Y_{(0,2)}}\} \tag{2}$$

where T^k is a multidimensional vector which denotes the texture feature of the k-th 8×8 block and texture static feature map (T) can be obtained; $AC_{Y_{(i,j)}}$ is the AC coefficient with coordinate (i, j) in the k-th 8×8 DCT block. Four static feature maps (L, C_2, C_1 and T) are obtained by the above calculation.

2) Filter Out Static Backgrounds: The static feature maps (L, C_2, C_1 and T) have a lots of static background parts which reduce the accuracy of saliency detection. In this paper, the Gauss model is used to filtered out the background. Specifically, each point in the static feature maps is considered as the center point and influenced by the surrounding points. The influence of each surrounding point is calculated by Gauss model which is defined by (3),

$$\alpha_{(x_s,y_s)} = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x_s - x_c)^2 + (y_s - x_c)^2}{2\sigma^2}\right) \tag{3}$$

where $\alpha_{(x_s,y_s)}$ is the influence of the surrounding point coordinate (x_s, y_s); σ is constant and $\sigma= 40$; (x_c, x_c) is the center point coordinate. The influence of all surrounding points represent the static saliency value of center point. The static saliency value for center point is calculated by (4),

$$S1^\lambda_{(x_c,y_c)} = \sum_{x_s=0}^{w}\sum_{y_s=0}^{h}\left(\alpha_{(x_s,y_s)} \times \left(v^\lambda_{(x_s,y_s)} - v^\lambda_{(x_c,y_c)}\right)\right) \tag{4}$$

where $S1^\lambda_{(x_c,y_c)}$ is the saliency value of center point coordinate (x_c, y_c) with the feature map λ, and $\lambda \in \{L, C_1, C_2, T\}$; w and h are the width and height of the static feature maps, respectively; $v^\lambda_{(x_s,y_s)}$ and $v^\lambda_{(x_c,y_c)}$ are the pixel values with the surrounding point coordinate (x_s, y_s) and center point coordinate (x_c, y_c) in the static feature mapλ. Statics saliency maps ($S1^L$, $S1^{C_1}$, $S1^{C_2}$ and $S1^T$) can be obtained.

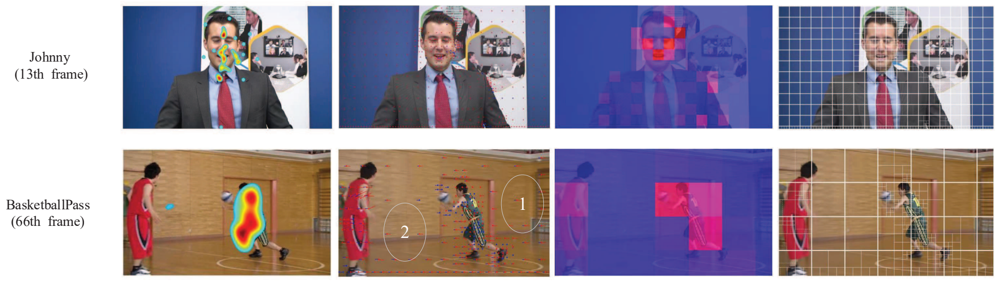
Then, the final static saliency map is obtained by fusing four static saliency maps. The linear weighting method is used for fusion. The weight of each part is 1/4. The final static saliency map is calculated by (5),

$$SS = \frac{S1^L + S1^{C_1} + S1^{C_2} + S1^T}{4} \tag{5}$$

where SS, $S1^L$, $S1^{C_1}$, $S1^{C_2}$ and $S1^T$ denote the final static saliency map, the luminance saliency map, the first-chroma saliency map, the second-chroma saliency map and the texture saliency map, respectively.

B. Dynamic Saliency Detection Model

The moving parts of videos are mainly concerned for people, so they are extracted to represent the dynamic saliency map in video frames.



(a) Human Fixations (b) MV (c) Bit Allocation (d) CU Depth

Fig. 2. The Human Fixations and HEVC Features.

1) Extract Dynamic Features: People focus on the moving parts in videos and MV can be used to detect video saliency, as shown in the Fig.2 (a). Some videos have static background, such as the 13th frame of *Johnny*. After encoding, the background parts of these videos contain very few MV. Therefore, regions with large motion vectors are considered as saliency regions. MV is extracted as a basic HEVC feature in our method. Each point can extract a motion vector in HEVC compressed domain. The basic dynamic saliency map is calculated by (6),

$$BSM_{(i,j)} = V_{(i,j)} \tag{6}$$

where $BSM_{(i,j)}$ is the basic dynamic saliency value with coordinate (i,j); $V_{(i,j)}$ is the motion vector with coordinate (i,j).

2) Filter Out The Dynamic Background: Unfortunately, the background of some videos is moving, such as the 66th frame of *Basketball* in Fig.2 (b). The part 1 and part 2 have lots of MVs which cause pseudo saliency region. Saliency detection regions are seriously affected. In this paper, CU depth and bit allocation are extracted in HEVC compressed domain and used to filter out the pseudo saliency region.

The maximum size of CU is 64×64, and the corresponding depth is 0. The minimum size is 8×8 , and the corresponding depth is 3. CU depth can be used to detect video saliency. As shown in Fig.2 (d), the saliency regions correspond to larger CU depths. If more bits are allocated to a CTU, it generally indicates that the CTU contains more valid information. As shown in Fig.2 (c), the saliency regions correspond to larger bit allocation.

A new map is calculated by CU depth and bit allocation and used to distinguish background and subject of the basic dynamic map (BSM). The new map is defined by (7) and (8),

$$pic_{(i,j)} = \begin{cases} 0 & d_{(i,j)} \times b_{(i,j)} < Th \\ 1 & d_{(i,j)} \times b_{(i,j)} > Th \end{cases} \tag{7}$$

$$\text{Th} = \frac{2 \times \left(\sum\limits_{(i,j)\in frame} d_{(i,j)} \times b_{(i,j)} \right)}{n_1} \tag{8}$$

where $pic_{(i,j)}$ is the pixel value in the new picture with coordinate (i,j); Th is the threshold for each frame; $d_{(i,j)}$ is the CU depth with coordinate (i,j); $b_{(i,j)}$ is the bit allocation with coordinate (i,j); n_1 is the total pixel number in the frame.

Finally, the final dynamic saliency map can be calculated by (9),

$$SM_{(i,j)} = Norm(pic_{(i,j)}) \times BSM_{(i,j)}) \tag{9}$$

where $SM_{(i,j)}$ is the final dynamic saliency value with coordinate (i,j); $Norm$ is the normalization operation.

As there is no dynamic saliency map for unpredicted frames (I-frame), the dynamic saliency map of the previous predicted frame is adopted to represent that of the current unpredicted frame.

C. Competitive Fusion Algorithm

The static saliency and dynamic saliency map are combined by fusion algorithm. In this paper, an adaptive fusion algorithm based on competition is presented by (10) and (11),

$$S = Norm\left(a_1 \times SS + a_2 \times SM + a_3 \times SF\right) \tag{10}$$

$$SF = SS \times SM \tag{11}$$

where a_1, a_2 and a_3 are the parameters to control the weight of static, dynamic and mixed map, respectively; SS, SM, SF and S denote the static, dynamic, mixed and final saliency map, respectively; $Norm$ is the normalization operation. The parameters (a_1, a_2 and a_3) calculated by (12) and (13),

$$\left[\begin{array}{l} a_1 = 1 \\ a_2 = \left(\frac{v^{SS}}{v^{SM}}\right)^{\frac{1}{2}} \\ a_3 = 2 \times \left(\frac{v^{SM}}{v^{SF}} \times \frac{v^{SS}}{v^{SF}}\right)^{\frac{1}{2}} \end{array}\right. \tag{12}$$

$$v^k = \left(\frac{1}{n_2}\sum\nolimits_{(i,j)} \left(S^k_{(i,j)} - \overline{S^k}\right)^2\right)^{\frac{1}{2}} \tag{13}$$

where $S^k_{(i,j)}$ is the saliency value with coordinate (i,j) in the saliency map k, and $k \in (SS, SM, SF)$; $\overline{S^k}$ is the mean of the saliency map k; n_2 represents the total number of pixels in a frame of video.

TABLE I
THE COMPARISION OF AUC VALUES BETWEEN THE PROPOSED AND THE OTHERS.

Video	SUN [17]	Bayes [9]	Seo [18]	Hou [19]	Itti [7]	Our
HallMonit	0.7071	0.7999	0.8393	0.7912	0.7839	0.8836
FOREMAN	0.5285	0.6905	0.5696	0.6543	0.5095	0.8832
HARBOUR	0.5236	0.5773	0.4273	0.4832	0.4844	0.6953
bus	0.7309	0.7202	0.7126	0.7462	0.6895	0.7256
Flower	0.4959	0.5166	0.5531	0.5262	0.6426	0.5395
BQmall	0.7129	0.7317	0.6652	0.7032	0.727	0.6889
BQSquare	0.4693	0.5291	0.4935	0.5423	0.66	0.6535
BasketPass	0.6452	0.7905	0.6748	0.7252	0.7169	0.7605
BasketDrill	0.5815	0.7025	0.6281	0.639	0.7141	0.7009
Johnny	0.7532	0.8813	0.7915	0.8594	0.6115	0.9374
FourPeople	0.735	0.6758	0.7323	0.8095	0.6928	0.8556
SlideEditing	0.5954	0.8559	0.6491	0.6956	0.6802	0.8506
SlideShow	0.788	0.7892	0.7296	0.7284	0.6332	0.8101
Kristen	0.8193	0.8163	0.837	0.8643	0.8359	0.9419
Cactus	0.7158	0.7584	0.7256	0.7566	0.6414	0.7699
Average	**0.6534**	**0.7223**	**0.6686**	**0.7016**	**0.6682**	**0.7798**

TABLE II
THE COMPARISION OF KL DIVERGENCE BETWEEN THE PROPOSED AND THE OTHERS.

Video	SUN [17]	Bayes [9]	Seo [18]	Hou [19]	Itti [7]	Our
HallMonit	1.8691	1.6449	1.4449	1.7159	1.5896	1.4157
FOREMAN	2.3997	2.3633	3.3546	2.3962	2.5121	1.8365
HARBOUR	1.673	1.9442	3.0244	1.8395	1.9248	1.6013
bus	1.7545	3.7337	2.2277	1.824	1.9191	1.9299
Flower	1.9633	5.3404	2.7645	1.9258	1.6062	1.9455
Bqmall	1.9227	3.0857	2.0036	1.86	1.7726	1.929
BQSquare	1.6783	2.1565	2.7102	1.4812	1.2892	1.2794
BasketballPass	1.5692	1.9868	3.2289	1.4113	1.5118	1.4022
BasketballDrill	2.1652	2.1419	2.8392	2.0602	1.9995	1.9363
Johnny	3.4402	3.0371	3.0761	3.0933	3.6574	2.8418
FourPeople	2.994	3.2652	2.8867	2.8302	3.1014	2.4501
SlideEditing	3.3471	2.4108	3.5077	2.8534	2.9414	2.665
SlideShow	5.9667	3.8064	4.8856	2.6344	3.0069	2.2952
KristeN	3.0774	3.0991	2.7087	2.9077	3.0863	2.6845
Cactus	3.2634	3.3286	3.102	3.1354	3.4071	3.1605
Average	**2.6056**	**2.8896**	**2.9177**	**2.2646**	**2.3550**	**2.0915**

III. EXPERIMENT RESULT AND DISCUSSION

A. Setting on Experiment

We implement the proposed algorithm into HEVC test model HM-16.0. Fifteen different videos including CIF (352×288), 240P (416 × 240), 480P (832×480), 720P (1280×720) and 1080P (1920×1080) sequences (each with 300 frames) are chosen to evaluate the performance of proposed saliency model. These videos are selected from the database SFU [20] and Xu et al. [16]. The common $lowdelay\_main$ configuration file of HM is used in experiment.

The experiments are performed to compare the performance of proposed video saliency detection model with other five state-of-the-art methods, ie., SUN [17], Bayes [9], Seo [18], Hou [19] and Itti [7]. KullbackLeibler (KL) divergence, Receiver operating characteristic (ROC) curve and Area under the curve (AUC) values [21] are used to evaluate saliency detection accuracy. The larger the AUC is, the better the saliency predication for the saliency detection model is for video frames. A lower KL divergence of a saliency map indicates a better approximation of the ground truth.

TABLE III
THE COMPARISION OF COMPUTATIONAL TIME BETWEEN THE PROPOSED AND THE OTHER METHODS.

	SUN	Bayes	Seo	Hou	Itti	Our
Time(s)	1.6	18.7	40.6	1.8	0.12	2.3

B. Experiment

Firstly, the common $lowdelay\_main$ configuration file of HM is used in this experiment. In Table I, the AUC values of six methods are compared. The average AUC value of our method is 0.7798. Among the methods of SUN [17], Bayes [9], Seo [18], Hou [19] and Itti [7], the average AUC values of Bayes [9] is the highest, with 0.7223. The average of our method is 0.05 higher than Bayes [9].

Next, the KL divergences of six methods are compared in Table II. The average KL divergence of our method is 2.0915. Among the methods of SUN [17], Bayes [9], Seo [18], Hou [19] and Itti [7], the average KL divergence of Hou [19] is the lowest, with 2.2646. The average of our method is 0.17 lower than Hou [19]. Therefore, our method has the best performance

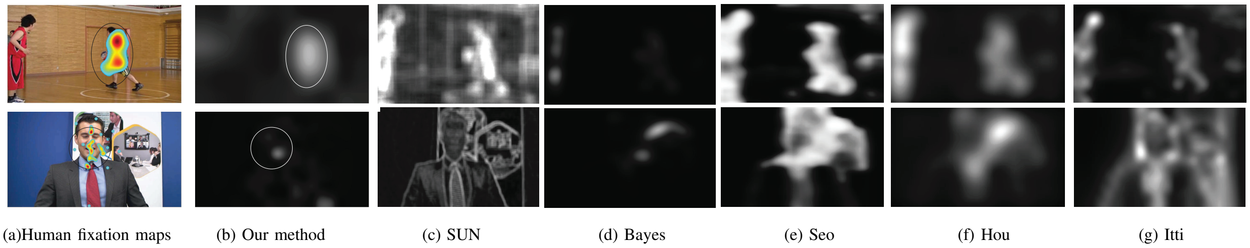


Fig. 3. Comparison of saliency maps from different models.

among the six methods.

Then, the computational time of our and other methods have been recoreded and list in Table III. Our method is slower than SUN [17], Hou [19] and Itti [7]. The quality of models can be effectively evaluated by time. However, as discussed above, the performance of these methods is worse than our method. The accuracy of other method is lower than our method. In summary, our method has higher accuracy in relatively less time.

Finally, several frames are selected as an instance. The results are shown in Fig.3. In Fig.3, (a) is the Human fixation maps; (b) to (f) are saliency maps of our method, SUN [17], Bayes [9], Seo [18], Hou [19] and Itti [7], respective. The areas marked in our experiment results are closer to human fixation map in all methods, so our proposed method has the best performance among six methods.

IV. CONCLUSION

In this paper, we propose a video saliency detection in compressed HEVC domain based on static saliency map and dynamic saliency map. The static saliency map is calculated by the static features, such as luminance, chroma and texture. These features are extracted in YCbCr color components. The dynamic saliency map is calculated by the HEVC features, such as CU depth, MV, and bit allocation. These features are extracted in video bitstream. We establish an adaptive fusion model based on competition to combine the static saliency and the dynamic saliency maps, then the final saliency map can be obtained. The parameters of each part of the fusion algorithm are adjustable. Finally, we calculate the KL divergence, ROC curve and AUC. The experiment results show that the performance of the proposed model is the best among these compared models.

ACKNOWLEDGMENT

This work was supported in part by the National Natural Science Foundation of China (61602383, 61772424 and 61702418), the Fundamental Natural Science Research Funds of Shaanxi Province (2018JQ6016 and 2017JQ6019)

REFERENCES

[1] E Matin, “Saccadic suppression: a review and an analysis.,” *Psychological Bulletin*, vol. 81, no. 12, pp. 899–917, 1974.

[2] N. J. Butko and J. R. Movellan, “Optimal scanning for faster object detection,” in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, June 2009, pp. 2751–2758.

[3] D. Gao, S. Han, and N. Vasconcelos, “Discriminant saliency, the detection of suspicious coincidences, and applications to visual recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 6, pp. 989–1005, June 2009.

[4] Michael Rubinstein, Diego Gutierrez, Ariel Shamir, and Ariel Shamir, “A comparative study of image retargeting,” in *ACM SIGGRAPH Asia*, 2010, p. 160.

[5] Zhicheng Li, Shiyin Qin, and Laurent Itti, “Visual attention guided bit allocation in video compression,” *Image & Vision Computing*, vol. 29, no. 1, pp. 1–14, 2011.

[6] Christof Koch and Shimon Ullman, “Shifts in selective visual attention: Towards the underlying neural circuitry,” *Hum Neurobiol*, vol. 4, no. 4, pp. 219–227, 1987.

[7] L. Itti, C. Koch, and E. Niebur, “A model of saliency-based visual attention for rapid scene analysis,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, Nov 1998.

[8] Bernhard Sch?lkopf, John Platt, and Thomas Hofmann, *Graph-Based Visual Saliency*, pp. 545–552, MIT Press, 2007.

[9] Laurent Itti and Pierre Baldi, “Bayesian surprise attracts human attention,” *Vision Research*, vol. 49, no. 10, pp. 1295–1306, 2009.

[10] Xiaodi Hou and Liqing Zhang, “Saliency detection: A spectral residual approach,” in *Computer Vision and Pattern Recognition, 2007. CVPR ’07. IEEE Conference on*, 2007, pp. 1–8.

[11] Luca Marchesotti, Claudio Cifarelli, and Gabriela Csurka, “A framework for visual saliency detection with applications to image thumbnailing,” in *IEEE International Conference on Computer Vision*, 2009, pp. 2232–2239.

[12] C. Guo and L. Zhang, “A novel multiresolution spatiotemporal saliency detection model and its applications in image and video compression,” *IEEE Transactions on Image Processing*, vol. 19, no. 1, pp. 185–198, Jan 2010.

[13] Yun Zhai and Mubarak Shah, “Visual attention detection in video sequences using spatiotemporal cues,” in *ACM International Conference on Multimedia*, 2006, pp. 815–824.

[14] Yuming Fang, Zhenzhong Chen, Weisi Lin, and Chia Wen Lin, “Saliency-based image retargeting in the compressed domain,” in *ACM International Conference on Multimedia*, 2011, pp. 1049–1052.

[15] Y. Fang, W. Lin, Z. Chen, C. M. Tsai, and C. W. Lin, “A video saliency detection model in compressed domain,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 1, pp. 27–38, Jan 2014.

[16] M. Xu, L. Jiang, X. Sun, Z. Ye, and Z. Wang, “Learning to detect video saliency with hevc features,” *IEEE Transactions on Image Processing*, vol. 26, no. 1, pp. 369–385, Jan 2017.

[17] L. Zhang, M. H. Tong, T. K. Marks, H. Shan, and G. W. Cottrell, “Sun: A bayesian framework for saliency using natural statistics,” *J Vis*, vol. 8, no. 7, pp. 32.1, 2008.

[18] H. J. Seo and P Milanfar, “Static and space-time visual saliency detection by self-resemblance.,” *J Vis*, vol. 9, no. 12, pp. 1–27, 2009.

[19] X. Hou, J. Harel, and C. Koch, “Image signature: Highlighting sparse salient regions,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 1, pp. 194–201, Jan 2012.

[20] H. Hadizadeh, M. J. Enriquez, and I. V. Bajic, “Eye-tracking database for a set of standard video sequences,” *IEEE Transactions on Image Processing*, vol. 21, no. 2, pp. 898–903, Feb 2012.

[21] Z. Bylinskii, T. Judd, A. Oliva, A. Torralba, and F. Durand, “What do different evaluation metrics tell us about saliency models?,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2018.