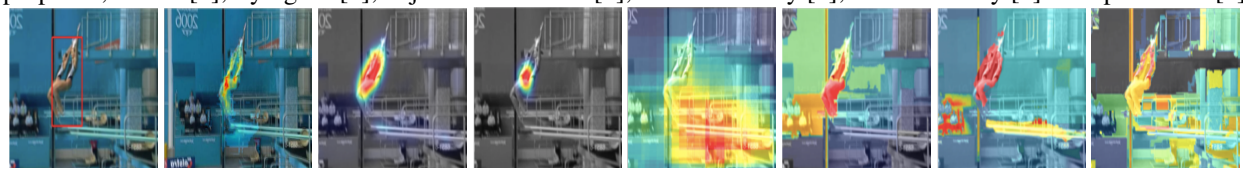
ANISOTROPIC PARTIAL DIFFERENTIAL EQUATION BASED VIDEO SALIENCY DETECTION

Wai Lam Hoo *Chee Seng Chan*

Center of Image and Signal Processing, Faculty of Computer Science and Info. Tech.,
University of Malaya, Kuala Lumpur, Malaysia
{*wlhoo; cs.chan*}*@um.edu.my*

Fig. 1. UCF Sports Dataset: Comparison between our proposed and conventional methods. From left to right: Groundtruth; Our proposed; EBSD [1]; Eye-gaze [2]; objectness detector [3]; colour saliency [4]; video saliency [5] and space-time [6]



ABSTRACT

In this paper, we propose a novel video saliency detection method using the Partial Differential Equations (PDEs). We first form a static adaptive anisotropic PDE model from the unpredicted frames of the video using a detection map and a saliency seeds set of most attractive image elements. At the same time, we also extract motion features from the predicted frames of the video to generate motion saliency map. Then, we combine these two maps to obtain the final saliency map (video). Experiments on various human-action datasets show that our video saliency detection model performs favourably against the conventional solutions.

Index Terms— saliency detection, partial differential equation

1. INTRODUCTION

Saliency detection is an attention mechanism that focuses on limited perceptual and cognitive regions, and thus eases out the process of carrying further tasks to analyse and understand images or videos. The saliency detection is closely related to how humans perceive the visual stimuli and therefore results in a saliency map, where each pixel value or pixel score describes how it stands out from its surrounding neighbourhood.

In this paper, our work is focused on video saliency detection using Partial Differential Equations (PDEs) as illustrated in Fig. 2. Compared with image saliency detection, video saliency detection algorithms have to calculate the motion saliency map since motion is an essential factor to attract human attention. There has been numerous saliency detection models for video saliency detection[1]. For instance, [1, 8] proposed a video saliency detection based on the features extracted from the DCT coefficients. Zhang *et al*. [9] model the saliency detection as a manifold ranking on a graph problem. Kim *et al*. [10] proposed random walk with restart to detect spatially and temporally salient regions. Wang *et al*. [11] proposed a late-fusion strategy to combine state-of-the-art visual saliency detections using confidence scores. With the advancement of deep learning solutions, [12–14] employed different convolutional models to study video saliency.

At the same time, PDEs have been used successfully in many low-level image processing tasks such as image denoising, inpanting etc [15], and recently in more complex tasks such as image saliency detection [16]. It is based on the linear elliptic system with Dirichlet boundary (LESD) model because traditional PDEs with fixed formulation and boundary condition could not efficiently quantify and explain complex visual saliency patterns, thus it will fail to solve the saliency detection problem. However, the work is not suitable for video saliency detection task as the original LESD model (i) does not consider the orientation and motion information contain in the video; and (ii) uses the center-prior.

The main contributions of this paper are as follows. First, we propose a novel method to generate the static saliency map based on the adaptive nonlinear PDEs model [16] with refinements. Particularly, we extend our model to flow-like structure so that it can rotates the PDEs flow towards the orientation of interesting features. Secondly, we do not use center-prior. Rather, an extensive direction map consists of background, color, texture and luminance prior are employed.

[1]For a more comprehensive literature, we refer interested readers to [7]

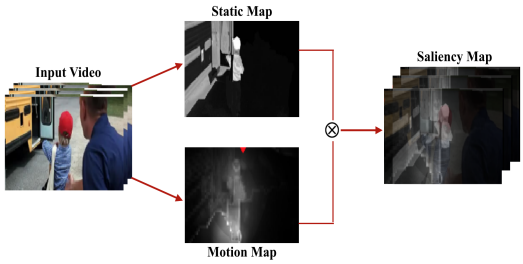


Fig. 2. Pipeline of our proposed non-linear PDE based video saliency detection method.

This is because most of the video datasets contain heavy noise and the salient object is usually moving within the frames, as opposed to images that are mostly noiseless and objects appearance are nearer to the image center. Finally, experiments on various human action datasets show that our proposed model performs favourably against the conventional methods (Section 3 and Fig. 4).

2. PROPOSED METHOD

Fig. 2 depicts the proposed framework. There are two main parallel sub-processes: static saliency and motion maps where these two maps are fused to derive the final saliency map.

2.1. Formulation of Static Saliency Map with PDEs

Let I be the unpredicted video frame. $I(x, y)$ is a set of all points corresponds to the video frame (*i.e.* pixels or superpixels). Let $u(x, y, t) : I \rightarrow R$ is the intensity of a frame with diffusion time t, for the image domain $I \in R \times R$. Also consider, $u(x, y, i)$ is the score function to measure the saliency of each pixel i in the domain $I(x, y)$. Therefore, given a set of saliency seeds S and their corresponding score function $u(x, y, i) = u_0, i \epsilon S$, we can mathematically formulate saliency detection as an evolutional PDEs with initial condition such as:

$$\begin{aligned} \frac{\partial u_i}{\partial t} &= G(u, \bigtriangledown u, |\bigtriangledown u|), \text{ on } I \times (0, \infty) \\ u(x, y, 0) &= u_0, \text{ on } I \\ \partial u_n &= 0, \text{ on } I \times (0, \infty) \end{aligned} \tag{1}$$

where $\frac{\partial u_i}{\partial t}$ denotes the first derivative with respect to the diffusion time t and G is a function of u. The function G as a diffusion term $div(g(|\bigtriangledown u|) \bigtriangledown u)$, is a nonlinear PDE with initial conditions given in Eq. 1. However, it does not give a reliable information in the presence of flow-like structures. To this end, we select the structure tensor, also referred as the second moment matrix to rotate the flow towards the orientation of interesting features. For a multivalued image, the structure tensor has the following form:

$$\begin{aligned} S_\sigma &= (\sum_{i=1}^{n} \bigtriangledown u_{i\sigma} \bigtriangledown u_{i\sigma}^T) \\ &= \begin{bmatrix} \sum_{i=1}^{n} u_{ix\sigma}^2 & \sum_{i=1}^{n} u_{ix\sigma} u_{iy\sigma} \\ \sum_{i=1}^{n} u_{ix\sigma} u_{iy\sigma} & \sum_{i=1}^{n} u_{iy\sigma}^2 \end{bmatrix}, \end{aligned} \tag{2}$$

With $\bigtriangledown u_{i\sigma} = K_\sigma * \bigtriangledown u_i = K_\sigma * (u_{ix}, u_{iy})$, the smoothed version of the gradient can be obtained by a convolution with a Gaussian kernel K_σ. The structure scale σ determines the size of the resulting flow-like patterns where increasing the σ gives an increased distance between the resulting flow lines. The advantages of these new gradient features allow a more precise description of the local gradient characteristics, and it's smoothed version of S_σ, can be represented as:

$$J_\rho = K_\rho * S_\sigma = \begin{bmatrix} j_{11} & j_{12} \\ j_{21} & j_{22} \end{bmatrix}, \tag{3}$$

where K_ρ is a Gaussian kernel with standard deviation σ. The integration scale ρ averages the orientation information, and so it helps to stabilize the directional behavior of the filter. In summary, the convolution with the Gaussian kernels K_ρ and K_σ make the structure tensor measure more coherent. Mathematically, the structure tensor J_ρ can be written over its eigenvalues (λ_+, λ_-) and eigenvectors (Θ_+, Θ_-), that is $J_\rho = \lambda_+ \theta_+ {\theta_+}^T + \lambda_- \theta_- {\theta_-}^T$ as shown in [17]. The eigenvectors of J_ρ give the preferred local orientations, while the corresponding eigenvalues denote the local contrast along these directions.

In order to create a truly anisotropic scheme, [17] proposed a nonlinear diffusion tensor, replacing the diffusivity function $G(\cdot)$ in Eq. 1 with the combination of two types of novel tensors as follows: one allows diffusion along the orientation of greatest coherence, while the other allows diffusion along orthogonal directions. That is,

$$\frac{\partial u_i}{\partial t} = div((K_1(J_\rho) + \alpha K_2(J_\rho)) \bigtriangledown u_i), \tag{4}$$

where $i = 1, \cdots, n$ on I; The parameter α is a regularization term to ensure the compromise between two tensors. K_1 is defined to control the local diffusivity at i, and is constructed from the local coordinate system, *i.e.*, for a neighborhood set of i, $N_i = \{j_1, j_2, \cdots, j_n\}$. $K_1 = K_1(J\rho)$ is defined as $K_1 = diag(k(i, j_1), k(i, j_2), \cdots, k(i, j_n))$, where, $k(i, j)$ is the Gaussian similarity between the nodes. $K_2(J_\rho)$ is the structure tensor and this tensor posses the eigenvectors θ_-, θ_+ as the structure tensor J_ρ and uses the eigenvalues λ_1, λ_2, to control the diffusion speeds in these two directions, that is $K_2 = K_2(J_\rho) = \lambda_1 \theta_+ {\theta_+}^T + \lambda_2 \theta_- {\theta_-}^T$.

To incorporate the high-level prior into the diffusion process, another regularization term, $\eta(.)$ is introduced such that:

$$\begin{aligned} \frac{\partial u_i}{\partial t} &= div((K_1(J_\rho) + \alpha K_2(J_\rho)) \bigtriangledown u_i) \\ &\quad + \eta(u(i) - d(i)), i \epsilon S, \end{aligned} \tag{5}$$

where $d(i)$ is the guidance map and $\eta \geq 0$ is a balance parameter. In this paper, we only consider the situation when the saliency evolution is stable and so only find a solution to the the following PDE[2]:

$$\begin{aligned} \frac{\partial u_i}{\partial t} &= 0, \text{ on } I \times (0, \infty) \\ u(x, y, 0) &= u_0, \text{ on } I \\ \partial u_n &= 0, \text{ on } I \times (0, \infty), i \epsilon S. \end{aligned} \tag{6}$$

Therefore, given a video frame, the saliency detection task grounds to the problem of solving Eq. 6 to achieve a stable state for visual attention diffusion. Note that, Eq. 5 can be solved numerically using the finite differences scheme [18]). As such, the time-derivative $\frac{\partial u_i}{\partial t}$ at (x, y, t_n) can be approximated by the forward difference $\frac{\partial u_i}{\partial t} = \frac{(u_i^{n+1} - u_i^n)}{\Delta t}$, which leads to the iterative scheme:

$$\begin{aligned} u_i^{n+1} &= u_i^n + \Delta t \, div((K_1(J_\rho) + \alpha K_2(J_\rho)) \bigtriangledown u_i^n) \\ &\quad + \eta(u(i) - d(i)), i \epsilon S \end{aligned} \tag{7}$$

2.2. Static Map

In previous section, we have formulated the anisotropic PDEs system for saliency diffusion. This section will explain how we incorporate the direction map d and saliency seed set S in Eq. 7. As aforementioned, we do not use center-prior as to [16]. Rather, we extract the color prior, texture and luminance features knowledge using the discrete cosine transform (DCT) coeffiencts of video frames to build the direction map d. The $YCrCb$ color space is used to encode the given video frames. Here, we first transfer the DC coefficients from $YCrCb$ color space to the RGB color space to extract the luminance and color features of the video frames. We calculate the color and luminance features L from DCT coefficients by generating four broadly-tuned color channels: $R = r - (g + b)/2; G = g - (r + b)/2; B = b - (r + g)/2$ and $Y = \frac{(r+g)}{2} - \frac{|r-b|}{2} - b$, where r, g, b denote the red, green and blue color components from the DC coefficients and R, G, B, Y denote the new red, new green, new blue and new yellow components, respectively. So the colour feature can be calculated as $C_{rg} = R - G$ and $C_{by} = B - Y$, where C_{rg}, and C_{by} are the color features for an 8×8 block in the video frame. For texture, we only use the AC coefficients from the Y component to extract the texture feature Te as AC coefficients of Cr and Cb components provide little information on the texture [19, 20].

2.2.1. Direction Map

For diffusion based saliency, we partition I into superpixels using SLIC [21] and obtain the superpixel set $V = \{s_1, s_2, \cdots, s_{|v|}\}$. That is, we divide the image into two parts to obtain the salient region: foreground F_c, which contains the salient object including some background regions and pure background B_c, which is the non-salient region using the convex hull. The foreground is obtained by collecting nodes inside the bulged out convex curve C and the remaining nodes will serve as the background nodes. Then, we map these nodes to a graph $G = (V, E)$ with superpixels as nodes. We formulate a simplified non-linear PDE to compute the background diffusion score, with $\eta = 0$ in Eq. 5 to get the probability of background score. Foreground score can be calculated as $u_f(i) = 1 - u_b(i)$. The final direction map is defined as $d(i) = L \times C_{rg} \times C_{by} \times Te \times u_f(i)$.

2.2.2. Saliency Seed Set

Saliency seed set S is obtained using optimizing saliency seeds via submodularity. Since, not all the nodes in F_c can be used as saliency seeds, it is very important to have the set of most representative seeds in the foreground. For this purpose, we maximise the sum of score function u with respect to all the superpixels in V when the saliency diffusion is stable. To this end, we solve the discrete optimization problem that is based on the approximation of discretize PDE formulation as:

$$\begin{aligned} &\max_{S \in M^n} L(S), s.t. \\ u(i) &= \frac{1}{d_i + \eta}\Big(\sum_{j \in N(i)} (K_1(j) + \alpha K_2(j)) u(j) \\ &\quad + \eta d(i)\Big) \\ u(i) &= s_i, i \in S, \end{aligned}$$

where $d_i = \sum_{j \in N(i)} (K_1(j) + \alpha K_2(j))$, $L(S) = \sum_{i \in V} u(i; S)$ and $M^n = \{S | S \subset F_c, |S| \leq n\}$, is an uniform matroid to enforce that the cardinality of S is no more than n, maximum number of saliency seeds.

2.3. Motion Map

The motion map is generated from the motion features of the predicted (P and B) frames using the motion vectors from the video bitstream. As there is only one prediction direction for P frame, the original motion vector MV is used to calculate the motion feature in the P frames. For bidirectional frames, we calculate the sum of both past (MV_p) and future (MV_f) vector frames as $V = MV_p + (-1) * MV_f$. The motion feature of each DCT block in the B frames is obtained from V, while the original motion vector is used to represent the motion feature for each DCT block in the P frames.

2.4. Final Saliency Map

Since unpredicted frames contain no motion features, the motion saliency map of the previous predicted frame is adopted to represent the motion saliency of the current unpredicted frame. Thus, the final saliency map for unpredicted frames

[2]The time parameter t is omitted at this stage in the notation.

Fig. 3. Sample video saliency maps obtained from our algorithm tested on three different datasets. (top) Weizmann dataset [22], (middle) Hollywood dataset [23] and (bottom) UCF sports dataset [24].

becomes $S_{un} = \psi(S_s, S_{m_p})$ where S_s is the static saliency map of the unpredicted frame, and S_{m_p} is the motion saliency map of the previous frame where ψ is a hybrid function. Similarly, since the predicted frames have no static features, static saliency map of the previous unpredicted frame is used to represent the static saliency map of current predicted frame. Therefore, for predicted frames, the final saliency map becomes $S_{re} = \psi(S_{s_p}, S_m)$ where S_{s_p} is the static saliency frame of previous unpredicted frame and S_m is the motion map of current predicted frame. The final (video) saliency map is represented as $S = S_{un} \otimes S_{re}$.

3. EXPERIMENTS

Datasets. We test the performance of proposed method on Weizmann [22], Hollywood [23] and UCF Sports [24] datasets. All these datasets depict challenging scenarios including camera motion, cluttered backgrounds, and non-rigid object deformations. We set the PDE model's parameters to: $\alpha = 0.7, \sigma = 1.35, \rho = 4.05, \eta = 0.5$.

Results. Fig. 3 shows the exemplar of video saliency maps obtained from our algorithm on Weizmann [22], Hollywood [23] and UCF Sports [24] datasets. It shows that our method is able to learn the dynamic saliency information and detects salient moving objects accurately. For the computational complexity, we conducted the experiment on both CPU and GPU machines. The average computational time of our proposed method for each video frame on the CPU[3] is 4:89sec; while on the GPU[4] is 0.25sec.

Comparison. Fig. 1 and 4 show a comparison of our proposed method with a few conventional algorithms in the UCF sports dataset. It is observed that the eye-gaze method [2] and encoding-based saliency detection (ESBD) [1], can only identify center surround differences but miss most of the object information. [3–5] performed badly as the low-rank assumption may be invalid when image contain complex structures. Meanwhile, our proposed method successfully highlights the salient region more uniformly.

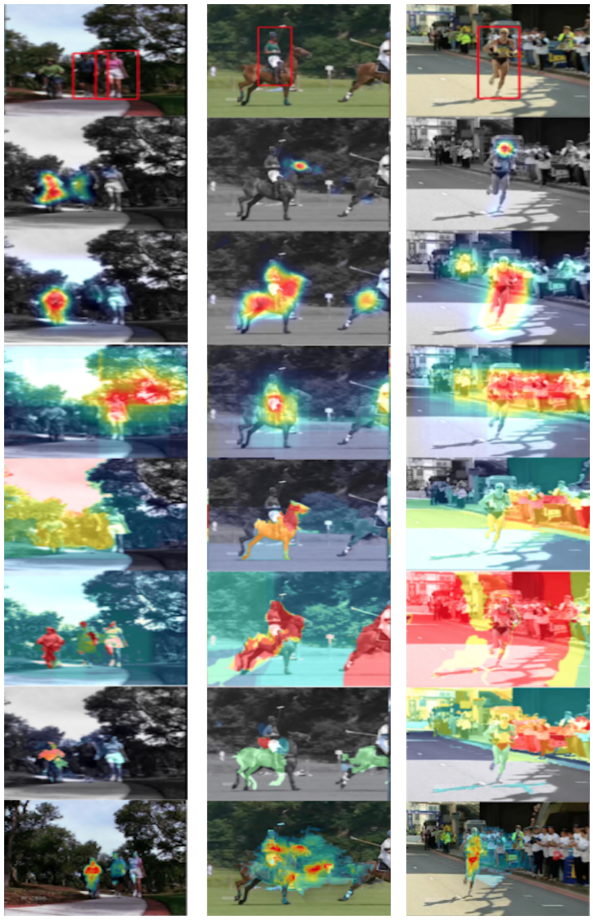


Fig. 4. Qualitative comparisons on UCF sports dataset [24]. (Top row): Input images with ground-truth annotations. (2^{nd} - 7^{th} row): Eye-gaze tracking [2]; EBSD [1]; objectness detector [3]; colour saliency [4]; video saliency [5] and space-time saliency [6]. (Final row): Our proposed model.

4. CONCLUSION

This paper proposes a novel video saliency detection method inspired by PDEs. Particularly, we introduce a novel method to generate static saliency map based on the adaptive nonlinear PDEs model. Experimental results had shown the effectiveness of the proposed method in three public human action datasets when compared to the conventional solutions.

[3]Intel Xeon(R) ES-2609, 2.50GHZ
[4]Nvidia GeForce GTX Titan Z

5. REFERENCES

[1] T. Mauthner, H. Possegger, G. Waltner, and H. Bischof, "Encoding based saliency detection for videos and images," in *CVPR*, 2015, pp. 2494–2502.

[2] S. Mathe and C. Sminchisescu, "Dynamic eye movement datasets and learnt saliency models for visual action recognition," in *ECCV*, pp. 842–856. 2012.

[3] B. Alexe, T. Deselaers, and V. Ferrari, "What is an object?," in *CVPR*, 2010, pp. 73–80.

[4] H. Jiang, J. Wang, Z. Yuan, T. Liu, N. Zheng, and S. Li, "Automatic salient object segmentation based on context and shape prior.," in *BMVC*, 2011, vol. 6, p. 9.

[5] E. Rahtu, J. Kannala, M. Salo, and J. Heikkilä, "Segmenting salient objects from images and videos," in *ECCV*, pp. 366–379. 2010.

[6] F. Zhou, S. Kang, and M. Cohen, "Time-mapping using space-time saliency," in *CVPR*, 2014, pp. 3358–3365.

[7] W. Wang, J. Shen, F. Guo, M. Cheng, and A Borji, "Revisiting video saliency: A large-scale benchmark and a new model," *arXiv preprint arXiv:1801.07424*, 2018.

[8] Y. Fang, W. Lin, C. Chen, Z.and Tsai, and C. Lin, "A video saliency detection model in compressed domain," *TCSVT*, vol. 24, no. 1, pp. 27–38, 2014.

[9] D. Zhang, D. Meng, and J. Han, "Co-saliency detection via a self-paced multiple-instance learning framework," *TPAMI*, vol. 39, no. 5, pp. 865–878, 2017.

[10] H. Kim, Y. Kim, J. Sim, and C. Kim, "Spatiotemporal saliency detection for video sequences based on random walk with restart," *TIP*, vol. 24, no. 8, pp. 2552–2564, 2015.

[11] J. Wang, A. Borji, C-C J. Kuo, and L. Itti, "Learning a combined model of visual saliency for fixation prediction," *TIP*, vol. 25, no. 4, pp. 1566–1579, 2016.

[12] W. Wang, J. Shen, and L. Shao, "Video salient object detection via fully convolutional networks," *TIP*, vol. 27, no. 1, pp. 38–49, 2018.

[13] Çağdaş Bak, Aykut Erdem, and Erkut Erdem, "Two-stream convolutional networks for dynamic saliency prediction," *arXiv preprint arXiv:1607.04730*, 2016.

[14] G. Li and Y. Yu, "Visual saliency based on multiscale deep features," *arXiv preprint arXiv:1503.08663*, 2015.

[15] D. Tschumperle and R. Deriche, "Vector-valued image regularization with pdes: A common framework for different applications," *PAMI*, vol. 27, no. 4, pp. 506–517, 2005.

[16] R. Liu, J. Cao, Z. Lin, and S. Shan, "Adaptive partial differential equation learning for visual saliency detection," in *CVPR*, 2014, pp. 3866–3873.

[17] F. Benzarti and H. Amiri, "Image denoising using non linear diffusion tensors," *Advances in Computing*, vol. 2, no. 1, pp. 12–16, 2012.

[18] J. Weickert and H. Scharr, "A scheme for coherence-enhancing diffusion filtering with optimized rotation invariance," *Journal of Visual Comm. and Image Representation*, vol. 13, no. 1, pp. 103–118, 2002.

[19] C. Theoharatos, V. K. Pothos, N. A. Laskaris, G. Economou, and S. Fotopoulos, "Multivariate image similarity in the compressed domain using statistical graph matching," *PR*, vol. 39, no. 10, pp. 1892–1904, 2006.

[20] Y. Zhong and A. K. Jain, "Object localization using color, texture and shape," *PR*, vol. 33, no. 4, pp. 671–684, 2000.

[21] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," *TPAMI*, vol. 34, no. 11, pp. 2274–2282, 2012.

[22] L. Gorelick, M. Blank, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," *TPAMI*, vol. 29, no. 12, pp. 2247–2253, 2007.

[23] M. Marszałek, I. Laptev, and C. Schmid, "Actions in context," in *CVPR*, 2009.

[24] M. D. Rodriguez, J. Ahmed, and M. Shah, "Action mach a spatio-temporal maximum average correlation height filter for action recognition," in *CVPR*, 2008, pp. 1–8.