Applied Soft Computing Journal 108 (2021) 107433

Contents lists available at ScienceDirect

Applied Soft Computing Journal

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

Video salient object detection using dual-stream spatiotemporal attention

Chenchu Xu [a,1], Zhifan Gao [b,1], Heye Zhang [b,*], Shuo Li [c,*], Victor Hugo C. de Albuquerque [d,e]

[a] Anhui University, Hefei, China
[b] Sun Yat-Sen University, Shenzhen, China
[c] Western University, London, Canada
[d] Federal University of Ceará, Fortaleza, Brazil
[e] Science and Technology of Ceará, Fortaleza, Brazil

ARTICLE INFO

Article history:
Received 14 January 2020
Received in revised form 14 March 2021
Accepted 20 April 2021
Available online 26 April 2021

Keywords:
Video salient object detection
Attention mechanism
Context information

ABSTRACT

Video salient object detection plays an important role in many exciting applications in different areas. However, the existing deep learning-based video salient object detection methods still struggle in scenes of large salient object variabilities and great background scene diversity between and within frames. In this paper, we propose a dual-stream spatiotemporal attention network (DSSANet) for saliency detection in videos. It creatively introduces a multiplex attention mechanism to effectively extract and fuse spatiotemporal features of video salient object over frames in the video, thereby improving saliency detection performance. The DSSANet consists of: (1) A context feature path leverages a novel attention-augmented convolutional LSTM to effectively model the long-range dependency of the great temporal variation in the salient object over frames. (2) A content feature path creatively leverages an attention-based 1D dilated convolution to effectively model the local pixel correlation structure of each pixel in the salient object and the surrounding objects. (3) A refinement fusion module fuses these two features from their paths and further refines the fused feature by an attention-based feature selection. By integrating these three parts, DSSANet accurately detects the salient object from the video. The extensive experiments are performed on four public datasets and demonstrate the effectiveness of DSSANet and the superiority to five state-of-the-art video salient object detection methods.

1. Introduction

Video salient object detection plays an important role in many exciting applications in different areas, including the movie industry [1], surveillance technologies [2], medical image analysis [3], to name but a few [4]. Video salient object detection aims to discover the most visually distinctive objects in a video, and accurately segments these objects over each frame [5]. Such detection effectively simulates the humans' vision ability that effortlessly and rapidly detects visually distinctive regions in the video. It helps to find the objects or regions which efficiently represent a scene, thus it is a useful step in complex vision problems such as scene understanding follow-up application [6].

As deep learning technology becoming more prevalent, video salient object detection has experienced significant advances [6]. Compared to hand-crafted features-based traditional machine learning video salient object detection methods [7,8], deep learning-based detection methods leverage the powerful representation learning ability of convolutional neural networks (CNNs) to extract more discriminative and more robust features of the object in the video [9,10]. Deep learning techniques have the ability to capture the most salient regions without any prior knowledge (e.g., segment-level information) [11,12]. Furthermore, their multi-level feature extraction can better locate the boundaries of the detected salient regions, even when shades or reflections exist [13–15]. For example, Le et al. [16] has proposed a CNN-based deep features learning method, which has significantly improve the detection of the accuracy of salient objects' boundaries and noise reduction during detection. Dong et al. [17] leveraged the 3D-CNN to enhance the temporal feature extraction during objects' spatiotemporal learning, thereby effectively improve the understanding of high-dimensional saliency cues from the temporal information. Thus, All of the deep learning-based detection methods have demonstrated superior results over existing works utilizing only hand-crafted features and are becoming the main stream solution. Some methods have become a new baseline at

* Corresponding authors.
E-mail addresses: zhangheye@mail.sysu.edu.cn (H. Zhang), slishuo@gmail.com (S. Li).

[1] Chenchu Xu and Zhifan Gao contributed equally to this work.

https://doi.org/10.1016/j.asoc.2021.107433
1568-4946/

various datasets by integrating recent deep learning technologies (such as FCN [13], DenseNet [18], etc.).

However, the existing deep learning-based video salient object detection methods still suffer from two challenges. The first challenge is the large salient object variabilities (size and location) between frames create ravines in object temporal information [6]. These variabilities make it difficult to build a systematic mathematical model of the global context features to discriminate continuously object in-sequence frames in the video. The second challenge is the great background scene diversity, as well as the complex interrelationships and interdependencies of each point in the salient object and the surrounding objects [6]. This background scene diversity makes it is difficult to extract and select suitable local spatial content features of the salient object for every single image to fine discriminate the boundary pixels of the salient object.

In this paper, we propose a dual-stream spatiotemporal attention network (DSSANet) for the saliency detection in videos. It disentangles spatiotemporal feature learning to different phases (spatial content feature extraction, temporal context feature extraction, and spatiotemporal feature fusion), as well as creatively integrates multiple attention models into each phase to jointly attend to information from different representation subspaces at different phases, improving saliency detection performance. The DSSANet consists of three novelty parts: the context feature path, the content feature path, and the refinement fusion module. As their names imply, the context feature path leverages a novel attention-augmented convolutional LSTM to capture global context information of the salient object that is responsible for the temporal dependencies between and within frames over the video. The content feature path leverages a novel attention-based 1D convolutional network to capture the detailed local spatial content information of the salient object that is responsible for the local pixel correlation structure of each pixel in the salient object and the surrounding objects. Finally, the refinement fusion module fuses these two types of features and further refines the feature representations, thereby accurately segmenting the salient object.

Our contributions can be summarized as:

- We propose a novel dual-stream framework to learn the salient object in the video. It is better able to capture the salient object spatiotemporal features in the video, including both context features and content features, to segment the salient object than the traditional video salient object detection technologies.
- We integrate the multi-attention models into the context feature capturing, the content feature capturing, and the features fusion respectively. It allows different models to devote feature learning in their own representation spaces, improving the effectiveness during object's feature learning.
- We create an attention-augmented convolutional LSTM to enrich the temporal dependency of the salient object. It extends the temporal dependency modeled from the short-range to long-range, thereby handling the large salient object variabilities.

The rest of the paper is arranged as follows: In Section 2 we briefly outline the related work, including the progress of existing video deep spatiotemporal learning, and the foundation of various attention models that can be employed. In Section 3 we describe our methodology, which includes three subsections for introducing network structure and one subsection for introducing loss function. In Section 4 we explain experimental setting, including benchmark datasets, implementation details, evaluation metrics, and state-of-the-art methods used for the comparison. In Section 5 we present the results of our experiments in detail and discuss these results based on the comparison with existing state-of-the-art methods. Conclusions are drawn in Section 6.

2. Related work

2.1. Video spatiotemporal feature learning

Video spatiotemporal feature learning is the fundamental of current video salient object detection. The existing video spatiotemporal feature learning methods can be divided into the traditional models and the deep learning models. Traditional models are mostly developed from traditional saliency models for still images by incorporating motion features to deal with moving objects [19]. These models leverage optical flow mapping [20], local feature trajectories, gradient flow field computation [21], and spatio-temporal motion boundary detection [22] to capture motion features, thereby detecting salient objects in videos. However, these models often fail on learning the semantic concept of objects. This because these models focus on motion features and have limitations in learning salient objects with cross the image boundary and similar appearance with the background.

The deep learning models are based on the recurrent neural network (RNN)-based methods [23,24] and the 3D-convolution (Conv)-based methods [25,26]. The RNN and its famous variants, such as long short-term memory (LSTM) [27,28], have been widely used to model the temporal information between frames in the video and improve the inter-frame consistency. However, LSTM, as a fully connected network, treats a video as a one-dimensional sequence and often leads to fails in taking spatial features into consideration. Also, 3D-Conv has the capability to learn the spatiotemporal representation of high-level context directly from a video. It has been used to capture temporal scene changes when actions are being performed. However, 3D-Conv is hard to effectively model all frames from the beginning to the end of the video because of the well-known granularity loss issue over time [29,30]. Moreover, recent convolutional LSTM [12,31] has been proposed to combine the advantages of LSTM and the convolution network to take the spatial information into consideration when the whole video temporal information is modeling. However, convolutional LSTM still struggles with its fixed kernel size, and this fixed kernel size leads to the ineffective receptive field to model the long-range spatiotemporal features of an object with different local correlation structures in different spatial locations and time periods.

2.2. Attention model

The attention model has shown impressive performance in the computer vision community [32,33]. Although only a few reports involving video salient object detection, attention model has successfully improved the accuracy of the saliency detection task of static images [34]. The attention model can simulate the visual attention mechanism in the human visual system, which allows selectively learn to the most informative and characteristic parts of a visual stimulus rather than the whole scene [35]. Such a model uses CNNs as basic building blocks and calculates long-range representations that respond to all positions in the input and output images [33]. It then determines the key parts that have high responses in the long-range representations and weighs these parts to motivate the networks to better learn the images. The self-attention model, as a recent variant, has been proposed to embed the spatiotemporal learning model to achieve object detection and segmentation [36] and has shown impressive performance [37]. In particular, Tang et al. [11] has used a single self-attention model as the last stage of a two-stream-based spatiotemporal neural network to perform feature selection in different types of spatiotemporal feature maps. However, deploying only one single self-attention model for feature selection is separated from feature capturing, which may be insufficient on whole feature representation learning. In comparison,

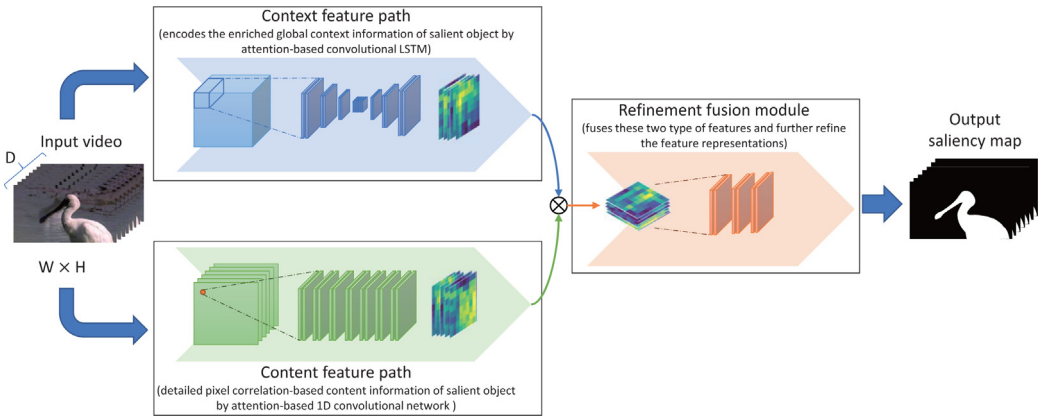

Fig. 1. DSSANet consists of three novelty parts: the context feature path, the content feature path, and the refinement fusion module. The context feature path encodes the enriched global context information of the salient object by an attention-augmented convolutional LSTM with a long-range dependency model. It includes a comprehensive representation to handle all variabilities of the salient object between frames. The content feature path encodes the detailed local spatial content information of the salient object by attention-based 1D dilated Convolution with a multi-scale structure modeling. It includes a fine representation that can handle the interrelationships and interdependency of each pixel in the salient object and surrounding objects. The refinement fusion module fuses two type features from two paths and further refines the feature representations, final accurate segments the salient object.

our DSSANet incorporates multi-attention models into all spatial content features capturing, temporal context features capturing, and feature fusion and selection, respectively. It allows different attention-driven feature capturing models to jointly attend to the information from different representation subspaces at different positions. Thus, it avoids averaging caused by a single attention model to inhibit the different subspace representation. Thus, the attention mechanism during video salient object detection is still an on going issue and has a space to improving.

3. Methodology

The structure of the DSSANet is shown in Fig. 1, the input of DSSANet is a 3*D* video dataset $X \in R^{D\times W\times H\times C}$, and the output is a 3*D* segmentation image set $Y \in R^{D\times W\times H\times C}$, which includes n binary masks where the salient object is segmented from each frame of the video. DSSANet consists of three different functional parts: (1) The context feature path Con_1 encodes the enriched global context information of the salient object from X through temporal variation learning with a long-range dependency modeling. The output of the context feature path is $\mathcal{F}_{Con1} \in R^{D\times W\times H\times 256}$, which includes a comprehensive representation to handle all variabilities of the salient object between frames in X. (2) The content feature path Con_2 encodes the detailed local spatial content information of the salient object from X through pixel correlation learning with a multi-scale structure modeling. The output of the content feature path is $\mathcal{F}_{Con2} \in R^{D\times W\times H\times 256}$, which includes a fine representation to handle all the diversity of interrelationships and interdependencies of each pixel in the salient object and the surrounding objects from X. (3) The refinement fusion module fuses and further refines the feature $\mathcal{F}_{Con1}$ and the feature $\mathcal{F}_{Con2}$ as the final feature $\mathcal{F}$ to segment the salient object. The output of the refinement fusion module is Y.

3.1. Context feature path for long-range temporal variation learning

The context feature path innovatively leverages eight attention-augmented ConvLSTM layers with residual blocks to build an encoder–decoder framework to learn the long-range temporal variation of the salient object effectively. This path uses four attention-augmented ConvLSTM layers with a spatiotemporal scaling step of 2 to sequentially encode the X from the high-resolution to the low-resolution. Then four attention-augmented ConvLSTM layers are used to build the decoder. Moreover, five residual blocks and skip connections are added between the encoder and decoder to ensure the effective learning of the high-level and abstract context information, as well as avoid the gradient vanishing issue of this context information in this deep network.

The core benefit of this path is to augment the attention mechanism in traditional ConvLSTM, as shown in Fig. 2. This attention-augmented ConvLSTM model extents the temporal dependency modeling from the short-range to the long-range, thereby handling the salient object variabilities (size and location). Concretely, attention-augmented ConvLSTM layer embeds the attention mechanism [33] into the convolutional and recurrence input gate, as well as the output gate of the channel-wise in ConvLSTM. Attention mechanism modifies the initial hidden state h in ConvLSTM $h \in R^{C\times N}$ to an attention-driven $h^{atte} \in R^{C\times N}$, which multiplies the output of the attention layer and adds back the input feature map: $h_i^{atte} = W_h^{atte} x_i + h_i$. The $x = (x_1, \ldots, x_j, \ldots, x_N) \in {}^{C\times N}$ is the output of attention layer in this state:

$$h_j^{atte} = \sum_{j=1}^{N} \beta(i,j) f(x_i), \text{ where } f(x_j) = W_h x_j, \tag{1}$$

where $\beta(i,j) = (\sum_{j=1}^{N} e^{\theta(x_i)^{\mathrm{T}}\phi(x_j)})^{-1} e^{\theta(x_i)^{\mathrm{T}}\phi(x_j)}$ is the weight of ith location (both spatial and temporal dimensional) for synthesizing the jth region, the $\theta(h) = W_\theta$ and the $\phi(h) = W_\phi h$.

3.2. Content feature path for multi-scale pixel correlation learning

The content feature path creatively stacks dilated 1D convolutions as multi-scale extractors to respectively focus on the pixel correlation between the salient object and other pixels in the video. This path initially stacks six dilated convolutions, and the corresponding dilation rate is [1, 1, 2, 4, 6, 8]. This setting allows the learned representation to include all 3×3 to 65×65 receptive fields of pixel scales. Note that the stack number still varies with the resolution of the video during learning.

The core benefit of this path is to embed the attention mechanism in dilated 1D convolutions, as shown in Fig. 2. Stacked dilated convolutions [38] as multi-scale extractors focus the correlation of the pixels in the 3D dataset. Dilated convolution consists of sparse filters that use skip points during convolution to increase the receptive field and aggregate multi-scale content features exponentially. Embedding the attention mechanism further selects and weighs the content features of the salient object, as

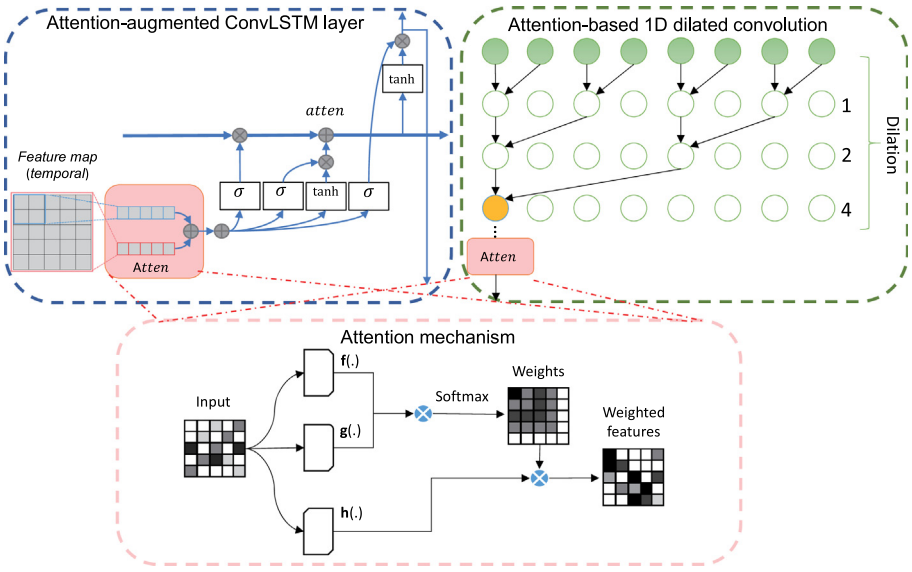

Fig. 2. The attention mechanism computes the response at a position as a weighted sum of the features at all positions. It can select the task-specific feature for each path. By embedding into the attention mechanism, the attention-augmented convolutional LSTM focuses on capturing the global temporal dependencies between and within frames over the video; attention-based 1D convolutional network focus on to capture the local pixel correlation structure of each pixel in the salient object and the surrounding objects.

well as avoids the interference from the features of background objects in the video, thereby improving the learning accuracy of salient objects. Concretely, The 1D dilated convolutions are formulated as:

$$\mathbf{1D}: (kernel *_l 1D)_t = \sum_{s=-\infty}^{\infty} kernel_s \cdot 1D_{t-ls}, \tag{2}$$

where $1D$ is the 1D pixels, and l is the dilation rater. By the same token, attention mechanism modifies the output feature $y \in R^{C\times N}$ of dilated 1D convolutions to an attention-driven $y^{atten} \in R^{C\times N}$

$$y_j^{atten} = \sum_{j=1}^{N} \beta(i,j) f(y_i). \tag{3}$$

3.3. Refinement fusion module for accurate salient object segmentation

The refinement fusion module firstly concatenates two type features from the above two paths (the context feature path and the content feature path), and then creates a 3D residual convolution network with the attention mechanism to fuse and refine the concatenated feature to segment the salient target. This module efficiently fuses features with multiple levels and refines them using feature weights, thereby obtaining accurate segmentation of the salient target. In particular, this module is built as a chain of 4 3D Convolutional blocks, each block consisting of two 3×3 convolution layer, and the self-attention layer is integrated after the first 3D Convolutional block. Therefore, the module weighs features by computing the response of a feature through a weighted sum of this feature at all features. Note that, the residual connections also build in the convolutional block, which once again facilitates gradient propagation during training. This module also leverages a 3D Convolutional layer with $1 \times 1 \times 1$ kernel to output the segmentation result. This layer is a non-linearity operation on the feature to ensure that the channel size of the final output is consistent with the size of the ground truth.

3.4. Synergistic loss term for accurate and stable learning

A synergistic loss term $\mathcal{L}$ is designed to greatly improve the learning accuracy and stability by integrating four complementary loss terms of the weighed cross-entropy loss $\mathcal{L}_{wce}$, the generalized dice loss $\mathcal{L}_{dice}$, and the mean squared error (MSE) loss $\mathcal{L}_{mse}$. This synergistic loss avoids the interference of the background pixels on the pixels of the salient object during training caused by a great imbalance between the numbers of two type pixels. The synergistic loss term can be formulated as follows:

$$\mathcal{L} = \omega_1 \mathcal{L}_{wce} + \omega_2 \mathcal{L}_{dice} + \omega_3 \mathcal{L}_{mae}, \tag{4}$$

Weighted cross-entropy loss $\mathcal{L}_{wce}$ can be formulated as follows:

$$\mathcal{L}_{wce} = -(\omega_4 p \log(\hat{p}) + (1-p)\log(1-\hat{p})), \tag{5}$$

where p is the ground truth for $\hat{p}$, and $\omega_4 = 1.5$. *Generalized dice loss* $\mathcal{L}_{dice}$ can be formulated as follows:

$$\mathcal{L}_{dice} = 1 - N\frac{\sum_{l=1}^{N} w_l \sum_n r_{ln} p_{ln}}{\sum_{l=1}^{2} w_l \sum_N r_{ln} + p_{ln}}, \tag{6}$$

where N is the number of salient objects, and $w_l = \frac{1}{\left(\sum_{n=1}^{N} r_{ln}\right)^2}$.

MSE loss $\mathcal{L}_{mse}$ can be formulated as follows:

$$\mathcal{L}_{mse} = \frac{\sum_{i=1}^{n} \left(p_i - \hat{p}_i\right)^2}{n}. \tag{7}$$

4. Experiments setting

4.1. Benchmark datasets

The DSSANet has evaluated its performance using four public benchmark datasets: SegTrack2 dataset [39], DAVIS dataset [40], FBMS dataset [41], and VOS dataset [19].

SegTrack2 dataset contains 14 video sequences and is originally designed for video object segmentation. It has a total of 1065 frames and an average of 76 frames per video. This dataset has the challenge that the great imbalance between background pixels and object pixels, and the average object pixel number only accounts for 7% of each frame.

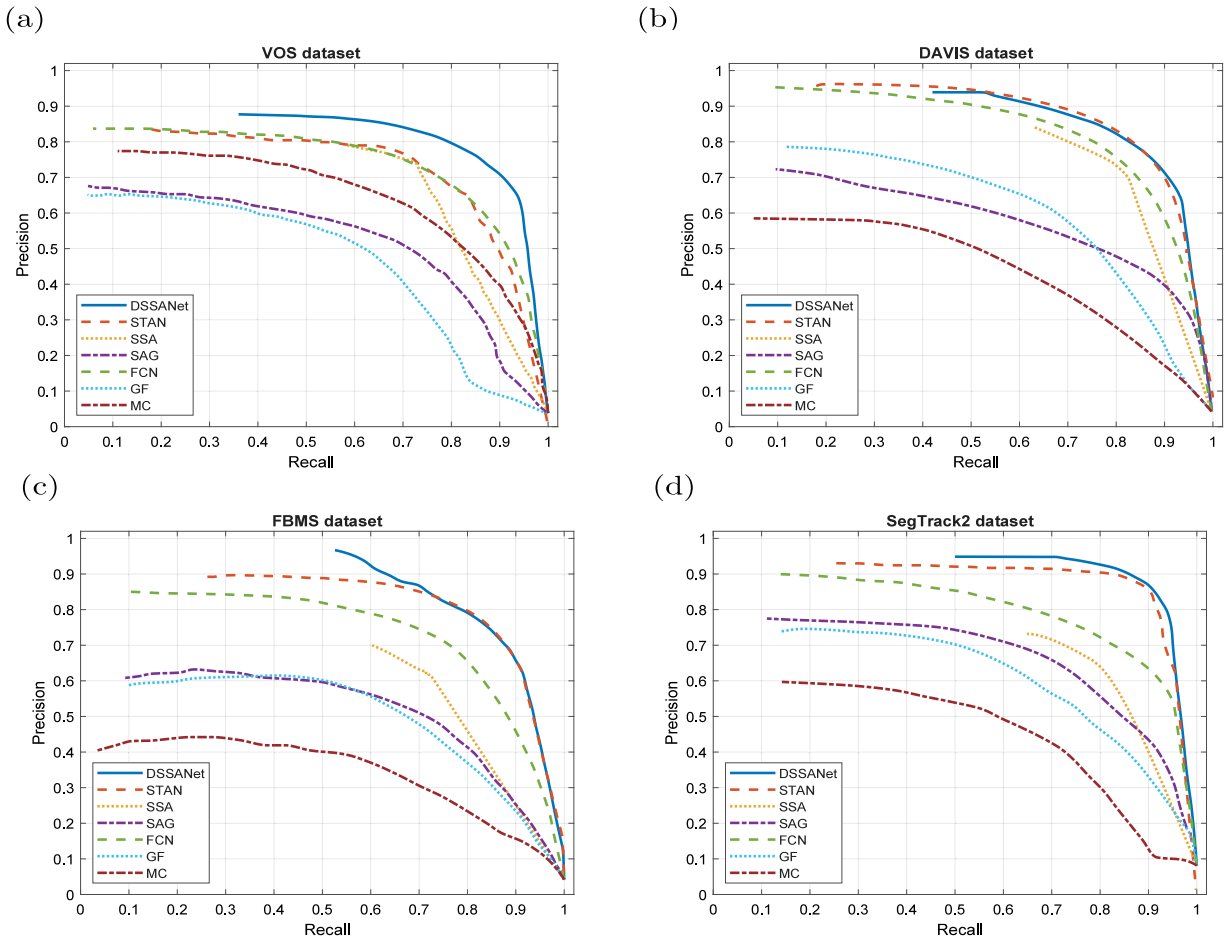

Fig. 3. The DSSANet achieved better precision scores and recall scores than the existing state-of-the-art methods (STAN, SSA, SAG, FCN, GF, and MC). In the precision–recall curve, the *x*-axis shows recall, the *y*-axis shows precision, a large area under the curve represents the high performance of the model. Each sub-figure corresponds to the results computed in a dataset ((a) VOS dataset, (b) DAVIS dataset, (c) FBMS dataset, and (d) SegTrack2 dataset).

DAVIS dataset contains 50 video sequences. It has a total of 3455 frames and an average of 69 frames per video. This dataset is a Full HD 1080p resolution and has five objects per frame. Thus, this dataset is challenging since it has a high resolution, and fast motion. The average object pixel number accounts for 9% of each frame.

FBMS dataset contains 59 video sequences. It has a total of 13 860 frames and an average of 235 frames per video. This dataset is challenging due to color similarity. It has 1.7 objects per frame, and the average object's pixel number accounts for 14% of each frame.

VOS dataset contains 200 video sequences and is divided into two subsets VOS-E and VOS-N. It has a total of 116 103 frames and an average of 581 frames per video. This dataset is challenging because of the extreme length of the video and the resulted complex background change and object motion. The average object pixel number accounts for 13% of each frame.

4.2. Evaluation metrics

The DSSANet leverages three types of metrics, which are precision–Recall curve, F-measure, and mean absolute error (MAE) for the performance evaluation.

Precision–recall curve is computed using the pairs of precision scores and recall scores. Both scores are calculated as follows:

$$Precision = \frac{TP}{TP + FP} \qquad Recall = \frac{TP}{TP + FN}, \tag{8}$$

where TP is the pixel number of true positives, FP is the pixel number of false positives, and FN is the pixel number of false negatives.

F-measure is also based on precision scores and recall scores. After the mean average recall (MAR) and the mean average precision (MAP) are computed [19], the F-measure is calculated as follows:

$$\begin{aligned} F - measure &= \frac{\left(1 + \beta^2\right) \cdot \mathrm{MAP} \cdot \mathrm{MAR}}{\beta^2 \cdot \mathrm{MAP} + \mathrm{MAR}}, \\ MAP &= \frac{\sum_1^K \frac{\sum_1^V Precision}{V}}{K}, \\ MAR &= \frac{\sum_1^K \frac{\sum_1^V Recall}{V}}{K}, \end{aligned} \tag{9}$$

where β^2 is a hyperparameter and it is set as 3 according to [42]. V is the number of frames in each video. K is the number of videos.

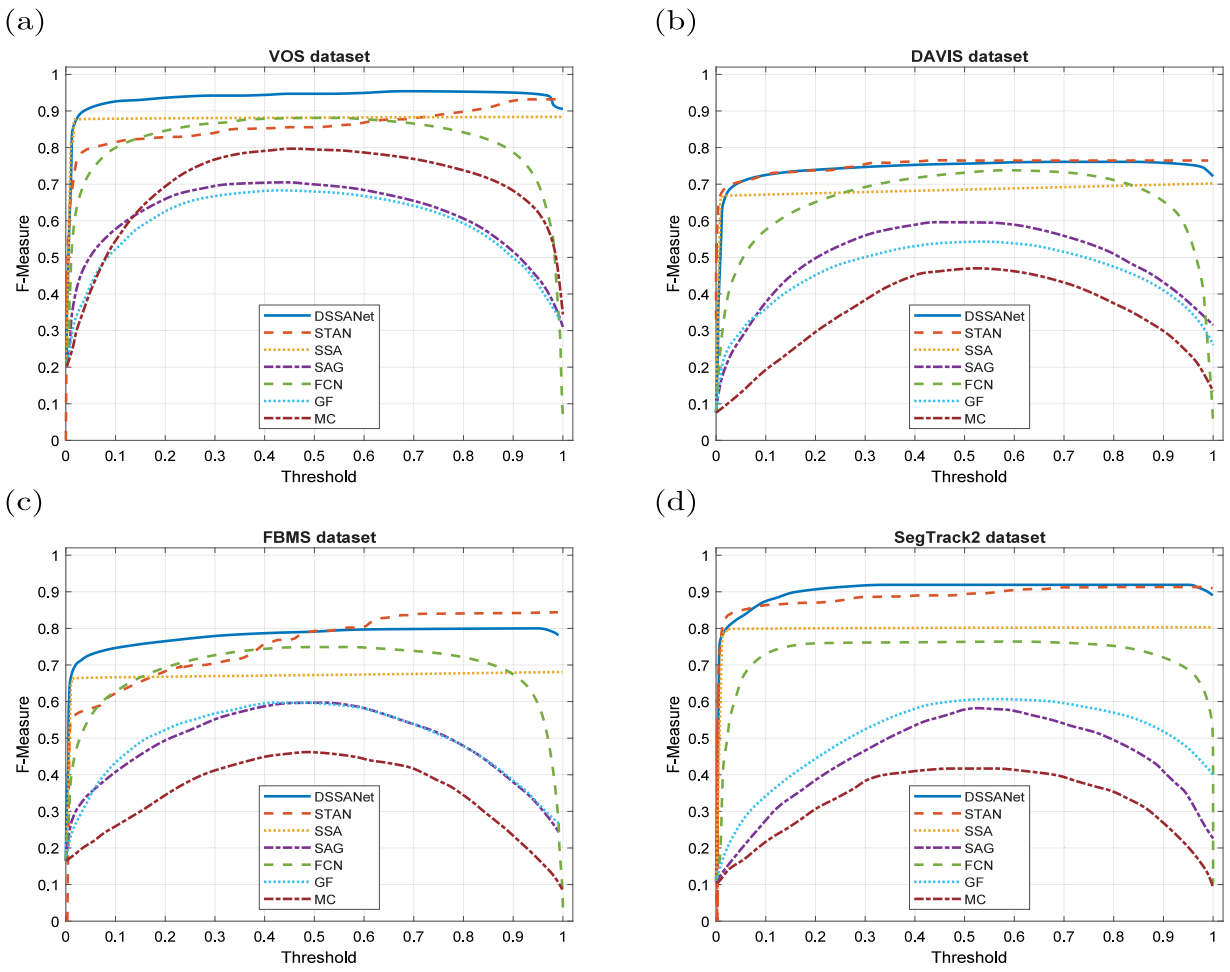

Fig. 4. The DSSANet achieved better F-measure scores than the existing state-of-the-art methods (STAN, SSA, SAG, FCN, GF, and MC). In F-measure curve, the *x*-axis shows F-measure, *y*-axis shows showing threshold, a large area under the curve represents the high performance of the model. Each sub-figure corresponds to the results computed in each dataset ((a) VOS dataset, (b) DAVIS dataset, (c) FBMS dataset, and (d) SegTrack2 dataset).

Mean absolute error is computed by summing of absolute differences between our target and predicted variables. It is calculated as follows:

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |p_i - \hat{p}_i|. \tag{10}$$

4.3. Implementation details

The DSSANet is implemented by Python with TensorFlow library and 2 × NVIDIA P100 GPU cards for training. It is optimized by the ADAM solver [43] with a batch size of 32 and an initial learning rate of 0.001. Batch normalization [44] and LeakyReLU activation [44] are applied. The hyper-parameter settings are $\omega_1 = 1$, $\omega_2 = 10$, and $\omega_3 = 5$.

5. Experimental results

To evaluate the performance of our DSSANet, we compare it with six state-of-the-art video salient object detection methods including two-stream based spatiotemporal attention network (STAN) [11], saliency-guided stacked autoencoders (SSA) [19], geodesic distance-based saliency (SAG) [45], saliency using fully convolutional networks (FCN) [13], saliency using local gradient flow (GF) [21], and saliency using absorbing Markov Chain (MC) [22] on the four benchmark datasets mentioned above. All the methods are designed for video salient object detection except MC, and STAN and FCN are the deep models among them.

5.1. Outperformance over the state-of-the-art methods

Figs. 3, 4, and 5 demonstrate that the DSSANet achieves better performance than the existing state-of-the-art methods (STAN, SSA, SAG, FCN, GF, and MC).

- Fig. 3 indicates that DSSANet outperforms those methods in precision scores and recall scores. In the precision–recall curve, the *x*-axis shows recall, the *y*-axis shows precision, a large area under the curve represents the high performance of the model. By comparison, the precision–recall curve of the DSSANet (Blue solid line) has an overall larger area under the curve in all four datasets.
- Fig. 4 indicates that the DSSANet outperforms those of methods in F-measure. In the F-measure curve, the *x*-axis shows F-measure, the *y*-axis shows threshold, (adaptive computing by [46]), a large area under the curve represents the high performance of the model. By comparison, the F-measure curve of the DSSANet (Blue solid line) has an overall larger area under the curve in all four datasets.
- Fig. 5 indicates that the DSSANet outperforms those of methods in MAE. In the MAE bar image, the *x*-axis shows MAE and the *y*-axis shows different methods, a lower MAE value represents the high performance of the model. By comparison, the MAE of the DSSANet (steel blue) has lower values in all three datasets.

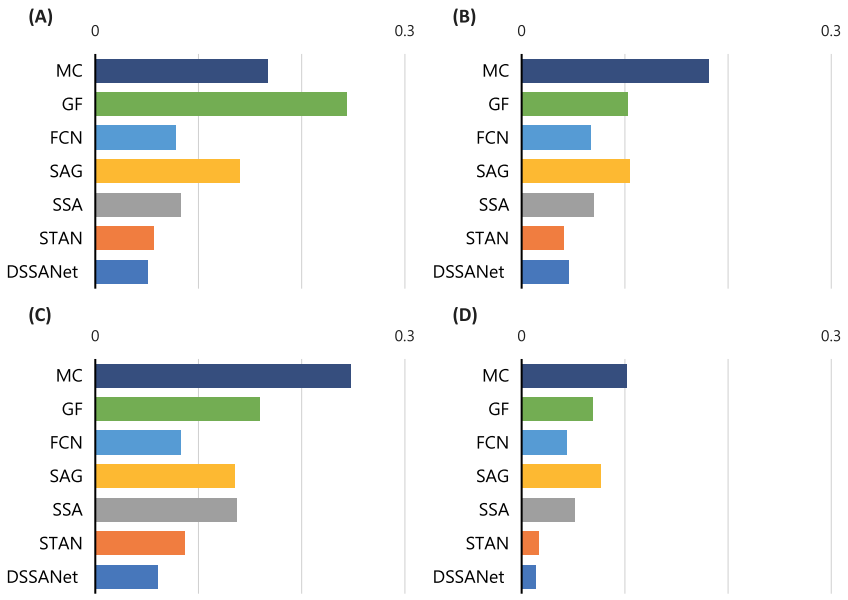

Fig. 5. The DSSANet achieves better MAE than the existing state-of-the-art methods (STAN, SSA, SAG, FCN, GF, and MC). In the MAE bar image, *x*-axis shows MAE and the *y*-axis shows different methods, a lower MAE value represents the high performance of the model. Each sub-figure corresponds to the results computed in each dataset ((A) VOS dataset, (B) DAVIS dataset, (C) FBMS dataset, and (D) SegTrack2 dataset).

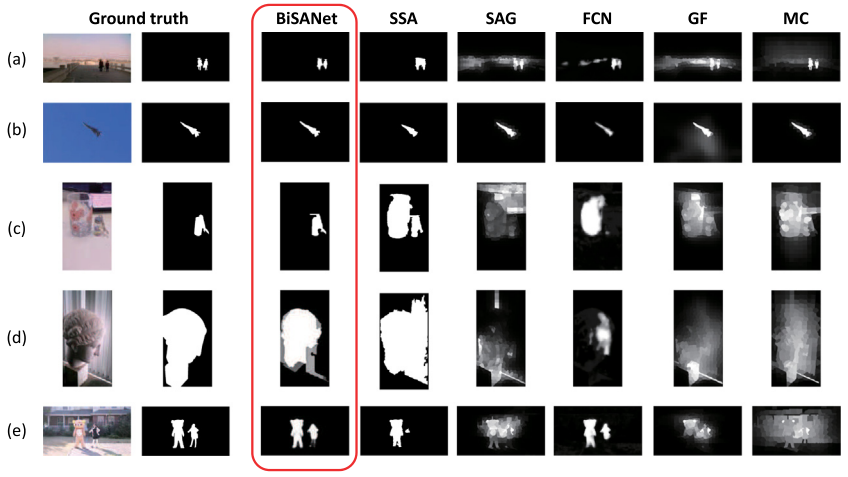

Fig. 6. Representative results to show the better performance of the DSSANet than the state-of-the-art methods in different aspects: (a) Great imbalance between background and object. (b) Fast motion. (c) and (d) Color similarity. (e) Multi-objects.

5.2. Better performance in the representative results

Fig. 6 displays some representative results to qualitative evaluate the performance of DSSANet from different aspects:

- Great imbalance between background and object. Fig. 6(a) shows an example that presents a great imbalance between background pixel number and object pixel number in the video. The DSSANet obtains better performance in avoiding the influence of background sky and water than the comparative methods concerning the ground truth.
- Fast motion. Fig. 6(b) shows an example that presents a fast-moving object in the video. The DSSANet obtains better performance in capturing the boundaries of the jet plane in all locations of the video than the comparative methods concerning the ground truth.
- Color similarity. Fig. 6(c) shows an example that presents two high color similarity objects in video and only one is the salient object. The DSSANet obtains better performance in identifying the salient object than the comparative methods concerning the ground truth.
- Multi-objects. Fig. 6(e) shows an example that presents two changed objects. The DSSANet obtains better performance in segmenting multi-objects than the comparative methods concerning the ground truth.

5.3. Time performance

Fig. 7 represents the time performance of our DSSANet and the existing state-of-the-art methods (STAN, SSA, SAG, FCN, GF, and MC). Note that the SSA, SAG, GF, and MC are traditional models. Our DSSANet, STAN, and FCN are deep learning models, which require extra training stages. All experiments are performed on the same environment with 2 × Intel E5-2683 (v4 Broadwell @ 2.1GH), 2 × NVIDIA P100 Pascal (12 GB HBM2 memory) and 24G available memory. All model has the same input: 8 continuous 480p video frames. We record the time consuming of each model (the average second of each frame) and the detection performance of each model (MAE). The results demonstrate that the

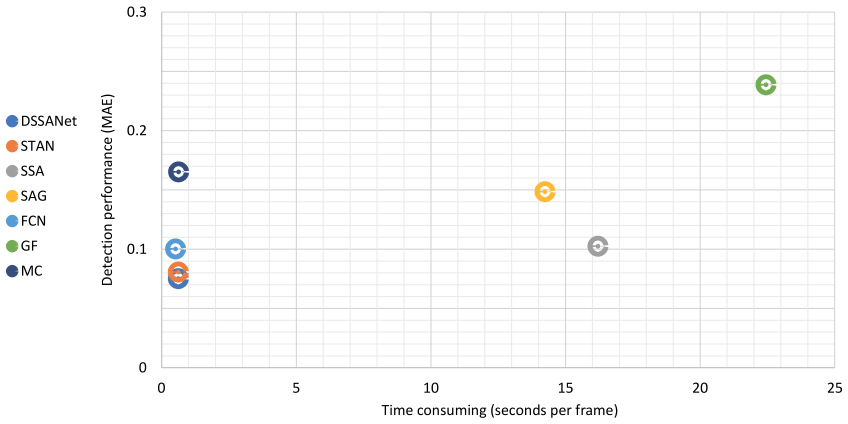

Fig. 7. DSSANet achieves overall highest detection performance while acceptable a time consuming compared to the existing state-of-the-art methods (STAN, SSA, SAG, FCN, GF, and MC). In Figure, the *x*-axis shows detection performance, the *y*-axis shows times consuming, a higher detection performance with the lower time consuming represents the high performance of the model.

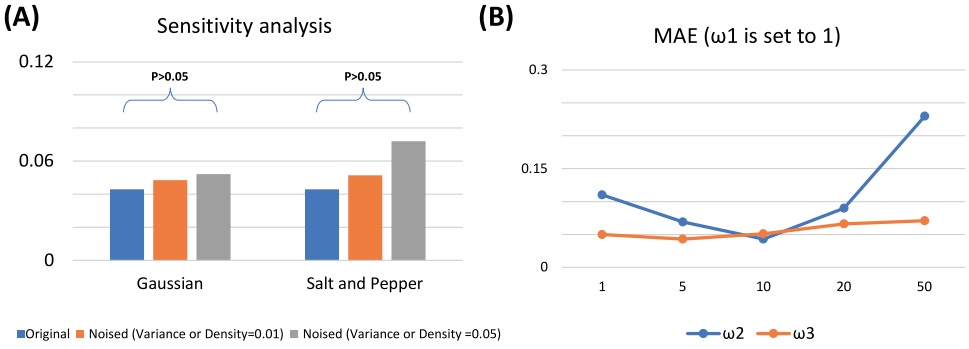

Fig. 8. Sensitivity analysis represents that DSSANet achieves strong robustness under noise introduced (A). Moreover, parameters tuning indicates that setting ω_2 as 10 and setting ω_3 as 5 achieve overall the highest performance when ω_1 is set to 1 (B).

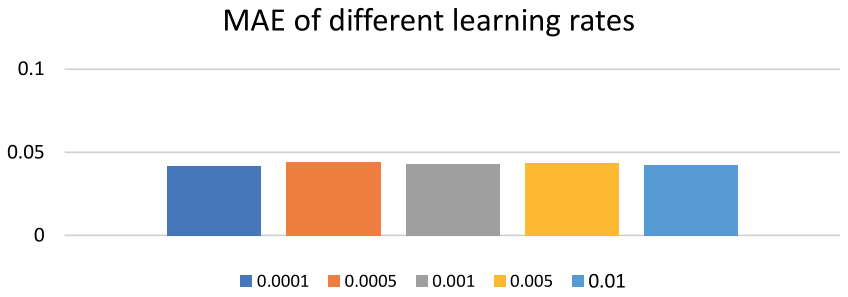

Fig. 9. The MAE of different learning rates indicates different learning rates make no significant difference under our network architecture with Adam optimizer.

DSSANet achieves overall highest detection performance while acceptable a time consuming compared to the existing state-of-the-art methods (STAN, SSA, SAG, FCN, GF, and MC). In Fig. 6, the *x*-axis shows detection performance, the *y*-axis shows times consuming, a higher detection performance with the lower time consuming represents the high performance of the model.

5.4. Sensitivity analysis and parameters tuning

Fig. 8 represents the sensitivity of our DSSANet under noises. We have inserted Gaussian noise (mean of 0 and variances of [0.01, 0.05]), and Salt and Pepper noise (noise densities of [0.01, 0.05]) in each image. The results demonstrate that the DSSANet achieves overall consistency ($P > 0.05$) between these noised images and the original images. Moreover, Fig. 6 represents hyper parameters ω_1, ω_2 and ω_3 tuning and how sensitive the results are. We fix hyperparameters ω_1 and tune the ω_2 and ω_3 between [1, 5, 10, 20, 50]. The experimental results indicate that setting ω_2 as 10 and setting ω_3 as 5 achieve overall the highest performance. Fig. 9 indicates different learning rates make no significant difference under our network architecture with Adam optimizer. This because Adam optimizer is adaptive estimates of lower-order moments, which can be adaptively suited for problems that are large in terms of parameters without tuning [43].

5.5. Limitations and further directions

DSSANet still has several limitations: (1) high computational cost. The attention model has large memory requirements due to its large affinity matrix in high-resolution images. DSSANet embeds multiple attention models in both video feature extraction and selection, which further increases the memory requirements several times. This causes that we have to use 2 *times* NVIDIA P100 GPU card for training. In the future, we plan to adapt

recent model compression technology to reduce the memory requirement, thereby training the DSSANet in a moderate computational cost while maintaining the accuracy. (2) Bad cases in small and fast-moving objects. DSSANet remains challenging to obtain precise detection results for small and fast-moving objects. This because the spatiotemporal features of these objects are too weak and easily interfered with by the spatiotemporal feature of surrounding objects. Furthermore, the detection of small and fast-moving objects still a well-known open challenge. In the future, we plan to improve the detection performance of small and fast-moving objects by applying new-designed constraints or loss-terms.

6. Conclusion

Video salient object detection is crucial in the current computer vision community. It has experienced significant advances with deep learning technology and becomes more prevalent. In this paper, we propose a DSSANet for saliency detection in videos. The DSSANet consists of (1) a context feature path captures global context information of object over the video; (2) a content feature path captures local spatial content information of the salient object; (3) a refinement fusion module then fuses these two types of features and further refines the feature representations, thereby accurately segmenting the salient object. By training and testing on four public datasets with different characteristics, the experimental results demonstrate the effectiveness of DSSANet, as well as its superiority to the five state-of-the-art methods.

CRediT authorship contribution statement

Chenchu Xu: Conceptualization, Methodology, Software, Writing - Original Draft, Writing - Review & Editing. **Zhifan Gao:** Conceptualization, Methodology, Software, Validation, Writing - Review & Editing. **Heye Zhang:** Data curation, Supervision, Visualization, Investigation, Writing - Review & Editing. **Shuo Li:** Supervision, Writing - Review & Editing. **Victor Hugo C. de Albuquerque:** Supervision, Project administration, Software, Writing - Review & Editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported in part by grant from the Key-Area Research and Development Program of Guangdong Province (2019B010110001), Shenzhen Innovation Funding(KCXFZ20202011009124), and the Fundamental Research Funds for the Central Universities(19lgzd36).

References

[1] Y. Luo, J. Yuan, P. Xue, Q. Tian, Salient region detection and its application to video retargeting, in: 2011 IEEE International Conference on Multimedia and Expo, IEEE, 2011, pp. 1–6.
[2] Z. Xu, H.R. Wu, Smart video surveillance system, in: 2010 IEEE International Conference on Industrial Technology, IEEE, 2010, pp. 285–290.
[3] C. Xu, L. Xu, Z. Gao, S. Zhao, H. Zhang, Y. Zhang, X. Du, S. Zhao, D. Ghista, H. Liu, et al., Direct delineation of myocardial infarction without contrast agents using a joint motion feature learning architecture, Med. Image Anal. 50 (2018) 82–94.
[4] E. Rahtu, J. Kannala, M. Salo, J. Heikkilä, Segmenting salient objects from images and videos, in: European Conference on Computer Vision, Springer, 2010, pp. 366–379.
[5] A. Borji, M.-M. Cheng, H. Jiang, J. Li, Salient object detection: A benchmark, IEEE Trans. Image process. 24 (12) (2015) 5706–5722.
[6] A. Borji, M.-M. Cheng, Q. Hou, H. Jiang, J. Li, Salient object detection: A survey, Comput. Vis. Media (2014) 1–34.
[7] H. Jiang, J. Wang, Z. Yuan, Y. Wu, N. Zheng, S. Li, Salient object detection: A discriminative regional feature integration approach, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2013, pp. 2083–2090.
[8] T. Liu, Z. Yuan, J. Sun, J. Wang, N. Zheng, X. Tang, H.-Y. Shum, Learning to detect a salient object, IEEE Trans. Pattern Anal. Mach. Intell. 33 (2) (2010) 353–367.
[9] R. Girshick, J. Donahue, T. Darrell, J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 580–587.
[10] D. Tran, L. Bourdev, R. Fergus, L. Torresani, M. Paluri, Learning spatiotemporal features with 3d convolutional networks, in: Proceedings of the IEEE international conference on computer vision, 2015, pp. 4489–4497.
[11] Y. Tang, W. Zou, Y. Hua, Z. Jin, X. Li, Video salient object detection via spatiotemporal attention neural networks, Neurocomputing 377 (2020) 27–37.
[12] H. Song, W. Wang, S. Zhao, J. Shen, K.-M. Lam, Pyramid dilated deeper convlstm for video salient object detection, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 715–731.
[13] W. Wang, J. Shen, L. Shao, Video salient object detection via fully convolutional networks, IEEE Trans. Image Process. 27 (1) (2017) 38–49.
[14] H. Li, G. Chen, G. Li, Y. Yu, Motion guided attention for video salient object detection, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 7274–7283.
[15] G. Li, Y. Xie, T. Wei, K. Wang, L. Lin, Flow guided recurrent neural encoder for video salient object detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 3243–3252.
[16] T.-N. Le, A. Sugimoto, Video salient object detection using spatiotemporal deep features, IEEE Trans. Image Process. 27 (10) (2018) 5002–5015.
[17] S. Dong, Z. Gao, S. Pirbhulal, G.-B. Bian, H. Zhang, W. Wu, S. Li, IoT-Based 3d convolution for video salient object detection, Neural Comput. Appl. 32 (3) (2020) 735–746.
[18] Z. Fang, T. Cao, J. Yang, Y. Xing, Dense dilation network for saliency detection, in: Tenth International Conference on Graphics and Image Processing (ICGIP 2018), Vol. 11069, International Society for Optics and Photonics, 2019, 110691V.
[19] 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) (2017) 349–364.
[20] T.-N. Le, A. Sugimoto, Contrast based hierarchical spatial–temporal saliency for video, in: Image and Video Technology, Springer, 2015, pp. 734–748.
[21] 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.
[22] B. Jiang, L. Zhang, H. Lu, C. Yang, M.-H. Yang, Saliency detection via absorbing markov chain, in: Proceedings of the IEEE international conference on computer vision, 2013, pp. 1665–1672.
[23] X. Wang, L. Gao, J. Song, H. Shen, Beyond frame-level cnn: saliency-aware 3-d cnn with lstm for video action recognition, IEEE Signal Process. Lett. 24 (4) (2016) 510–514.
[24] Z. Gao, Y. Li, Y. Sun, J. Yang, H. Xiong, H. Zhang, X. Liu, W. Wu, D. Liang, S. Li, Motion tracking of the carotid artery wall from ultrasound image sequences: a nonlinear state-space approach, IEEE Trans. Med. Imaging 37 (1) (2017) 273–283.
[25] S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, K. Saenko, Sequence to sequence-video to text, in: Proceedings of the IEEE international conference on computer vision, 2015, pp. 4534–4542.
[26] K. Muhammad, S. Khan, V. Palade, I. Mehmood, V.H.C. De Albuquerque, Edge intelligence-assisted smoke detection in foggy surveillance environments, IEEE Trans. Ind. Inf. 16 (2) (2019) 1067–1075.
[27] B. Fan, L. Xie, S. Yang, L. Wang, F.K. Soong, A deep bidirectional lstm approach for video-realistic talking head, Multimedia Tools Appl. 75 (9) (2016) 5287–5309.
[28] K. Muhammad, T. Hussain, J. Del Ser, V. Palade, V.H.C. De Albuquerque, Deepres: A deep learning-based video summarization strategy for resource-constrained industrial surveillance scenarios, IEEE Trans. Ind. Inf. 16 (9) (2019) 5938–5947.
[29] S. Zha, F. Luisier, W. Andrews, N. Srivastava, R. Salakhutdinov, Exploiting image-trained cnn architectures for unconstrained video classification, arXiv preprint arXiv:1503.04144.
[30] K. Muhammad, T. Hussain, M. Tanveer, G. Sannino, V.H.C. de Albuquerque, Cost-effective video summarization using deep cnn with hierarchical weighted fusion for iot surveillance networks, IEEE Internet Things J. 7 (5) (2019) 4455–4463.

[31] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, W.-c. Woo, Convolutional lstm network: A machine learning approach for precipitation nowcasting, in: Adv. Neural Inf. Process. Syst., 2015, pp. 802–810.
[32] A.M. Rush, S. Chopra, J. Weston, A neural attention model for abstractive sentence summarization, arXiv preprint arXiv:1509.00685.
[33] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: Adv. Neural Inf. Process. Syst., 2017, pp. 5998–6008.
[34] F. Sun, W. Li, Y. Guan, Self-attention recurrent network for saliency detection, Multimedia Tools Appl. (2018) 1–15.
[35] Y. Ji, H. Zhang, Q.J. Wu, Salient object detection via multi-scale attention cnn, Neurocomputing 322 (2018) 130–140.
[36] X. Wang, R. Girshick, A. Gupta, K. He, Non-local.neural. networks,
[37] R.R.A. Pramono, Y.-T. Chen, W.-H. Fang, Hierarchical self-attention network for action localization in videos, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 61–70.
[38] F. Yu, V. Koltun, Multi-scale context aggregation by dilated convolutions, arXiv preprint arXiv:1511.07122.
[39] K. Fukuchi, K. Miyazato, A. Kimura, S. Takagi, J. Yamato, Saliency-based video segmentation with graph cuts and sequentially updated priors, in: 2009 IEEE International Conference on Multimedia and Expo, IEEE, 2009, pp. 638–641.
[40] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, A. Sorkine-Hornung, A benchmark dataset and evaluation methodology for video object segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 724–732.
[41] P. Ochs, J. Malik, T. Brox, Segmentation of moving objects by long term video analysis, IEEE Trans. Pattern Anal. Mach. Intell. 36 (6) (2013) 1187–1200.
[42] R. Achanta, S. Hemami, F. Estrada, S. Süsstrunk, Frequency-tuned salient region detection, in: IEEE International Conference on Computer Vision and Pattern Recognition (CVPR 2009), CONF, 2009, pp. 1597–1604.
[43] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980.
[44] I. Goodfellow, Y. Bengio, A. Courville, Y. Bengio, Deep Learning, Vol. 1, MIT press Cambridge, 2016.
[45] W. Wang, J. Shen, F. Porikli, Saliency-aware geodesic video object segmentation, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 3395–3402.
[46] Y. Jia, M. Han, Category-independent object-level saliency detection, in: Proceedings of the IEEE international conference on computer vision, 2013, pp. 1761–1768.