Pattern Recognition 97 (2020) 106998

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

Video salient object detection using a virtual border and guided filter

Qiong Wang[a,b,*], Lu Zhang[b,*], Wenbin Zou[c], Kidiyo Kpalma[b]

[a] College of Computer Science and Technology, Zhejiang University of Technology, No.288 Road Liuhe, Hangzhou 310023, China
[b] Univ Rennes, INSA Rennes, CNRS, IETR (Institut d'Electronique et de Télécommunication de Rennes) - UMR 6164, Rennes F-35000, France
[c] College of Electronic and Information Engineering, Shenzhen Key Laboratory of Advanced Machine Learning and Applications, Guangdong Key Laboratory of Intelligent Information Processing, Shenzhen University, Shenzhen 518060, China

ARTICLE INFO

Article history:
Received 21 December 2017
Revised 31 May 2019
Accepted 7 August 2019
Available online 7 August 2019

Keywords:
Video salient object detection
Distance transform
Guided filter
Global motion

ABSTRACT

In this paper, we present a novel method for salient object detection in videos. Salient object detection methods based on background prior may miss salient region when the salient object touches the frame borders. To solve this problem, we propose to detect the whole salient object via the adjunction of virtual borders. A guided filter is then applied on the temporal output to integrate the spatial edge information for a better detection of the salient object edges. At last, a global spatio-temporal saliency map is obtained by combining the spatial saliency map and the temporal saliency map together according to the entropy. The proposed method is assessed on three popular datasets (Fukuchi, FBMS and VOS) and compared to several state-of-the-art methods. The experimental results show that the proposed approach outperforms the tested methods.

© 2019 Elsevier Ltd. All rights reserved.

1. Introduction

The human vision system has an effective ability to easily recognize interesting regions from complex scenes, even if the focused regions have similar colors or shapes as the background. Salient object detection aims to detect the salient object that attracts the most the visual attention. The output of the salient object detection is a saliency map where the pixel values indicate the probability of each pixel of belonging to the salient object. Higher value represents higher saliency. This topic has gained much attention for its wide applications, such as image registration [1,2], object segmentation [3,4], person identification [5], spectral-spatial reconstruction [6] and etc.

Existing salient object detection methods can be roughly divided into two categories: traditional methods and deep learning-based methods, which are interesting and useful for different applications. For a given database, deep learning-based methods have a better performance than many recent traditional methods. But the premise is it should be trained with huge and rich training datasets, which is impossible for some applications where the available data is small. Traditional methods are however intrinsically unassailable from such limitation. In this study, we will focus on the traditional approach, but we will also show how the performance of our proposed model can be further improved by integrating a deep learning-based method.

According to the type of source information, salient object detection approaches can be broadly grouped into two categories: image salient object detection models and video salient object detection models. Image salient object detection models the visual input viewing process based on the appearance of the scene. Since the human vision system is sensitive to motions, video salient object detection detects the salient object using cues in both spatial domain and temporal domain and becomes much popular. However, due to the limitation of leverage of the saliency cues from two domains, video salient object detection is still challenging. In this paper, we focus on video salient object detection.

The "background prior" [7] assumption is widely used in salient object detection approaches. It assumes that a narrow border of the image is the background region. This assumption is normally true because the important object is often located in the frame center by the photographers. Based on this assumption, the distance transform has been widely used for saliency computation. Traditionally, the distance transforms measure the distance of a pixel and the seed set using different path cost functions. Since background regions are assumed to be connected to image borders, the border pixels are initialized as the seed set and the distance transform detects a pixel's saliency by computing the shortest path from the pixel to the seed. The larger the shortest path is, the higher the saliency is. It has achieved a success in salient object detection, but a few commonly observable issues still exist. In the background prior, all the border pixels are regarded as

* Corresponding authors at: College of Computer Science and Technology, Zhejiang University of Technology, No.288 Road Liuhe, Hangzhou 310023, China.
E-mail addresses: wangqiong819@gmail.com (Q. Wang), lu.ge@insa-rennes.fr (L. Zhang).

https://doi.org/10.1016/j.patcog.2019.106998
0031-3203/© 2019 Elsevier Ltd. All rights reserved.

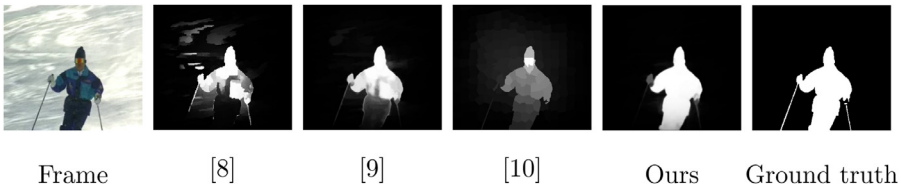

Fig. 1. State-of-the-art saliency maps [8–10].

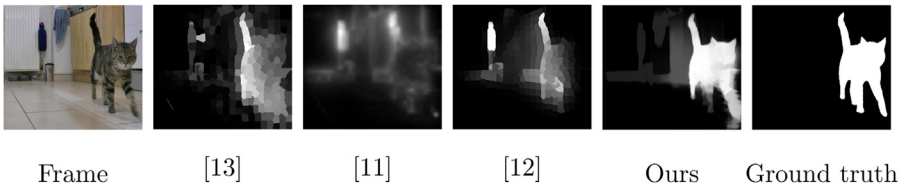

Fig. 2. State-of-the-art saliency maps [11–13].

background. Thus, in the distance transform, all the border pixels are set to be seed and their saliency values are thus zeros. When the salient object pixels appear in the border, their saliency values are consequently set to zeros. Though some methods [8–10] can alleviate this problem, but not enough. Fig. 1 illustrates this problem by showing the saliency maps of some existing methods on one example image.

Video salient object detection detects the salient object from both spatial domain and temporal domain. How to combine these two saliencies together during the detection is complex. One usual way (called "Feature fusion") is to fuse the extracted spatial feature and extracted temporal feature together to give a spatio-temporal feature. Considering the spatial gradient magnitudes and fusing them with the temporal gradient magnitudes into spatio-temporal edges is a popular Feature fusion way. The resulted spatio-temporal edges may still give inaccurate salient object detection. Another usual way (called "Map fusion") is to combine the spatial saliency map and the temporal saliency map together. The existing simple linear or non-linear way is still insufficient to decide the confidence weight for each saliency map. In order to employ more video saliency information, these two techniques are used together recently. However, in complex scenes, the methods still could not fully make use of detected saliency from the two domains. Some examples are shown in Fig. 2. For models [11–13], the salient object has been located but still with blur edges. Thus, the fusion is still a much more challenging problem.

Facing these open issues, we propose a new video salient object detection algorithm by addressing:

1) the problem of detecting a complete salient object connected to borders using the distance transform with a virtual border-based technique which consists of four steps which are a) Frame Border Selection, b) Frame Border Division, c) Representative Pixel Selection and d) Virtual Border Padding. In spatial domain, the virtual border is added to the frame aiming to detect the whole salient object. In temporal domain, it is also added to the color optical flow map in order to detect the complete salient object motion and then obtain the salient object by filtering the global motion out.
2) the Feature fusion problem by using an edge-aware filter, called the guided filter [14]. It is introduced to preprocess the virtual border-based color optical flow map for enhancing object edges.
3) the Map fusion problem by computing the entropy and the standard deviations to decide the confidence level of the spatial saliency map and the temporal saliency map.

The remaining of this paper is organized as follows. Section 2 briefly describes the related work. Section 3 presents the proposed method in detail. In Section 4, we conduct comparison experiments to evaluate the performance of the proposed method. Section 5 concludes the paper.

2. Related work

This section introduces the recent works related to the video salient object detection (SOD). SOD in videos is closely related to SOD in images. Recent traditional methods for image SOD and video SOD are introduced respectively. Then, deep learning-based methods are summarized.

2.1. Traditional image SOD methods

Image SOD methods are fully exploited in recent years. We will give examples of some important categories, including graph-based approaches, probabilistic models and cognitive methods.

For graph-based approaches, Shan et al. [15] use background weight map as propagating seeds and design a third-order smoothness framework to improve the performance of manifold ranking. Jiang et al. [10] propose a saliency detection via absorbing Markovian chain. Zhang et al. [9], Tu et al. [8] and Huang et al. [16] compute the saliency based on the minimum barrier distance transform. Lie et al. [17] improve the detection speed using the upsampling of random color distance map. For probabilistic models, Aytekin et al. [18] adopt a probabilistic mass function to encode the boundary connectivity saliency cue and smoothness constraints into a global optimization problem. Li et al. [19] propose an optimization model based on conditional random fields and geodesic weighted Bayesian model. For cognitive method, Yan et al. [20] combine bottom-up and top-down attention mechanisms to focus on the salient object. Peng et al. [21] propose a tree-structured sparsity-inducing norm, and introduce a Laplacian regularization, and employ the high-level prior to detect the salient object.

2.2. Traditional video SOD methods

According to different types of spatial and temporal information to be fused, we roughly divide the traditional methods into three categories: "Feature fusion", "Map fusion" and "Hybrid fusion".

As a "Feature fusion" method, Wang et al. [12] fuses the color gradient magnitude and optical flow gradient magnitude in a non-linear way. Wang et al. [22] fuse the spatial edge to temporal op-

tical flow by using guided filter. Bhattacharya et al. [23] use a weighted sum of the sparse spatio-temporal features.

As a "Map fusion" method, Tu et al. [24] generate two types of saliency maps based on a foreground connectivity saliency measure, and exploit an adaptive fusion strategy. Yang et al. [25] propose a confidence-guided energy function to adaptively fuse spatial and temporal saliency maps.

"Hybrid fusion" can be considered as a combination of "Feature fusion" and "Map fusion". Li et al. [26] fuse the spatial and temporal channel to generate saliency maps, and then use saliency-guided stacked autoencoders to get the final saliency map. Chen et al. [27] obtain the motion saliency map with spatial cue, then use k-Nearest Neighbors-histogram based filter and Markov random field to eliminate the dynamic backgrounds. Kim et al. [11] detect the salient object based on the theory of random walk with restart. Liu et al. [13] obtain temporal saliency propagation using spatial appearance, which spatial propagation is performed via the temporal saliency map. Wang et al. [4,28] produce spatio-temporal edge map to get the saliency map based on the geodesic distance, which is then combined with global appearance models and with dynamic location models. Xi et al. [29] first get spatio-temporal background priors, and then take the sum of appearance and motion saliency as the final saliency. Zhou et al. [30] propose localized estimation to generate the temporal saliency map, and deploy the spatio-temporal refinement to get the final saliency map, which is then used to update the initial saliency map. Chen et al. [31] detect the motion cues and spatial saliency map to get the motion energy term, which are combined with some constraints and formulated into the optimization framework. Ramadan et al. [32] applies the pattern mining algorithm to detect spatio-temporal saliency patterns. Guo et al. [33] select a set of salient proposals via a ranking strategy. Chen et al. [34] get the temporal saliency map to facilitate the color saliency computation. Chen et al. [35] utilize Markov random field to conduct semantic labeling and learn multiple nonlinear feature transformations to enlarge the feature difference between the salient object and backgrounds.

2.3. Deep learning-based methods

Recently, deep neural networks are more and more used in SOD for their high efficiency and effectiveness. For image SOD, Liu et al. [36] use a hierarchical convolutional neural network to detect the object; Hou et al. [37] use deep multi-scale features instead of hand-crafted features; Lee et al. [38] combine hand-crafted features and deep features together; Chen et al. [39] learn depth cue to help saliency detection; Yuan et al. [40] propose a multiscale and multidepth network. For video SOD, Wang et al. [41] input two successive frames into the network to learn spatio-temporal saliency; Tang et al. [42] employ a weakly-supervised network without needing all training datasets with pixel-wise ground truth. Compared with the above deep learning and image-based SOD, the deep learning and video-based SOD has not been studied widely yet. This is due to the lack of the large-scale video salient object dataset and the complexity of the spatial and temporal fusion.

3. Proposed algorithm

The block-diagram of the proposed Virtual Border and Guided Filter-based (VBGF) method is shown in Fig. 3. Given an input video sequence, in spatial saliency detection (SD), the virtual border is built for each frame. Then, the saliency is computed to get the spatial saliency map (SSM). Secondly, in temporal saliency detection (TD), the motion information is extracted from the input video. Then the virtual border building, the Feature fusion and the saliency computation are applied to obtain the temporal saliency map (TSM). At last, the two saliency maps are fused to get the spatio-temporal saliency map (STSM). The method is detailed in the following parts.

3.1. Spatial saliency detection (SD)

In this section, the virtual border-based distance transform in spatial domain is designed.

3.1.1. Virtual border building

We propose to add the virtual border around the original frame to obtain with-virtual-border frame. The virtual border is built as shown in Fig. 4.

a) Frame Border Selection: one frame border is selected to build the virtual border by two steps:
   - FastMBD [9] is applied to frame α to obtain the map M.
   - The frame border nearest to the non-zero region in the map M is selected to build the virtual border.

b) Frame Border Division: after one border selected, the corresponding divided border is obtained from the original frame border (with width u). The divided up border (DUB), divided down border (DDB), divided left border (DLB) and divided right border (DRB) are shown in the bottom left part in Fig. 4. The reason lying behind this division is that: the region in the frame corner is often connected with two borders and its feature is also related to these two borders. Thus, the irregular shape connecting three borders is used to calculate the virtual border. The parameters u and l are selected empirically. In this paper, u is set to 5 and l is set to 18%. Preliminary experiments showed that these values make the algorithm robust to various background complexities.

c) Representative Pixel Selection: for the generated divided border, the sum of absolute differences (SAD) is computed for each pixel by summing all the absolute differences between this pixel and other pixels in the divided border:

$$\mathrm{SAD}(x) = \sum_{x' \in \mathrm{DB}} \left| I(x) - I(x') \right| \tag{1}$$

where DB $\in$ {DLB, DUB, DUB, DDB}, I is the feature channel. The pixel having the minimum SAD is selected to represent the divided border. For color images, the SAD is computed by summing the three color channels:

$$\mathrm{colorSAD}(x) = \sum_{x' \in \mathrm{DB}} \sum_{i \in \{r,g,b\}} \left| I^i(x) - I^i(x') \right| \tag{2}$$

We have also considered using the mean or median value of the borderâs intensities as the representative pixel value. Various experiments conducted on different frames have shown that the minimum SAD choice performs better than the mean and the median values in most of the cases (cf. the 1st example image in Fig. 4 where the representative pixel is chosen from the salient object instead of the background when using the mean value of the borderâs intensities). The same way, choosing the median value of the border's intensities as the representative pixel value fails, which can be seen on the 2nd example image in Fig. 4.

d) Virtual Border Padding: around the selected original frame border, we build the corresponding virtual border with the above representative pixel. The virtual up border (VUB), the virtual down border (VDB), the virtual left border (VLB) and the virtual right border (VRB) are shown in the bottom right part in Fig. 4. Existing methods usually regard the border (with width 1) to be background and seed sizes are set to be 1. Here we set the virtual border size v to 5, which helps the proposed "virtual border building" to be applied to other distance transform based saliency detection methods.

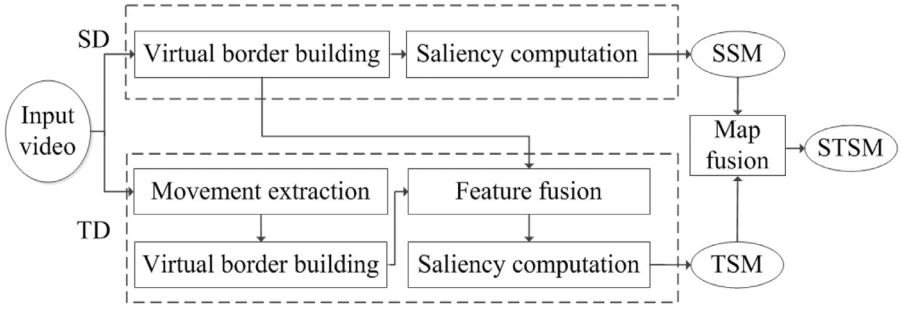

Fig. 3. The proposed block-diagram. SD: spatial saliency detection; SSM: spatial saliency map; TD: temporal saliency detection; TSM: temporal saliency map; STSM: spatio-temporal saliency map.

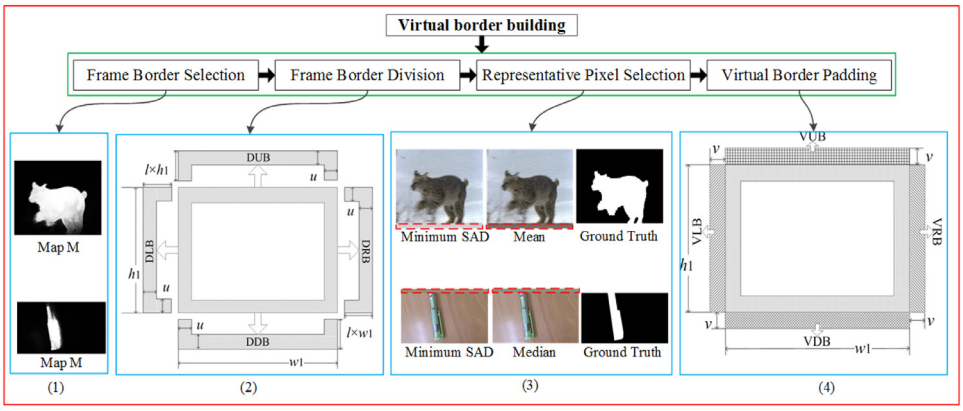

Fig. 4. Virtual border building: (1) two examples of map M obtained by applying FastMBD [9] on the frame; and then for each frame, the closest border to the salient region is selected to build the virtual border; (2) generating the divided border from the highlighted frame border (with width u), h_1 is the frame height, w_1 is the frame width and l is set to 18%, four divided borders: divided up border (DUB), divided down border (DDB), divided left border (DLB), divided right border (DRB) are shown; (3) the red dotted line denotes the virtual border padded with the selected representative pixel; (4) building and padding the virtual border (with size v) with representative pixel value, four virtual borders: virtual up border (VUB), the virtual down border (VDB), the virtual left border (VLB) and the virtual right border (VRB), are shown in four different textures. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

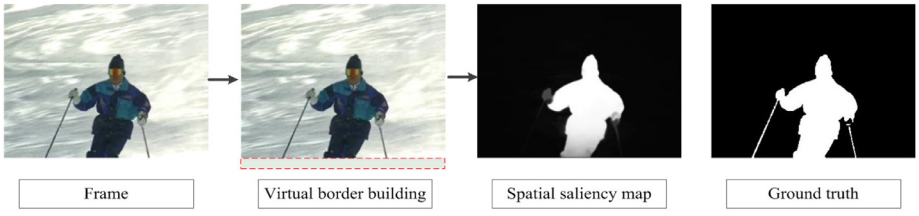

Fig. 5. An example of the spatial saliency detection. The red dotted line denotes the virtual border. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

3.1.2. Saliency computation

After the "virtual border building", the spatial saliency map SSM is obtained by apply the FastMBD [9] to the with-virtual-border frame D and then remove the virtual border region from the resulted map. One example is given to show the process of spatial saliency detection in Fig. 5.

3.2. Temporal saliency detection (TD)

Given an input video sequence, the movement information is extracted from the whole video and then the salient object is detected from this movement information.

3.2.1. Movement extraction

The optical flow vectors between pairs of successive frames are obtained using a fast optical flow method [43]. Then the optical flow vector is mapped to Munsell color system to produce the color optical flow map E (an example image can be found in Fig. 6).

3.2.2. Virtual border building

Based on the background cue, the global motion is usually connected to E borders. The global motion is mainly generated by the background and camera motion. The distance of each pixel to the border pixels of E calculated by the FastMBD [9] can indicate its temporal saliency. The larger the distance, the higher the temporal saliency value. As the same problem in the spatial saliency detection, when the salient object touches frame borders, its movement information also touches E borders. If we directly apply the FastMBD [9] on E, the salient object movement part connected to E borders is hard to be detected. Thus, we add virtual borders on E using the same procedure as described in Section 3.1.1 to obtain the with-virtual-border color optical flow map F.

3.2.3. Feature fusion

We propose a new Feature fusion way that fuses the spatial edge with the temporal information, considering that: 1) the salient object movement is often bigger than the background movement, thus the background and the salient object are often

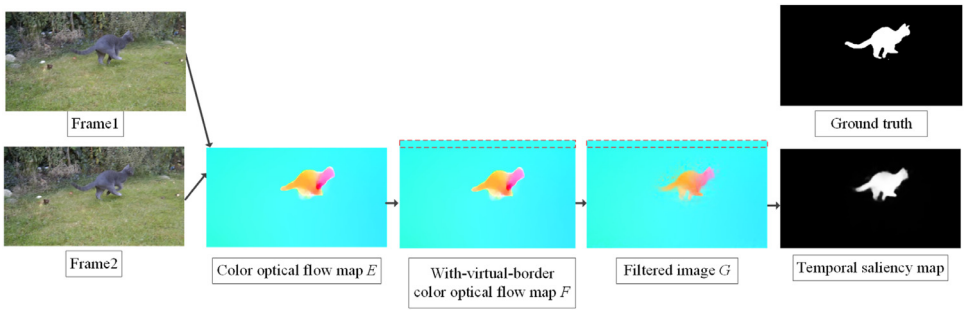

Fig. 6. An example of the temporal saliency detection: from two successive frames, the optical flow vector is extracted and mapped to be the color optical flow map *E*. The virtual border is built on map *E* to generate with-virtual-border color optical flow map *F*. The red dotted line denotes the virtual border. After guided filtering, the filtered image *G* is generated to produce the temporal saliency map. Ground truth is provided for comparison. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

shown in different colors in the color optical flow map; 2) if the movements within the salient object are different, the salient object cannot be detected completely. If the spatial edges are added onto *F*, the salient object edges will be enhanced. The pixel's distance in blur edges will be increased if the pixel belongs to the salient object or decreased if the pixel belongs to the background. Thus we performed the guided image filtering. The guided filter [14] is a linear filtering process, which involves a guidance image C^1, an input image C^2 and an output image C^3. The C^3 at a pixel i is computed using the filter kernel K which is a function of C^1 but independent of C^2.

$$C^3{}_i = \sum_j K_{ij}(C^1)C^2{}_j, \tag{3}$$

where i and j are pixel indexes, and

$$K_{ij}(C^1) = (|\omega_k|)^{-2} \sum_{(i,j)\in\omega_k} (1 + (C^1{}_i - \mu_k)(C^1{}_j - \mu_k)(\sigma_k{}^2 + \epsilon)^{-1}), \tag{4}$$

where ω_k is the square window centered at the pixel k in C^1, $|\omega_k|$ is the number of pixels in ω_k, ϵ is a regularization parameter, and μ_k and σ_k^2 are the mean and the variance of C^1 in ω_k. The main assumption of the guided filter is a local linear model between C^1 and C^3. Thus, C^3 has an edge if C^1 has an edge.

The proposed method use with-virtual-border frame *D* as the guidance image and with-virtual-border color optical flow map *F* as the input image to get the filtered image *G* as Eq. (5),

$$G_i = \sum_j |\omega_k|^{-2} \sum_{(i,j)\in\omega_k} (1 + (D_i - \mu_k)(D_j - \mu_k)(\sigma_k{}^2 + \epsilon)^{-1})F_j, \tag{5}$$

where i and j are pixel indexes, ω_k is the square window centered at the pixel k in D_i, μ_k and σ_k are the mean and the variance of D_i in ω_k. ϵ is set to be 10^{-6}. $|\omega_k|$ is decided by the frame size. Large frame size needs large $|\omega_k|$. We use 20×20 for Fukuchi and FBMS datasets, and use 60×60 for VOS dataset since VOS has larger average frame size than that of Fukuchi and FBMS [26,26,44].

3.2.4. Saliency computation

The FastMBD [9] is applied on the filtered image *G* and then the virtual border region is removed to obtain the temporal saliency map TSM. One example is given to show the process of the temporal saliency detection in Fig. 6.

3.3. Map fusion

Given the spatial saliency map SSM and the temporal saliency map TSM, the fusion is made to obtain spatio-temporal saliency map STSM by four steps:

- SSM and TSM are firstly fused as Eq. (6), where $ratio_1 = mu_T/(mu_S + mu_T)$, $ratio_2 = 1 - ratio_1$.

$$\mathrm{STSM} = \mathrm{ratio}_1 \times \mathrm{SSM} + \mathrm{ratio}_2 \times \mathrm{TSM} \tag{6}$$

where mu_S and mu_T are respectively the mean entropies of all the spatial saliency maps and all the temporal saliency maps for a video sequence (with $\varkappa$ the number of frames) as Eq. (7).

$$mu_S = \sum_{j=1}^{\varkappa} \left(-\sum_{j'=1}^{255} \left(Prob_{j'}^{Sj} \times \log\left(Prob_{j'}^{Sj} \right) \right) \right) \Big/ \varkappa$$
$$mu_T = \sum_{j=1}^{\varkappa} \left(-\sum_{j'=1}^{255} \left(Prob_{j'}^{Tj} \times \log\left(Prob_{j'}^{Tj} \right) \right) \right) \Big/ \varkappa \tag{7}$$

where $Prob_{j'}^{Sj}$ and $Prob_{j'}^{Tj}$ are respectively the normalized histogram of jth spatial saliency map and jth temporal saliency map: $Prob_{j'} = num_{j'}/(h_1 \times w_1)$, $num_{j'}$ is the number of pixel (equal to j') in saliency map. Here, the idea is that mu_i $(i = S, T)$ are used to decide the confidence of SSM and TSM. The disorder degree of saliency map reflects the difficulty degree to detect the salient objects. If mu_i $(i \in \{S, T\})$ is larger, the saliency detection in this domain is worser.

- STSM is optimized using Eq. (8)

$$\mathrm{STSM} = \mathrm{SSM} \quad \text{if} \quad \mathrm{mu_S} < \mathrm{mu_T} \tag{8}$$

The frame is often more complex than the color optical flow map, which results in that the disorder degree of SSM is usually larger than that of TSM. If mu_S is smaller than mu_T, it means it is difficult to detect the salient object in TSM. Thus, SSM has a high confidence.

- STSM is optimized using Eq. (9)

$$\mathrm{STSM} = \mathrm{SSM} \quad \text{if} \quad \sigma_S > \sigma_T \tag{9}$$

σ_S and σ_T are respectively the standard deviations of non-zero regions in two grayscale images H_S and H_T, which are generated by the following steps: firstly, converting frame α from RGB to HSI color space, then eliminating the hue and saturation information while retaining the luminance to get the grayscale images α'; secondly, using a threshold δ to neglect the pixels with low saliency value from the images SSM and TSM as in Eq. (10)

$$H_{S_{ij}} = \begin{cases} 0 & if \quad \mathrm{SSM_{ij}} < \delta \\ \alpha'_{ij} & otherwise \end{cases} \quad H_{T_{ij}} = \begin{cases} 0 & if \quad \mathrm{TSM_{ij}} < \delta \\ \alpha'_{ij} & otherwise \end{cases} \tag{10}$$

where i and j are pixel indexes in the images. The appearance of the wrongly detected background is mostly different from the salient object in the grayscale image, which results in that H_i $(i \in \{\mathrm{S,T}\})$ contains more luminance values and thus σ_i

($i \in \{S,T\}$) is smaller. If σ_S is bigger than σ_T, it means SSM has a high confidence.
- Low saliency value (lower than δ) in STSM is decreased to 0.1 times.

The pixels with low saliency value in saliency map are unimportant for visual saliency but have a large influence in computing the detection confidence. Thus, δ is used to decrease their affection and set to 70 in all this paper.

4. Experiments and analyses

In this section, the performance of the proposed method is assessed and discussed.

4.1. Performance evaluation

Three metrics are used to measure the similarity between the generated saliency map (SM) and the Ground truth (GT):

- Precision-recall (P-R) curve [7]: the saliency map is normalized to [0, 255] and converted to a binary mask (BM) via a threshold that varies from 0 to 255. The precision and the recall are:

$$\text{Precision} = \left|\text{BM} \bigcap \text{GT}\right| / |\text{BM}|, \text{Recall} = \left|\text{BM} \bigcap \text{GT}\right| / |\text{GT}| \quad (11)$$

  For each threshold, a pair of (Precision, Recall) values are computed and used for plotting P-R curve. The curve closest to the top right corner (1.0, 1.0) corresponds to the best performance.
- *F*-measure [45]: higher *F*-measure means better performance.

$$F-\text{measure} = (1+\beta^2) \times (\text{Precision} \times \text{Recall}) \times (\beta^2 \times \text{Precision} + \text{Recall})^{-1} \quad (12)$$

  β^2 is often set to 0.3. Average precision (the average of precision values at all ranks) and average recall (the average of recall values at all ranks) are used.
- Mean Absolute Error (MAE) [7]: smaller MAE means higher similarity and better performance.

$$\text{MAE} = (\text{h1} \times \text{w1})^{-1} \sum_{\text{i}=1}^{\text{h1}\times\text{w1}} |\text{GT(i)} - \text{SM(i)}| \quad (13)$$

For each tested dataset, we compute the average metric for each video sequence and then compute the average metric for all the videos.

4.2. Test datasets

Three datasets with various contents and various conditions are used for models' performance evaluation and comparison.

4.2.1. Datasets with many salient objects connected to the frame border

Fukuchi dataset [44] includes 10 sequences. The salient object touches the frame border in most video sequences. All tested methods hardly detect the salient object for the video "BR128T". As in [34], the video "BR128T" is excluded in the test.

4.2.2. Datasets with complex backgrounds

FBMS dataset [26] is with 59 heterogeneous video sequences. The GT is available for only a part of frames. We use the test set that contains 30 videos with provided GT for evaluation. The global motion with high complexity exists in most of the video sequences.

4.2.3. Datasets with large daily videos

VOS dataset [26], proposed for video salient object detection, contains 200 indoor/outdoor videos (64 min, 116,103 frames). The GT is available for part of frames. VOS-E and VOS-N are two subsets: VOS-E contains 97 easy videos and VOS-N contains 103 videos (the background is cluttered and salient object is highly dynamic). This large-scale dataset is used to benchmark models with the evaluation metrics: MAE, Precision, Recall and *F*-measure. Note that for the calculation of metrics, an adaptive threshold (computed as the minimum value between "maximum pixel value of saliency map" and "twice the average values of saliency map") is used for converting the saliency map to a binary mask (BM). Except for MAE, the author denotes other three metrics in the benchmark [26] as the mean Average Precision (MAP), mean Average Recall (MAR) and FBeta.

4.3. Results and discussions

Two experimental parts with assorted aims are shown for analysis. Firstly, the proposed method (based on traditional image-based salient object detection [9]) in Section 3, denoted as VBGF, is evaluated in Section 4.3.1. The performance of each component of the model is shown to demonstrate our contributions. The VBGF's performance is then compared with nine state-of-the-art traditional salient object detection methods. Secondly, the VBGF is further improved by integrating a deep learning based image salient object detection method [36] and denoted as VBGFd. In Section 4.3.2, the contributions are shown by analyzing the performance of each component. Then performance benchmarking of our approaches (VBGF and VBGFd) and 13 state-of-the-art models is reported. Finally, the run-time complexity is compared in Section 4.3.3.

4.3.1. Performance of the VBGF

Nine state-of-the-art saliency models are tested: MST16 [8], FastMBD15 [9], AMC13 [10], TGFV17 [22], SGSP16 [13], RWR15 [11], GF15 [12], SAG15 [28], FD17 [34] on Fukuchi and FBMS dataset. For all the methods, the experimental results are obtained using the source codes or saliency results provided by the authors.

1) Contributions of each proposed component to the performance

a) Contribution of the proposed virtual border building

The method (based on the "background prior") may miss the salient object connected to the image borders and the proposed virtual border aims to improve this problem. Since MST16 [8], FastMBD15 [9] and AMC13 [10] detect the salient object in image domain based on the "background prior", we compare the proposed spatial saliency map with them by using the Fukuchi dataset, in which many salient objects connected to the frame border. Quantitative performance can be found in Fig. 7. The proposed spatial saliency detection has a better performance since it can detect salient objects more completely.

b) Contribution of the proposed Feature fusion

The proposed Feature fusion employ the guided filter to fuse the spatial edges with the information in temporal domain. We compare the performance of the proposed temporal saliency map with guided filtering and without guided filtering. In the Fukuchi dataset the salient object motion is small, and in the FBMS dataset, the global motion varies largely. These two different datasets are both used. Quantitative performance can be found in Figs. 8 and 9. We can see that fusing the spatial salient object edges to the temporal information by using guided filtering can improve the detection accuracy. It help to optimize the salient object edges and remove the background part from the saliency region.

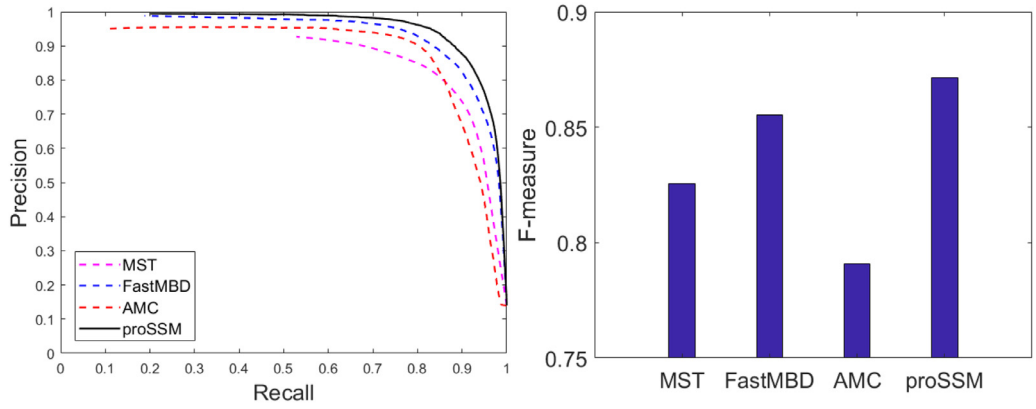

Fig. 7. Quantitative comparisons between our proposed spatial saliency map (proSSM) and three image salient object detection models over the Fukuchi dataset. Some state-of-the-art methods, including: MST16 [8], FastMBD15 [9] and AMC13 [10]. The left parts show the Precision-Recall curves, the right parts shows the *F*-measure↑ scores.

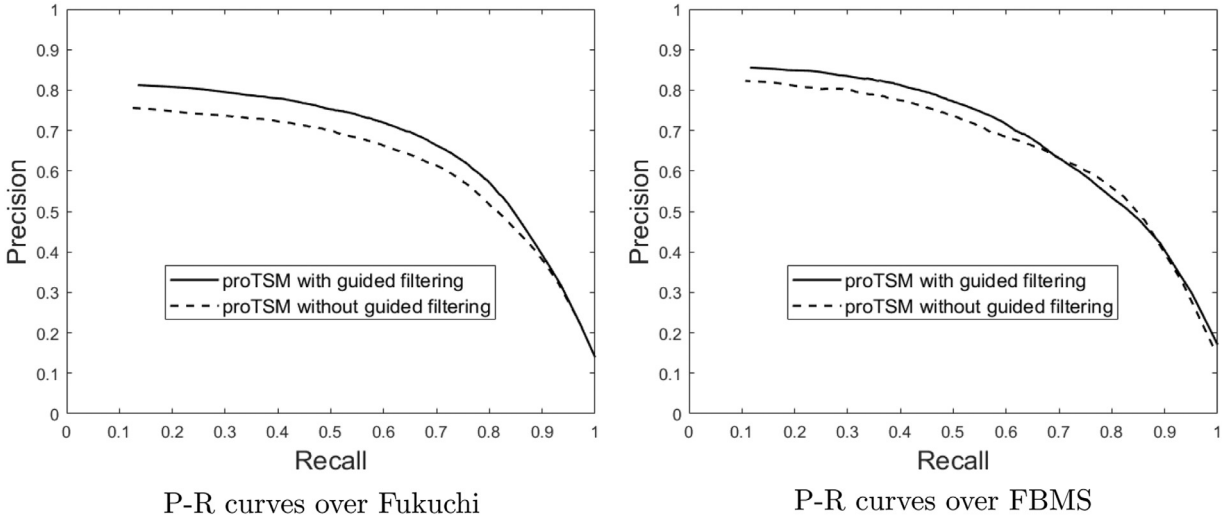

Fig. 8. Precision-Recall (P-R) curves of the proposed temporal saliency map (proTSM) with guided filtering and without guided filtering over the Fukuchi dataset and the FBMS dataset.

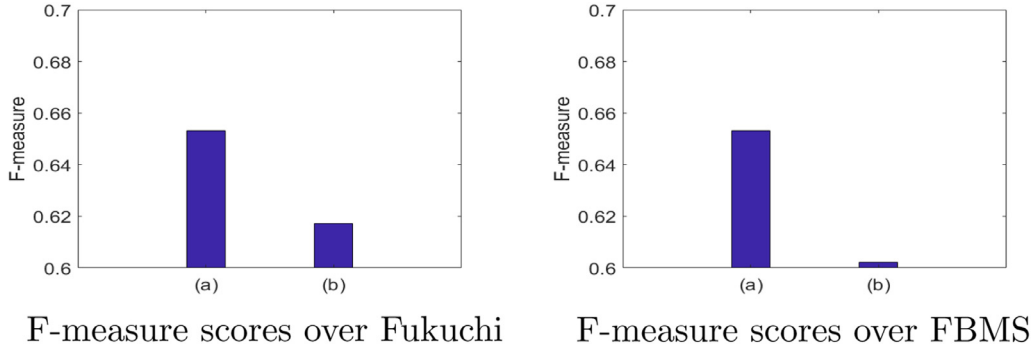

Fig. 9. *F*-measure↑ scores of the proposed temporal saliency map: (a) with guided filtering and (b) without guided filtering over the Fukuchi dataset and the FBMS dataset.

c) Contribution of the proposed Map fusion method

Our proposed method first generates spatial saliency map (cf. Section 3.1), then generates the temporal saliency map (cf. Section 3.2), finally generates the spatio-temporal saliency map (cf. Section 3.3). Therefore, we separately test the performance of each proposed saliency map, then compared quantitative results can be found in Figs. 10 and 11. For the Fukuchi dataset, the salient object motion is slow while the salient object and the background are in high contrast. Compared with the spatial saliency map, the detected temporal saliency has a lower confidence. The proposed fusion can still get a good performance by retaining the spatial saliency map while neglecting the temporal detection influence. For the FBMS dataset, the low contrast and the complex background in the spatial domain make the spatial saliency detection inaccurate. Though the global motion is intricacy, the temporal saliency map is still better than the spatial saliency map. The proposed fusion method takes advantages of results from both domains and gives a higher overall performance.

2) Comparison of the proposed method with state-of-the-art methods

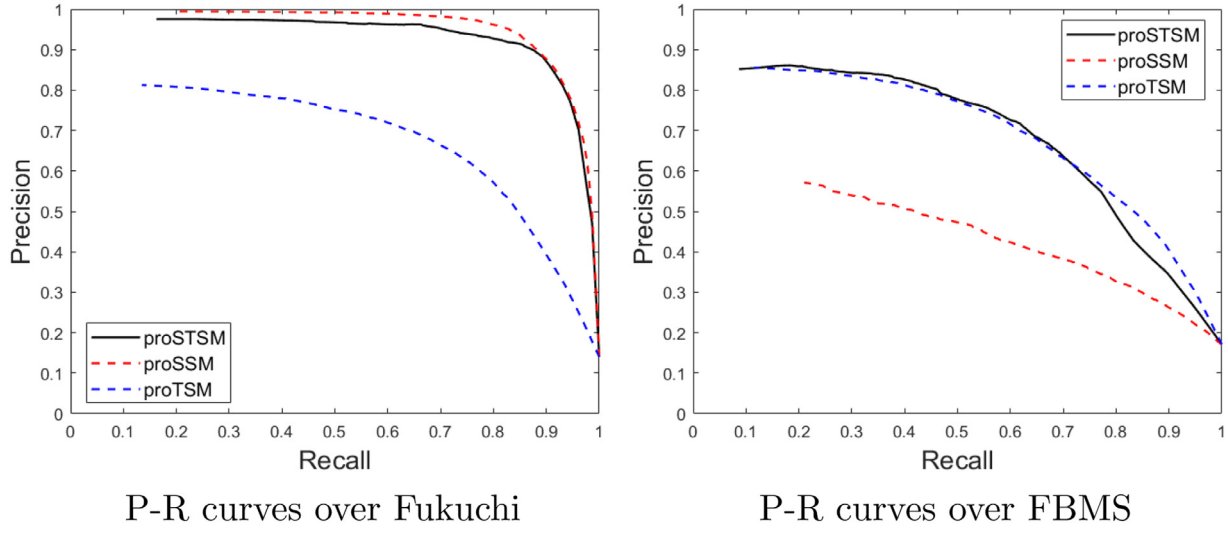

Fig. 10. Precision-Recall (P-R) curves of proSSM, proTSM and proSTSM over the Fukuchi dataset and FBMS dataset. proSSM: proposed spatial saliency map; proTSM: proposed temporal saliency map; proSTSM: proposed spatio-temporal saliency map.

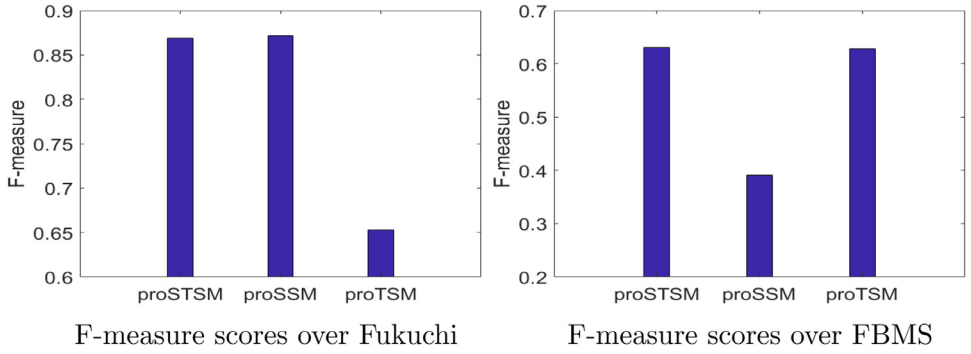

Fig. 11. *F*-measure↑ scores of proSSM, proTSM and proSTSM over the Fukuchi dataset and the FBMS dataset. proSSM: proposed spatial saliency map; proTSM: proposed temporal saliency map; proSTSM: proposed spatio-temporal saliency map.

a) Quantitative comparison with video salient object detection models

We compare our proposed method with several video salient object detection models with the Fukuchi dataset and the FBMS dataset respectively.

For the Fukuchi dataset, six compared models are: TGFV17 [22], SGSP16 [13], RWR15 [11], GF15 [12], SAG15 [28], FD17 [34]. The P-R curves, *F*-measure and MAE values are drawn in Fig. 12, from which we can see that the proposed method has the best P-R curve, the highest *F*-measure and the smallest MAE values. The detailed MAE and *F*-measure scores over four video sequences are shown in Table 1 and the proposed method achieves the best performance. These four video sequences are selected with different cases: in "AN119T", the salient object locates in the frame center; in "DO01_013", all the salient object touch the frame border and in "DO01_055" and "DO02_001" part of salient objects touch frame border. In the Fukuchi dataset, the contrast between the salient object and the background is large and the salient object movement is slow. Spatial saliency detection thus can already provide a high confidence, while the wrong detections in the temporal domain may influence the final saliency map. Compared with methods TGFV17 [22], SGSP16 [13], RWR15 [11], GF15 [12], SA15 [28], and FD17 [34], the proposed fusion method can better select higher confidence spatial saliency information from two domains.

For the FBMS dataset, five compared models are TGFV17 [22], SGSP16 [13], RWR15 [11], GF15 [12], SAG15 [28]. Fig. 13 reports the P-R curves, *F*-measure and MAE values. We can see that our proposed method performs the best, while all the methods get lower performances on this dataset since it is the most challenging one. Five videos with difficult cases (the salient object is similar to

Table 1
A table comparing the proposed method and six video salient object detection models in Mean Absolute Error and *F*-measure scores over 4 video sequences chosen from the Fukuchi dataset.

Method	Mean absolute error↓ scores			
	AN119T	DO01_013	DO01_055	DO02_001
TGFV17 [22]	0.0119	0.0084	0.0462	0.0324
SGSP16 [13]	0.0772	0.0675	0.0996	0.1463
RWR15 [11]	0.0692	0.0773	0.052	0.0826
GF15 [12]	0.0312	0.0306	0.0334	0.0378
SAG15 [28]	0.0264	0.0247	0. 026	0.0162
FD17 [34]	0.0062	0.0086	0. 0165	0.0113
Ours	**0.0027**	**0.0052**	**0.0053**	**0.0014**
Method	*F*-measure↑ scores			
	AN119T	DO01_013	DO01_055	DO02_001
TGFV17 [22]	0.9069	0.704	0.7228	0.808
SGSP16 [13]	0.7318	0.6343	0.5411	0.5925
RWR15 [11]	0.4878	0.5379	0.6533	0.6182
GF15 [12]	0.8659	0.6842	0.7417	0.8292
SAG15 [28]	0.8432	0.5486	0.7393	0.8348
FD17 [34]	0.9449	0.685	0.7852	0.8656
Ours	**0.9516**	**0.801**	**0.8051**	**0.9322**

The Bold number indicates the best result.

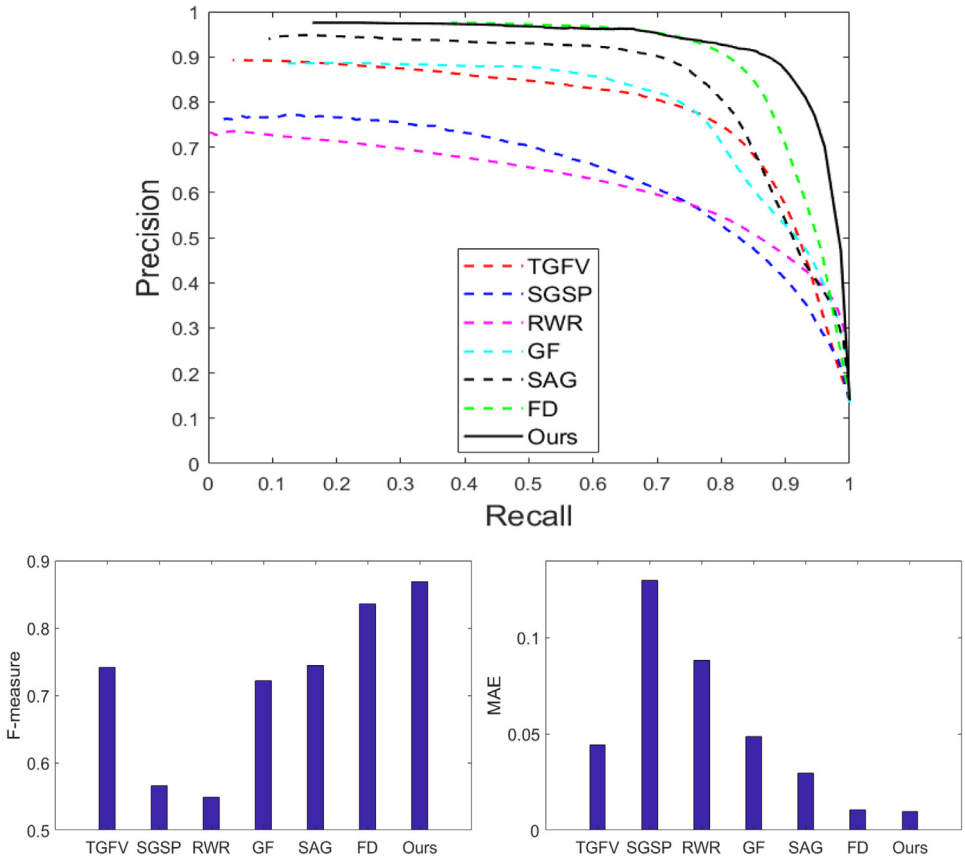

Fig. 12. Quantitative comparisons between our method and six video salient object detection models over the Fukuchi dataset. The upper parts show the Precision-Recall curves, the left below shows the *F*-measure↑ scores and the right below shows the Mean Absolute Error (MAE)↓ scores. Some state-of-the-art methods, including: TGFV17 [22], SGSP16 [13], RWR15 [11], GF15 [12], SAG15 [28], FD17 [34].

Table 2
A table comparing the proposed method and five video salient object detection models in Mean Absolute Error and *F*-measure scores over 5 video sequences chosen from the FBMS dataset.

Method	Mean absolute error↓ scores				
	Cars5	Cars10	Cats03	Horses04	Horses05
TGFV17 [22]	0.0205	0.0248	0.0536	0.0454	0.0363
SGSP16 [13]	0.0708	0.0599	0.1089	0.0964	0.0877
RWR15 [11]	0.1905	0.1485	0.1471	0.1175	0.0968
GF15 [12]	0.0438	0.0388	0.1148	0.1049	0.0598
SAG15 [28]	0.0486	0.034	0. 0941	0.1427	0.0689
Ours	**0.0161**	**0.0218**	**0.0103**	**0.0243**	**0.0215**
Method	*F*-measure↑ scores				
	Cars5	Cars10	Cats03	Horses04	Horses05
TGFV17 [22]	0.751	0.6494	0.6573	0.7021	0.6018
SGSP16 [13]	0.6359	0.6595	0.6558	0.6476	0.6105
RWR15 [11]	0.3485	0.4056	0.2219	0.3389	0.3666
GF15 [12]	0.5877	0.6339	0.2762	0.6415	6067
SAG15 [28]	0.4964	0.584	0.3532	0.3797	0.6495
Ours	**0.7712**	**0.7281**	**0.7184**	**0.7294**	**0.6593**

The Bold number indicates the best result.

the background or the clustering background is complex) are selected and the detailed corresponding MAE and *F*-measure scores are shown in Table 2, in which the proposed method is always the best method. In the FBMS dataset, on one hand, the global motion exists in many sequences and is with high complexity which make the temporal detection more difficult. On the other hand, the salient object appearance is similar to that of the background and the clustering background is complex which makes the spatial detection more difficult. Among methods TGFV17 [22], SGSP16 [13], RWR15 [11], GF15 [12] and SAG15 [28], TGFV17 [22] gets a better result since they put emphasize on the temporal saliency detection. However, compared with TGFV17 [22], the proposed method leverage the spatial saliency and fuse them in a more confidence way to obtain better result.

b) Subjective comparison with 3 image salient object detection models and 5 video salient object detection models

To evaluate the overall performances and disparities between our method and the state-of-the-art methods, we also show a subjective comparison in Fig. 14, (a), (e), (f) and (g) are chosen from the Fukuchi dataset; (b), (c), (d), (h), (i), (j) and (k) are from the FBMS dataset. We can see that RWR15 [11] tends to detect salient object edges rather than the whole salient object. Methods: MST16 [8], FastMBD15 [9], AMC13 [10], TGFV17 [22], SGSP16 [13], GF15 [12], SAG15 [28] can detect salient object region located in the frame center but not the salient part close to frame borders. By visually comparing on this figure, we can see that the proposed method can detect the salient object more completely and more accurately.

4.3.2. Performance of the VBGFd

It may be worthy to look at the performance of the VBGF with an integration of a deep learning based method, named VBGFd. In VBGF, the "Saliency computation" part adopts the traditional method [9], and the "Virtual border building" part is proposed to solve the problem appeared in this type of traditional methods (cf. Fig. 3). For the VBGFd, we replace the "Saliency computation" and "Virtual border building" parts in both "SD" and "TD" blocks in Fig. 3 by a deep-salient detection method proposed in [36] -

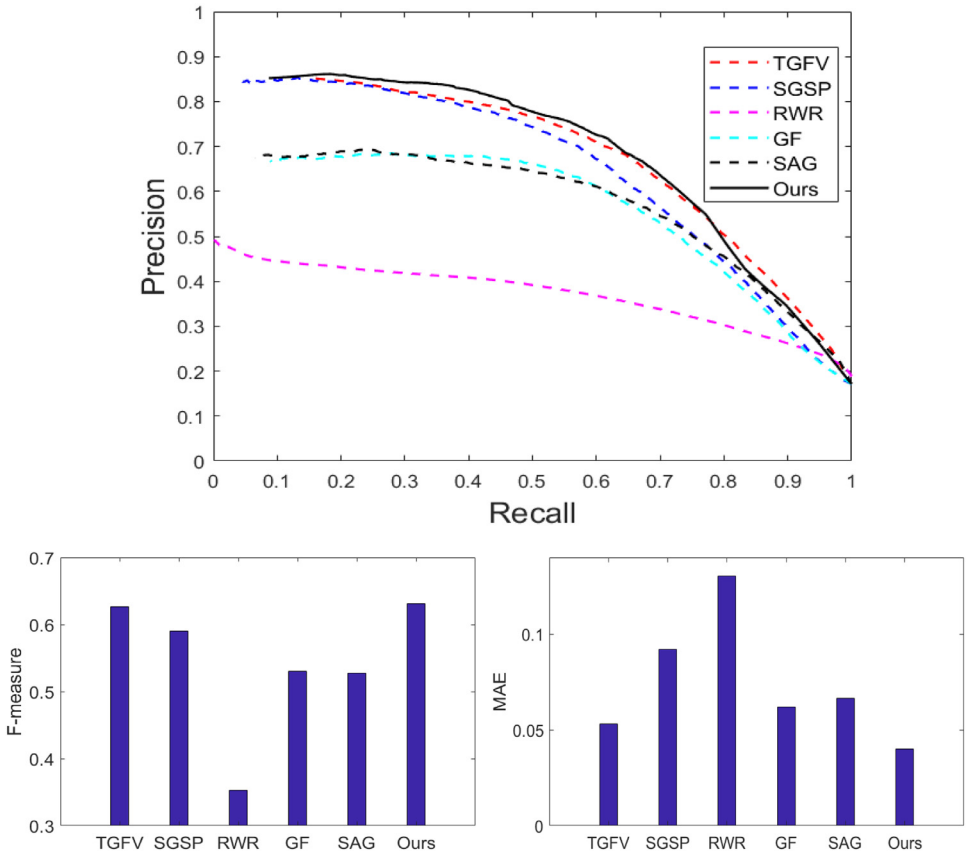

Fig. 13. Quantitative comparisons between our method and five video salient object detection models over the FBMS dataset. The upper parts show the Precision-Recall curves, the left below shows the *F*-measure↑ scores and the right below shows the Mean Absolute Error (MAE)↓ scores. Some state-of-the-art methods, including: TGFV17 [22], SGSP16 [13], RWR15 [11], GF15 [12] and SAG15 [28].

Table 3
Comparison of the proposed VBGFd componets' performance on dataset VOS, VOS-E, VOS-N. proSSM: proposed spatial saliency map; proTSM: proposed temporal saliency map; proSTSM: proposed spatio-temporal saliency map.

Dataset	Metrics	Proposed VBGFd components			
		proSSM	proTSM without guided filtering	proTSM with guided filtering	proSTSM
VOS-E	MAP↑	0.863	0.398	0.528	**0.881**
	MAR↑	**0.905**	0.380	0.480	0.877
	FBeta↑	0.872	0.394	0.516	**0.880**
	MAE↓	0.049	0.189	0.154	**0.046**
VOS-N	MAP↑	0.649	0.407	0.407	**0.690**
	MAR↑	**0.851**	0.389	0.392	0.806
	FBeta↑	0.686	0.403	0.403	**0.714**
	MAE↓	0.055	0.136	0.132	**0.059**
VOS	MAP↑	0.753	0.403	0.466	**0.783**
	MAR↑	**0.877**	0.385	0.435	0.840
	FBeta↑	0.778	0.399	0.458	**0.795**
	MAE↓	**0.052**	0.162	0.143	0.053

The Bold number indicates the best result in each line.

DHSNet (because of the availability of its source code). Besides, the first two steps in the "Map fusion" part in Fig. 3 change to use the ratio between the entropies for each frame in Eq. (6), instead of using the ratio between mean entropies for the whole video sequence. In this section, the large-scale video salient object detection dataset VOS and its two subsets VOS-E, VOS-N are used.

(1) Contributions of the proposed components

In Table 3, we list the performances of the VBGFd with different components. We can see that its performance is better for all performance evaluation metrics with the "guided filtering" by comparing the 4th and 5th columns in Table 3 (contribution (2)); and its performance is better for most performance evaluation metrics when the spatial and the temporal information is fused by comparing the 3rd, 5th and 6th columns in Table 3 (contribution (3)).

(2) Performance benchmarking of our approach

In Table 4, we inserted the performance of our proposed models into the benchmarking table (cf. Table III in the paper [26]) provided with the VOS dataset. Note that here we only list 13 state-of-the-art models (image-based deep learning and video-based unsupervised models) reported in [26], not the image-based classic non-deep learning models (because we have already compared with some classic models in Section 4.3.1). We can see that among the tested 15 models, the VBGFd has the best score for 7 times, when the best benchmarked model DHSNet has the best score for 5 times. Thus in general, we can say that the VBGFd performs the best among the tested models.

4.3.3. Time

A PC with Intel Core i7 4910 2.9 GHz CPU and 16GB RAM is used for testing the speed of traditional methods, and the deep learning method is performed on a NVIDIA 1080 GPU. Note that the video (Fukuchi and FBMS datasets) with original resolution is used. For different models that tested in Section 4.3.1 (except the model FD17 [34] with the unpublished code), the average run-time is listed in Table 5. Video-based method SFCN and 3 image-based models have low computation costs. Others have higher computation costs since the optical flow estimation is usually time consuming. The proposed VBGF and VBGFd models are among the three fastest video-based detection models, and the average run-time per frame of each element can be found in Table 6 in detail.

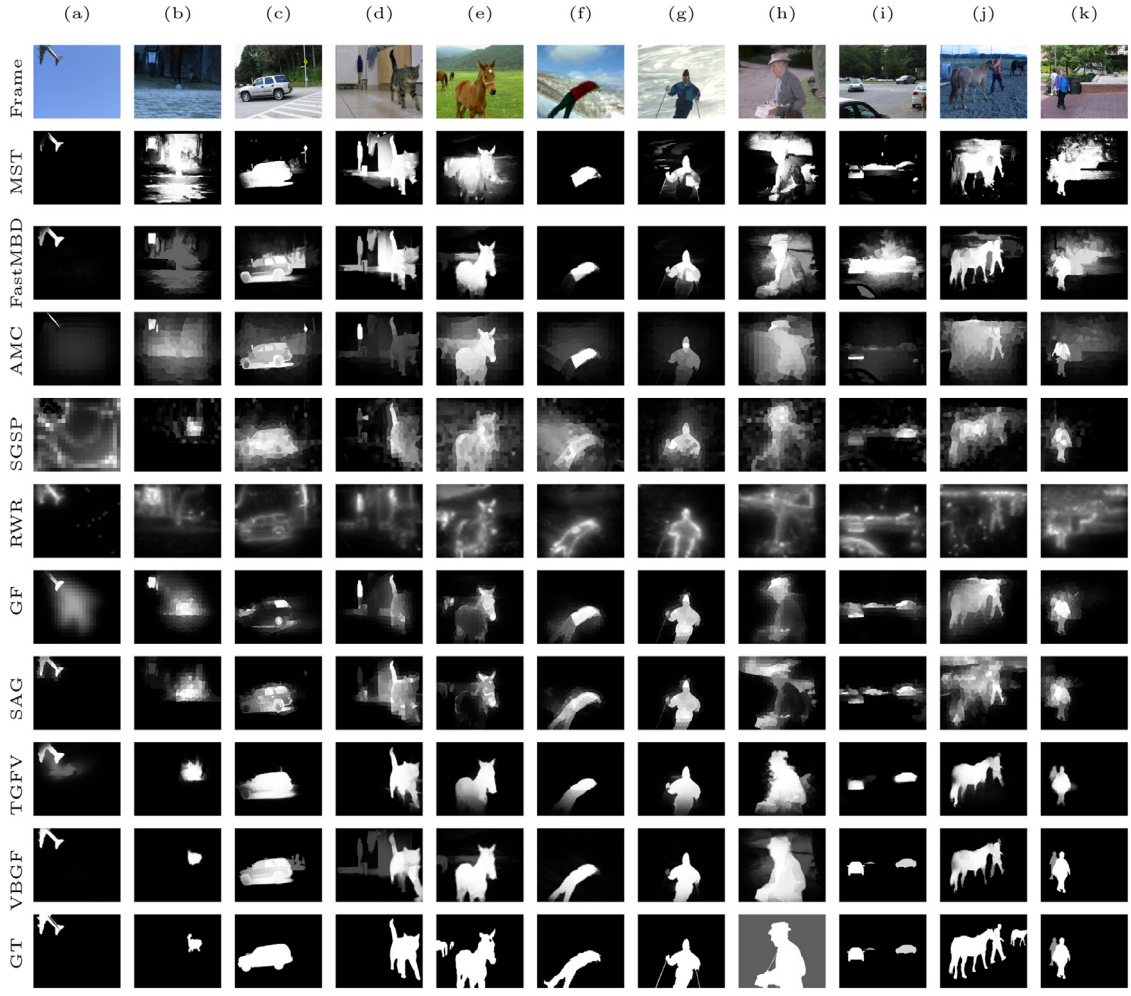

Fig. 14. Comparison of the saliency maps. (a)–(k) are 11 different video sequences. Some state-of-the-art methods, including: MST16 [8], FastMBD15 [9], AMC13 [10], TGFV17 [22], SGSP16 [13], RWR15 [11], GF15 [12], SAG15 [28]. GT: Ground truth.

Table 4
Performance benchmarking of our approach and 13 state-of-the-art models on the dataset VOS and two subsets VOS-E and VOS-N. These models are categorized into two parts: [I+D] for deep learning and image-based, [V+U] for video-based and unsupervised, [V+D] for deep learning and video-based.

	Models	VOS-E				VOS-N				VOS			
		[MAP↑	MAR↑	FBeta↑	MAE↓]	[MAP↑	MAR↑	FBeta↑	MAE↓]	[MAP↑	MAR↑	FBeta↑	MAE↓]
[I+D]	LEGS	0.820	0.685	0.784	0.193	0 556	0.593	0.564	0.215	0.684	0.638	0.673	0.204
	MCDL	0.831	0.787	0.821	0.081	0.570	0.645	0.586	0.085	0.697	0.714	0.701	0.083
	MDF	0.740	0.848	0.762	0.100	0.527	0.742	0.565	0.098	0.630	0.793	0.661	0.099
	ELD	0.790	**0.884**	0.810	**0.060**	0.569	*0.838*	0.615	0.081	0.676	*0.861*	0.712	**0.071**
	DCL	**0.864**	0.735	0.830	0.084	0.583	0.809	0.624	**0.079**	0.719	0.773	0.731	0.081
	RFCN	0.834	0.820	0.831	0.075	0.614	0.783	0.646	0.080	0.721	0.801	0.738	0.078
	DHSNet	0.863	0.905	*0.872*	*0.049*	**0.649**	0.851	*0.686*	0.055	**0.753**	0.877	*0.778*	0.052
[V+U]	SIV	0.693	0.543	0.651	0.204	0.451	0.523	0.466	0.201	0.568	0.533	0.560	0.203
	FST	0.781	*0.903*	0.806	0.076	0.619	0.691	0.634	0.117	0.697	0.794	0.718	0.097
	NLC	0.439	0.421	0.435	0.204	0.561	0.610	0.572	0.123	0.502	0.518	0.505	0.162
	SAG	0.709	0.814	0.731	0.129	0.354	0.742	0.402	0.150	0.526	0.777	0.568	0.140
	GF	0.712	0.798	0.730	0.153	0.346	0.738	0.394	0.331	0.523	0.767	0 565	0.244
	SSA [26]	*0.875*	0.776	**0.850**	0.062	*0.660*	0.682	**0.665**	0.103	*0.764*	0.728	**0.755**	0.083
	VBGF	0.797	0.773	0.791	0.085	0.558	0.688	0.583	0.130	0.674	0.729	0.686	0.108
[V+D]	SFCN [41]	0.806	0.842	0.814	0.063	0.577	**0.815**	0.619	0.086	0.688	0.829	0.716	0.075
	VBGFd	0.881	0.877	0.880	0.046	0.690	0.806	0.714	*0.059*	0.783	**0.840**	0.795	*0.053*

The best three scores in each column are marked in **red**, *green* and **blue**, respectively.
13 state-of-the-art models (LEGS,MCDL,MDF,ELD,DCL,RFCN,DHSNet,SIV,FST,NLC,SAG,GF) can be referenced from the paper [26]. For SFCN, the result is generated using the provided source code.

Table 5
Average run time (per frame) of the compared models (MST16 [8], FastMBD15 [9], AMC13 [10], TGFV17 [22], SGSP16 [13], RWR15 [11], GF15 [12], SAG15 [28]), SFCN [41]).

Image_based	MST	FastMBD	AMC	–	–	–	–	
Time(s)↓	0.200	0.018	0.153	–	–	–	–	
Video_based	SGSP	RWR	GF	SAG	TGFV	SFCN	VBGF	VBGFd
Time(s)↓	15.37	14.25	13.50	15.38	33.17	0.072	3.56	3.14

Table 6
Average run time (per frame) of each component in the proposed models.

Component	VBGF		VBGFd	
	Time(s)	Ratio(%)	Time(s)	Ratio(%)
Virtual border building	0.50	14.04	–	–
Saliency detection	0.07	1.97	0.15	4.78
Optical flow computation	2.80	78.65	2.80	89.17
Feature fusion(guided filtering)	0.07	1.97	0.07	2.23
Map fusion	0.12	3.37	0.12	3.82

5. Conclusion

In this paper, a novel video salient object detection method (the VBGF) and its extension integrating deep representations (the VBGFd) are proposed. Using virtual border concept has helped to address the problem of distance transform employed for saliency computation in previous approaches. The guided filter-based Feature fusion and the Map fusion are efficiently used for fusing spatial and temporal information together by applying appropriate balance. When tested on various video databases, the proposed approach yields satisfactory performance and even outperforms the state-of-the-art methods.

The virtual border can be used as an optimization operation for salient object detection methods that are based on background prior. The guided filter-based Feature fusion helps to remove background regions for moving object detection and segmentation. The Map fusion provides a new way to combine various individual saliency maps into a more robust one. However, the proposed fusion can lead to information loss as the used hand-crafted features are not robust in some complex cases, which may be improved with more informative features; so there is still a room for improvement. Hence for the future work, we intend to explore deep-learning based methods for salient object detection in videos. We also plan to improve the above fusion by training deep networks to learn more useful deep representations.

Acknowledgments

This work was supported in part by the China Scholarship Council (CSC) under Grants 201504490048, in part by the NSFC Project under Grants 61771321, in part by the Guangdong Key Research Platform of Universities under Grants 2018WCXTD015, and in part by the Interdisciplinary Innovation Team of Shenzhen University.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.patcog.2019.106998.

References

[1] B. Qin, Z. Gu, X. Sun, Y. Lv, Registration of images with outliers using joint saliency map, IEEE Signal Process. Lett. 17 (1) (2010) 91–94.
[2] B. Qin, Z. Shen, Z. Fu, Z. Zhou, Y. Lv, J. Bao, Joint-saliency structure adaptive kernel regression with adaptive-scale kernels for deformable registration of challenging images, IEEE Access 6 (2018) 330–343.
[3] X. Zhi, H. Shen, Saliency driven region-edge-based top down level set evolution reveals the asynchronous focus in image segmentation, Pattern Recognit. 80 (2018) 241–255.
[4] W. Wang, J. Shen, R. Yang, F. Porikli, Saliency-aware video object segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 40 (1) (2018) 20–33.
[5] Z. Zhao, B. Zhao, F. Su, Person re-identification via integrating patch-based metric learning and local salience learning, Pattern Recognit. 75 (2018) 90–98.
[6] W. Xie, Y. Shi, Y. Li, X. Jia, J. Lei, High-quality spectral-spatial reconstruction using saliency detection and deep feature enhancement, Pattern Recognit. 88 (2019) 139–152.
[7] A. Borji, M. Cheng, H. Jiang, J. Li, Salient object detection: a survey, arXiv:1411.5878 (2014).
[8] W. Tu, S. He, Q. Yang, S. Chien, Real-time salient object detection with a minimum spanning tree, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27–30, 2016, 2016, pp. 2334–2342.
[9] J. Zhang, S. Sclaroff, Z.L. Lin, X. Shen, B.L. Price, R. Mech, Minimum barrier salient object detection at 80 FPS, in: 2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7–13, 2015, 2015, pp. 1404–1412.
[10] B. Jiang, L. Zhang, H. Lu, C. Yang, M. Yang, Saliency detection via absorbing Markov chain, in: 2013 IEEE International Conference on Computer Vision, ICCV 2013, Sydney, Australia, December 1–8, 2013, 2013, pp. 1665–1672.
[11] H. Kim, Y. Kim, J. Sim, C. Kim, Spatiotemporal saliency detection for video sequences based on random walk with restart, IEEE Trans. Image Process. 24 (8) (2015) 2552–2564.
[12] W. Wang, J. Shen, L. Shao, Consistent video saliency using local gradient flow optimization and global refinement, IEEE Trans. Image Process. 24 (11) (2015) 4185–4196.
[13] Z. Liu, J. Li, L. Ye, G. Sun, L. Shen, Saliency detection for unconstrained videos using superpixel-level graph and spatiotemporal propagation, IEEE Trans. Circt. Syst. Video Technol. 27 (12) (2017) 2527–2542.
[14] K. He, J. Sun, X. Tang, Guided image filtering, in: Computer Vision – ECCV 2010, 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5–11, 2010, Proceedings, Part I, 2010, pp. 1–14.
[15] D. Shan, X. Zhang, C. Zhang, Visual saliency based on extended manifold ranking and third-order optimization refinement, Pattern Recognit. Lett. 116 (2018) 1–7.
[16] X. Huang, Y. Zhang, Water flow driven salient object detection at 180 fps, Pattern Recognit. 76 (2018) 95–107.
[17] M.M.I. Lie, G.B. Borba, H.V. Neto, H.R. Gamba, Joint upsampling of random color distance maps for fast salient region detection, Pattern Recognit. Lett. 114 (2018) 22–30.
[18] Ç. Aytekin, A. Iosifidis, M. Gabbouj, Probabilistic saliency estimation, Pattern Recognit. 74 (2018) 359–372.
[19] X. Li, H. Ma, X. Wang, K. Zhang, Saliency detection via alternative optimization adaptive influence matrix model, Pattern Recognit. Lett. 101 (2018) 29–36.
[20] Y. Yan, J. Ren, G. Sun, H. Zhao, J. Han, X. Li, S. Marshall, J. Zhan, Unsupervised image saliency detection with gestalt-laws guided optimization and visual attention based refinement, Pattern Recognit. 79 (2018) 65–78.
[21] H. Peng, B. Li, H. Ling, W. Hu, W. Xiong, S.J. Maybank, Salient object detection via structured matrix decomposition, IEEE Trans. Pattern Anal. Mach. Intell. 39 (4) (2017) 818–832.
[22] Q. Wang, L. Zhang, K. Kpalma, Fast filtering-based temporal saliency detection using minimum barrier distance, in: 2017 IEEE International Conference on Multimedia & Expo Workshops, ICME Workshops, Hong Kong, China, July 10–14, 2017, 2017, pp. 232–237.
[23] S. Bhattacharya, V.K. Subramanian, S. Gupta, Visual saliency detection using spatiotemporal decomposition, IEEE Trans. Image Process. 27 (4) (2018) 1665–1675.
[24] Z. Tu, Z. Guo, W. Xie, M. Yan, R.C. Veltkamp, B. Li, J. Yuan, Fusing disparate object signatures for salient object detection in video, Pattern Recognit. 72 (2017) 285–299.
[25] B. Yang, X. Zhang, L. Chen, Z. Gao, Spatiotemporal salient object detection based on distance transform and energy optimization, Neurocomputing 266 (2017) 165–175.
[26] J. Li, C. Xia, X. Chen, A benchmark dataset and saliency-guided stacked autoencoders for video-based salient object detection, IEEE Trans. Image Process. 27 (1) (2018) 349–364.
[27] C. Chen, Y. Li, S. Li, H. Qin, A. Hao, A novel bottom-up saliency detection method for video with dynamic background, IEEE Signal Process. Lett. 25 (2) (2018) 154–158.
[28] W. Wang, J. Shen, F. Porikli, Saliency-aware geodesic video object segmentation, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7–12, 2015, 2015, pp. 3395–3402.

[29] T. Xi, W. Zhao, H. Wang, W. Lin, Salient object detection with spatiotemporal background priors for video, IEEE Trans. Image Process. 26 (7) (2017) 3425–3436.
[30] X. Zhou, Z. Liu, C. Gong, W. Liu, Improving video saliency detection via localized estimation and spatiotemporal refinement, IEEE Trans. Multimed. 20 (11) (2018) 2993–3007.
[31] Y. Chen, W. Zou, Y. Tang, X. Li, C. Xu, N. Komodakis, SCOM: spatiotemporal constrained optimization for salient object detection, IEEE Trans. Image Process. 27 (7) (2018) 3345–3357.
[32] H. Ramadan, H. Tairi, Pattern mining-based video saliency detection: application to moving object segmentation, Comput. Electr. Eng. 70 (2018) 567–579.
[33] F. Guo, W. Wang, J. Shen, L. Shao, J. Yang, D. Tao, Y.Y. Tang, Video saliency detection using object proposals, IEEE Trans. Cybern. 48 (11) (2018) 3159–3170.
[34] C. Chen, S. Li, Y. Wang, H. Qin, A. Hao, Video saliency detection via spatial-temporal fusion and low-rank coherency diffusion, IEEE Trans. Image Process. 26 (7) (2017) 3156–3170.
[35] C. Chen, S. Li, H. Qin, Z. Pan, G. Yang, Bilevel feature learning for video saliency detection, IEEE Trans. Multimed. 20 (12) (2018) 3324–3336.
[36] N. Liu, J. Han, Dhsnet: deep hierarchical saliency network for salient object detection, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27–30, 2016, 2016, pp. 678–686.
[37] Q. Hou, M. Cheng, X. Hu, A. Borji, Z. Tu, P.H.S. Torr, Deeply supervised salient object detection with short connections, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21–26, 2017, 2017, pp. 5300–5309.
[38] G. Lee, Y. Tai, J. Kim, Eld-net: an efficient deep learning architecture for accurate saliency detection, IEEE Trans. Pattern Anal. Mach. Intell. 40 (7) (2018) 1599–1610.
[39] H. Chen, Y. Li, D. Su, Multi-modal fusion network with multi-scale multi–path and cross-modal interactions for RGB-D salient object detection, Pattern Recognit. 86 (2019) 376–385.
[40] Q. Yuan, Y. Wei, X. Meng, H. Shen, L. Zhang, A multiscale and multidepth convolutional neural network for remote sensing imagery pan-sharpening, IEEE J Sel. Top. Appl. Earth Observ. Remote Sens. 11 (3) (2018) 978–989.
[41] W. Wang, J. Shen, L. Shao, Video salient object detection via fully convolutional networks, IEEE Trans. Image Process. 27 (1) (2018) 38–49.
[42] Y. Tang, W. Zou, Z. Jin, Y. Chen, Y. Hua, X. Li, Weakly supervised salient object detection with spatiotemporal cascade neural networks, IEEE Trans. Circt. Syst. Video Technol. (2018).
[43] Y. Hu, R. Song, Y. Li, Efficient coarse-to-fine patch match for large displacement optical flow, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27–30, 2016, 2016, pp. 5704–5712.
[44] K. Fukuchi, K. Miyazato, A. Kimura, S. Takagi, J. Yamato, Saliency-based video segmentation with graph cuts and sequentially updated priors, in: Proceedings of the 2009 IEEE International Conference on Multimedia and Expo, ICME 2009, June 28, - July 2, 2009, New York City, NY, USA, 2009, pp. 638–641.
[45] A. Borji, M. Cheng, H. Jiang, J. Li, Salient object detection: a benchmark, IEEE Trans. Image Process. 24 (12) (2015) 5706–5722.

Qiong Wang received the Ph.D. degree from the “National Institute of Applied Sciences of Rennes”, Rennes, France, in 2019. She is currently a faculty of Zhejiang University of Technology. Her current research interests include visual saliency detection, video object segmentation, and deep learning.

Lu Zhang received the Ph.D. degree from University of Angers, Angers, France, in 2012. She is currently an Associate Professor in the “National Institute of Applied Sciences of Rennes” in France. Her research interests include visual saliency detection, multimedia quality assessment, medical imaging and human perception understanding.

Wenbin ZOU received the Ph.D. degree from the “National Institute of Applied Sciences of Rennes” in France, in 2014. Since 2015, he has been with the faculty of the College of Information Engineering, Shenzhen University, China. His current research interests include saliency detection, object segmentation, and semantic segmentation.

Kidiyo Kpalma received his Ph.D. in Image Processing INSA Rennes in 1992. Since 2014, he became Professor at INSA: he teaches Signal and Systems, Signal Processing and DSP. As a member of IETR UMR CNRS 6164, his research interests include pattern recognition, semantic image segmentation, facial micro-expression and saliency object detection.