Pyramid Dilated Deeper ConvLSTM for Video Salient Object Detection

Hongmei Song[1⋆], Wenguan Wang[1⋆][0000−0002−0802−9567], Sanyuan Zhao[1⋆⋆], Jianbing Shen[1,2], and Kin-Man Lam[3]

[1] Beijing Lab of Intelligent Information Technology, School of Computer Science, Beijing Institute of Technology, China
[2] Inception Institute of Artificial Intelligence, Abu Dhabi, UAE
[3] The Hong Kong Polytechnic University, Kowloon, Hong Kong
{songhongmei,zhaosanyuan,shenjianbing}@bit.edu.cn
wenguanwang.ai@gmail.com enkmlam@polyu.edu.hk
https://github.com/shenjianbing/PDB-ConvLSTM

Abstract. This paper proposes a fast video salient object detection model, based on a novel recurrent network architecture, named Pyramid Dilated Bidirectional ConvLSTM (PDB-ConvLSTM). A Pyramid Dilated Convolution (PDC) module is first designed for simultaneously extracting spatial features at multiple scales. These spatial features are then concatenated and fed into an extended Deeper Bidirectional ConvLSTM (DB-ConvLSTM) to learn spatiotemporal information. Forward and backward ConvLSTM units are placed in two layers and connected in a cascaded way, encouraging information flow between the bi-directional streams and leading to deeper feature extraction. We further augment DB-ConvLSTM with a PDC-like structure, by adopting several dilated DB-ConvLSTMs to extract multi-scale spatiotemporal information. Extensive experimental results show that our method outperforms previous video saliency models in a large margin, with a real-time speed of **20 fps** on a single GPU. With unsupervised video object segmentation as an example application, the proposed model (with a CRF-based post-process) achieves state-of-the-art results on two popular benchmarks, well demonstrating its superior performance and high applicability.

1 Introduction

Video saliency detection aims at finding the most interesting parts in each video frame that mostly attract human attention. It can be applied as a fundamental module in many visual tasks, such as video object segmentation, scene rendering, object tracking, and so on. Similar to visual saliency detection in static images, research on video saliency detection can also be divided into two categories, *i.e.*, eye fixation prediction [41, 39] and salient object detection [49, 47]. The purpose of eye fixation prediction is to locate the focus of human eyes when looking at

⋆ *Hongmei Song* and *Wenguan Wang* contributed equally.
⋆⋆ Corresponding author: *Sanyuan Zhao*.

a scene, which is helpful for understanding the mechanism of biological visual attention. Salient object detection focuses on uniformly highlighting the most salient objects with clear contour. In this paper, we focus on the latter task.

Most existing video saliency methods [8, 42, 43, 11], are built upon shallow, hand-crafted features (*e.g.*, color, edge, *etc.*), and especially rely on motion information from optical flow. These methods are typically heuristic and suffer from slow speed (due to time-consuming optical flow computation) and low prediction accuracy (due to the limited representability of low-level features). Currently, only a few works [44, 24] on video saliency detection, based on deep learning, can be found in the literature. For example, Wang *et al*. [44] proposed a fully convolutional network (FCN) based video saliency model, as a very early attempt towards an end-to-end deep learning solution for this problem, achieves a speed of 2 fps. In their method, temporal dynamics between only two adjacent video frames are considered. Clearly, it faces difficulties to achieve a real-time speed and lacks exploration of motion information from a longer time span.

To employ deep learning techniques for video saliency detection, two problems should be considered [44]. The first problem is how to describe the temporal and spatial information and how to combine them together. Optical flow offers explicit motion information, but also incurs significant computational cost, which severely limits the applicability of current video saliency models. The second problem is data. A sufficiently large, densely labeled video saliency training data is desirable, but hard to obtain. Wang *et al*. [44] synthesize motion frames from static images to enrich the video training data. However, the quality of the synthesized data is unsatisfactory.

To address above issues, first, we base our model upon a convolutional LSTM (ConvLSTM) structure [32], which captures the long and short-term memory of video sequences and contains both temporal and spatial information, for implicitly learning temporal dynamics and efficiently fusing temporal and spatial features. For encouraging information exchange between LSTM units in bi-directions, we propose a Deeper Bidirectional ConvLSTM (DB-ConvLSTM) structure which learns temporal characteristics in a cascaded and deeper way, *i.e.*, the ConvLSTM units in the backward layer are built upon the forward layer (instead of directly connecting to the inputs). Thus the forward ConvLSTM units, each of which corresponds to a specific input frame, can exchange their sequential knowledge with the backward layers. For further improving the spatial learning ability of DB-ConvLSTM, we introduce a multi-scale receptive field module, called Pyramid Dilated Convolution (PDC), into ConvLSTM to obtain more spatial details. Second, we train our model with massive static-image saliency data, in addition to video saliency data. In this way, our network could capture different object appearances which are important for video saliency prediction. Above designs lead to a powerful and very fast deep video saliency model, which achieves state-of-the-art performance on three video datasets [31, 2, 43] with fast speed of 20 fps (all steps on one GPU). With unsupervised video object segmentation as an example application task, we further show that the proposed video saliency model, equipped with a CRF segmentation module,

gains best performance over two popular video segmentation benchmarks (*e.g.*, DAVIS [31] and FBMS [2]), which clearly demonstrates the high applicability of our model.

2 Related Work

Image/Video Salient Object Detection. Recently, with the popularity of deep neural network, various deep learning based image salient object detection models were proposed, *e.g.*, multi-stream network with embedded superpixels [22, 25], recurrent module [26, 40], and multi-scale and hierarchical feature fusion [16, 51, 36], *etc*. These models generally achieve far better performance compared with traditional static saliency models [49, 45].

Conventional video salient object detection methods [8, 9, 28, 42] extract spatial and temporal features separately and then integrate them together to generate a spatiotemporal saliency map. The spatiotemporal result can be refined by various mechanisms [43, 27]. The computational cost, especially for temporal features, is usually expensive. Recently, Wang *et al*. [44] introduced FCN to video salient object detection by using adjacent pairs of frames as input, which substantially improves the precision and achieves a speed of 2 fps. However, this speed is still so slow for real-time processing, and more spatiotemporal information should be explored by considering more frames in video sequences.

Unsupervised Video Segmentation. Some unsupervised video segmentation tasks, like temporal superpixel/supervoxel over-segmentation [48, 3] and motion segmentation [2], are typically based on clustering methods with low-level appearance and motion information. The unsupervised video primary object segmentation [46], which is the most related video segmentation topic to our approach, aims at extracting the primary object(s) in video sequences with the use of object-level information (*e.g.*, object proposal) and various heuristics [38, 10, 7]. Those models have similar goal with video salient object detection, aside from they seeking to get a binary fore-/background mask for each video frame. As demonstrated in [42], video saliency models are able to offer valuable information for guiding video object segmentation. Recent unsupervised video object segmentation methods are mainly based on deep learning models, such as two-stream architecture [19], recurrent neural network [34], bottom-up top-down model [33], FCN network [5], *etc*. In this work, we show our model is well applicable to unsupervised video object segmentation task.

3 Our Approach

This section elaborates on the details of the proposed video salient object detection model, which consists of two key components. The first one, named *Pyramid Dilated Convolution* (PDC) module, is used for explicitly extracting spatial saliency features on multi-scales (as shown in Fig. 2). This is achieved via a set of parallel dilated convolution layers with different sampling rates (§ 3.1).

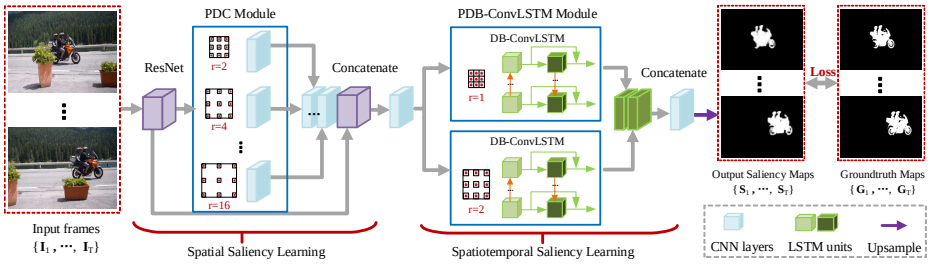

Fig. 1. Architecture overview of the proposed video salient object detection model, which consists of two components, *e.g.*, a spatial saliency learning module based on Pyramid Dilated Convolution (PDC) (§ 3.1) and a spatiotemporal saliency learning module via Pyramid Dilated Bidirectional ConvLSTM (PDB-ConvLSTM) (§ 3.2).

The second module, named *Pyramid Dilated Bidirectional ConvLSTM* (PDB-ConvLSTM), which augments the *vanilla* ConvLSTM with the powerful structure of PDC module and is improved with a cascaded bi-directional feature learning process, *i.e.*, learning deeper, backward information upon forward features. PDB-ConvLSTM takes the spatial features learnt from the PDC module as inputs, and outputs improved spatiotemporal saliency representations for final video salient object prediction (§ 3.2). In § 3.3, detailed implementations of our model are presented.

3.1 Spatial Saliency Learning via PDC Module

A typical CNN model is comprised of a stack of convolution layers, interleaved with non-linear downsampling operation (*e.g.*, max pooling) and point-wise non-linearity (*e.g.*, *ReLU*). Downsampling operation is effective for enlarging the receptive field, but quite harmful for pixel-wise prediction tasks, such as video salient object detection, since too many spatial details are lost. The recently proposed dilated convolution [50] provides a good alternative that efficiently computes dense CNN features at any receptive field sizes without loss of resolution. This is achieved by a specially designed 'hole' kernel which has sparsely aligned weights.

Additionally, multi-scale information often plays an important role for many computer vision tasks, such as image classification [13] and semantic segmentation [53, 4]. Previous studies [18, 39] in cognitive psychology also emphasized the multi-scale nature as an essential element of visual saliency. Motivated by above research, we utilize a PDC module, which consists of a set of dilated convolutions with different dilation rates, for emphasizing multi-scale spatial saliency representation learning (see Fig. 2).

More specially, let $\mathbf{F} \in \mathbb{R}^{W \times H \times M}$ denote the input 3D feature tensor, a set of K dilated convolution layers with kernels $\{\mathbf{C}_k \in \mathbb{R}^{c \times c \times C}\}_{k=1}^{K}$ and different dilation factors $\{r_k\}_{k=1}^{K}$ (strides are set as 1) are adopted for generating a set of

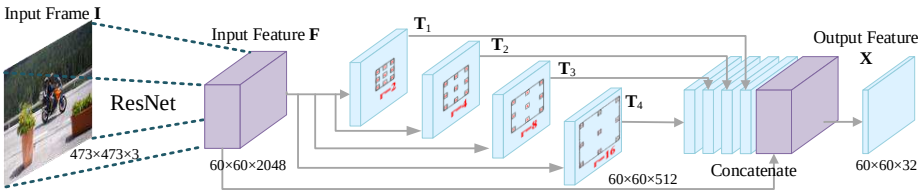

Fig. 2. Illustration of PDC module, where features from 4 parallel dilated convolution branches with different dilated rates are concatenated with the input features for emphasizing multi-scale spatial feature learning. See § 3.1 for details.

output feature maps $\{\mathbf{T}_k \in \mathbb{R}^{W\times H\times C}\}_{k=1}^{K}$:

$$\mathbf{T}_k = \mathbf{C}_k \circledast \mathbf{F}. \tag{1}$$

Here '$\circledast$' indicates the dilated convolution operation. For a certain dilated convolution layer with $c\times c$ kernel and r_k dilation rate, it could preserve a receptive field with size of $[(c-1)r_k+1]^2$. Thus the dilated convolution increases receptive field size exponentially while with linear parameter accretion. Although the sizes of the output features are identical, the receptive field sizes $[(c-1)r_k+1]^2$ differ significantly with the change of the dilation rate r_k, sometimes even being much larger than the input frame size. This is similar to observing the image from different distances. A region will be reasonably salient if only we see it from a proper distance and see its proper spatial context.

After that, multi-scale spatial features $\{\mathbf{T}_k\}_{k=1}^{K}$ are concatenated together and fed into PDB-ConvLSTM (detailed in next section), thus the network is able to learn the importance of the scales automatically (such as learning saliency feature from a proper distance). The combined feature $\mathbf{X} \in \mathbb{R}^{W\times H\times KC}$ is calculated as:

$$\mathbf{X} = [\mathbf{T}_1,\ \mathbf{T}_2, \ldots,\ \mathbf{T}_K], \tag{2}$$

where '$[.,.]$' represents the concatenation operation.

Inspired by the residual connection [14], we further combine the original input feature $\mathbf{F}$ into $\mathbf{X}$ to address the degradation problem. Thus $\mathbf{X} \in \mathbb{R}^{W\times H\times(KC+M)}$ in above equation is improved in a residual form:

$$\mathbf{X} = [\mathbf{F},\ \mathbf{T}_1,\ \mathbf{T}_2, \ldots,\ \mathbf{T}_K]. \tag{3}$$

In comparison, [4] proposes an *Atrous Spatial Pyramid Pooling* (ASPP) module, which applies multiple parallel atrous (dilated) convolutions with different sampling rates. Our PDC module has similar structure. However, ASPP simply performs element-wise sum operation (denoted by '$\oplus$') on the output features from dilated convolution layers: $\mathbf{X} = \mathbf{T}_1 \oplus \mathbf{T}_2 \oplus \ldots \oplus \mathbf{T}_K$, treating the features from different scales equally. Differently, PDC lets the network automatically learn the weights of different features. Our design is more intuitive and effective, which will be further quantitatively verified in § 4.4. With above definition, we build a powerful spatial feature learning model that emphasizes multi-scales. In

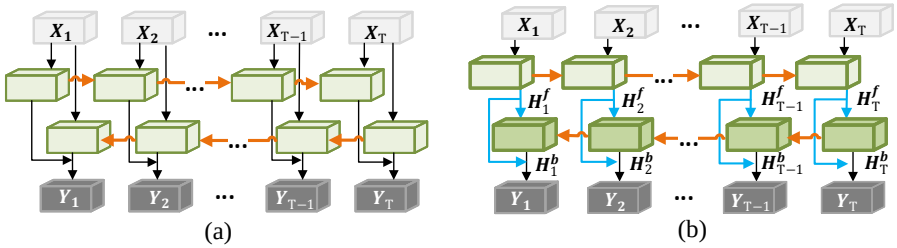

Fig. 3. Illustration of (a) Bidirectional ConvLSTM and (b) the proposed DB-ConvLSTM module. In PDB-ConvLSTM module, two DB-ConvLSTMs with different dilate rates are adopted for capturing multi-scale information and encouraging information flow between bi-directional LSTM units. See § 3.2 for details.

next section, we will improve traditional ConvLSTM with deeper spatiotemporal information extraction and a PDC-like structure.

3.2 Spatiotemporal Saliency Learning via PDB-ConvLSTM Module

Given an input video sequence $\{\mathbf{I}_t\}_{t=1}^T$ with T frames, we adopt PDC Module to produce a corresponding sequence of multi-scale spatial saliency features $\{\mathbf{X}_t\}_{t=1}^T$. Then these spatial features are fed into a modified ConvLSTM structure, called Pyramid Dilated Bidirectional ConvLSTM (PDB-ConvLSTM), for interpreting the temporal characteristics of video frames and fusing spatial and temporal features automatically. The PDB-ConvLSTM is improved in two ways. First, previous shallow, parallel bi-directional feature extraction strategy is replaced with a deeper and cascaded learning process, *i.e.*, building backward LSTM unit upon spatiotemporal features learnt in forward process. Second, incorporating pyramid dilated convolutions into LSTM for learning saliency features in multi-scales. Before detailing the proposed PDB-ConvLSTM module, we first give a brief introduction of classic ConvLSTM.

Vanilla ConvLSTM. ConvLSTM [32], as a convolutional counterpart of conventional fully connected LSTM (FC-LSTM) [15], introduces convolution operation into input-to-state and state-to-state transitions. ConvLSTM preserves spatial information as well as modeling temporal dependency. Thus it has been well applied in many spatiotemporal pixel-level tasks, such as dynamic visual attention prediction [41], video super-resolution [12]. Similar to FC-LSTM, ConvLSTM unit consists of a memory cell $\mathbf{c}_t$, an input gate $\mathbf{i}_t$, an output gate $\mathbf{o}_t$ and a forget gate $\mathbf{f}_t$. The memory cell $\mathbf{c}_t$, acting as an accumulator of the state information, is accessed, updated and cleared by self-parameterized controlling gates: $\mathbf{i}_t$, $\mathbf{o}_t$ and $\mathbf{f}_t$. As soon as an input arrives, the new data will be accumulated to the memory cell if the input gate is activated. Similarly, the past cell status $\mathbf{c}_{t-1}$ could be forgotten if the forget gate $\mathbf{f}_t$ is switched on. Whether the latest memory cell's value $\mathbf{c}_t$ will be transmitted to the final state $\mathbf{h}_t$ is further controlled by the output gate $\mathbf{o}_t$. With above definitions, ConvLSTM can be

formulated as follows:

$$\begin{aligned}
\mathbf{i}_t &= \sigma(\mathbf{W}_i^X * \mathbf{X}_t + \mathbf{W}_i^H * \mathbf{H}_{t-1}),\\
\mathbf{f}_t &= \sigma(\mathbf{W}_f^X * \mathbf{X}_t + \mathbf{W}_f^H * \mathbf{H}_{t-1}),\\
\mathbf{o}_t &= \sigma(\mathbf{W}_o^X * \mathbf{X}_t + \mathbf{W}_o^H * \mathbf{H}_{t-1}),\\
\mathbf{c}_t &= \mathbf{f}_t \circ \mathbf{c}_{t-1} + \mathbf{i}_t \circ \tanh(\mathbf{W}_c^X * \mathbf{X}_t + \mathbf{W}_c^H * \mathbf{H}_{t-1}),\\
\mathbf{H}_t &= \mathbf{o}_t \circ \tanh(\mathbf{c}_t),
\end{aligned} \tag{4}$$

where '$*$' denotes the convolution operator and '$\circ$' denotes the Hadamard product. For simplicity, bias terms are omitted. All the gates $\mathbf{i}, \mathbf{f}, \mathbf{o}$, memory cell $\mathbf{c}$, hidden state $\mathbf{H}$ and the learnable weights $\mathbf{W}$ are 3D tensors.

It can be seen that above ConvLSTM simply 'remembers' the past sequences, since it accumulates the past information in the memory cells. However, in video sequences, information from both the forward and backward frames are important and complementary for predicting video saliency. Thus Bidirectional ConvLSTM (B-ConvLSTM) should be used for capturing temporal characteristics in bi-directions (see Fig. 3 (a)):

$$\mathbf{Y}_t = \tanh(\mathbf{W}_y^{H^f} * \mathbf{H}_t^f + \mathbf{W}_y^{H^b} * \mathbf{H}_{t-1}^b), \tag{5}$$

where $\mathbf{H}^f$ and $\mathbf{H}^b$ indicates the hidden states from forward and backward ConvLSTM units, and $\mathbf{Y}_t$ indicates the final output considering bidirectional spatiotemporal information.

Deeper Bidirectional ConvLSTM. In B-ConvLSTM, there is no information exchange between the forward and backward directional LSTM units. We first improve B-ConvLSTM by organizing the forward and backward ConvLSTM units in a cascaded and tighter way, called Deeper Bidirectional ConvLSTM (DB-ConvLSTM). The DB-ConvLSTM has two layers, a shallow, forward layer and a deeper, backward layer (see Fig. 3 (b)). The ConvLSTM units in the forward layer receive spatial feature maps $\{\mathbf{X}_t\}_{t=1}^T$ from T frames as inputs, and output forward sequential feature maps $\{\mathbf{H}_t^f\}_{t=1}^T$ (according to Eq. 4). The deeper layer is constituted of the backward units that receive the output features from the forward layer $\{\mathbf{H}_t^f\}_{t=1}^T$ as inputs. Formally, the backward ConvLSTM unit is formulated as:

$$\begin{aligned}
\mathbf{i}_t^b &= \sigma(\mathbf{W}_i^{H^f} * \mathbf{H}_t^f + \mathbf{W}_i^{H^b} * \mathbf{H}_{t+1}^b),\\
\mathbf{f}_t^b &= \sigma(\mathbf{W}_f^{H^f} * \mathbf{H}_t^f + \mathbf{W}_f^{H^b} * \mathbf{H}_{t+1}^b),\\
\mathbf{o}_t^b &= \sigma(\mathbf{W}_o^{H^f} * \mathbf{H}_t^f + \mathbf{W}_o^{H^b} * \mathbf{H}_{t+1}^b),\\
\mathbf{c}_t^b &= \mathbf{f}_t^b \circ \mathbf{c}_{t+1}^b + \mathbf{i}_t^b \circ \tanh(\mathbf{W}_c^{H^f} * \mathbf{H}_t^f + \mathbf{W}_c^{H^b} * \mathbf{H}_{t+1}^b),\\
\mathbf{H}_t^b &= \mathbf{o}_t^\mathbf{b} \circ \tanh(\mathbf{c}_t^\mathbf{b}).
\end{aligned} \tag{6}$$

Then the forward features $\{\mathbf{H}_t^f\}_{t=1}^T$ and the backward features $\{\mathbf{H}_t^b\}_{t=1}^T$ are combined for final outputs: $\{\mathbf{Y}_t\}_{t=1}^T$, using Eq. 5. In this way, information are

encouraged to flow between the forward and backward ConvLSTM units, and deeper spatiotemporal features can be extracted by the backward units.

Deeper Bidirectional ConvLSTM with Pyramid Dilated Convolution. In order to extract more powerful spatiotemporal information and let the network adapt to salient targets at different scales, we further extend DB-ConvLSTM with a PDC-like structure. Specifically, the outputs $\{\mathbf{X}_t\}_{t=1}^{T}$ from the spatial PDC module are fed into several parallel DB-ConvLSTMs (see Fig. 1). In these DB-ConvLSTM modules, convolution operation '$*$' is further replaced by dilated convolution '$\circledast$' and different dilation factors are adopted. Such designs lead to a more powerful ConvLSTM structure, named Pyramid Dilated Bidirectional ConvLSTM (PDB-ConvLSTM). It is able to utilize different features from different receptive fields for capturing more complementary spatiotemporal features.

3.3 Detailed Network Architecture

Base Network. At the bottom the suggested model resides a stack of convolutional layers, which are borrowed from the first five convolution blocks of the model in [53] (a ResNet-50 [14]-like model). Given an input frame $\mathbf{I}$ with resolution of 473×473, the feature $\mathbf{F} \in \mathbb{R}^{60\times60\times2048}$ extracted from the last convolution block is fed into our PDC module for multi-scale spatial feature learning.

PDC Module. In PDC module, four parallel dilated convolution layers ($K = 4$) are adopted, where the size of the kernel is set as $c = 3, C = 512$, and the four dilation factors are set as $r_k = 2^k$ ($k = \{1, \ldots, 4\}$). Thus the PDC module is able to extract features on four different scales. Following Eq. 3, the outputs $\{\mathbf{T}_k \in \mathbb{R}^{60\times60\times512}\}_{k=1}^{4}$ of the four dilated convolution branches and the inputs $\mathbf{F} \in \mathbb{R}^{60\times60\times2048}$ of the PDC module are further concatenated for generating a multi-scale spatial saliency feature $\mathbf{X} \in \mathbb{R}^{60\times60\times4096}$. A 1×1 convolution layer with 32 channels is then applied for feature dimension reduction. Therefore, for the input video $\{\mathbf{I}_t\}_{t=1}^{T}$, PDC module produces a sequence of multi-scale spatial features $\{\mathbf{X}_t \in \mathbb{R}^{60\times60\times32}\}_{t=1}^{T}$, which will be further fed into the PDB-ConvLSTM module for spatiotemporal saliency prediction.

PDB-ConvLSTM Module. PDB-ConvLSTM module consists of two DB-ConvLSTMs, which are equipped with 3×3 kernels (32 channels). The dilation factors are set as 1 and 2 respectively, due to our limited computational resources. Note that, when the dilation factor is set as 1, the dilation kernel can be viewed as a normal convolution kernel without any 'holes'. For each frame, the outputs of the two DB-ConvLSTM branches in the PDB-ConvLSTM module are further concatenated as a multi-scale spatiotemporal saliency feature with $60 \times 60 \times 64$ dimensions. Then the output features from the PDB-ConvLSTM module are fed into a 1×1 convolution layer with 1 channel and *sigmoid* activation for producing final saliency map. The saliency map is upsampled into the original input frame size, *i.e.*, 473×473, via bilinear interpolation.

Loss Function. For producing better saliency prediction and training the suggested model more efficiently, we propose here a fused loss function that accounts for multiple evaluation metrics, inspired by [40]. Let $\mathbf{G} \in \{0, 1\}^{473\times473}$

and $\mathbf{S} \in [0,1]^{473\times473}$ denote the groundtruth saliency map and predicted saliency respectively, the overall loss $\mathcal{L}$ can be formulated as follows:

$$\mathcal{L}(\mathbf{S},\mathbf{G}) = \mathcal{L}_{cross\_entropy}(\mathbf{S},\mathbf{G}) + \mathcal{L}_{MAE}(\mathbf{S},\mathbf{G}), \tag{7}$$

where $\mathcal{L}_{cross\_entropy}$ and $\mathcal{L}_{MAE}$ indicate cross entropy loss and MAE loss respectively. $\mathcal{L}_{cross\_entropy}$ is computed as:

$$\mathcal{L}_{cross\_entropy}(\mathbf{S},\mathbf{G}) = -\frac{1}{N}\sum\nolimits_{i=1}^{N}[g_i log(s_i) + (1-g_i) log(1-s_i)] \tag{8}$$

where $g_i \in \mathbf{G}$, and $s_i \in \mathbf{S}$. N indicates the total pixel number, *i.e.*, $N = 473\times473$.

$\mathcal{L}_{MAE}$ is based on MAE metric, which is widely used in salient object detection. $\mathcal{L}_{MAE}$ computes the absolute difference between the predicted saliency map $\mathbf{S}$ and the corresponding ground truth $\mathbf{G}$:

$$\mathcal{L}_{MAE}(\mathbf{S},\mathbf{G}) = \frac{1}{N}\sum\nolimits_{i=1}^{N}|g_i - s_i|. \tag{9}$$

Training Settings. During training, with training batch size H, the network can be fed with H video frames or H copies of the same image. The latter one can be interpreted as using the PDB-ConvLSTM to refine the single-image saliency map for $2H$ times in each DB-ConvLSTM branch. This means we can utilize the massive static-image saliency data to let the network capture more appearances of the objects and the scenes. Therefore, our training procedure has three steps. First, we pre-train the spatial-learning part (including PDC module and base network) using two image saliency datasets: MSRA10K [6], and DUT-OMRON [49], and one video dataset: the training set of DAVIS dataset [31]. Initial learning rate of SGD algorithm is 10^{-8}. Then we set the learning rate of spatiotemporal learning part (PDB-ConvLSTM module) as 10^{-6}, and use above static and video data to train the whole model. After that, we fix the weights of the spatial-learning part and fine-tune the spatiotemporal learning part with the training set of DAVIS dataset only (learning rate is set as 10^{-6}). In this way, although the densely labelled video saliency data is scarce, our video saliency detection model still achieves good generalization performance for unseen videos. Quantitative experiments regarding to our training strategy can be found in § 4.4. The proposed videos saliency model is implemented using PYTHON, with the Caffe toolbox. Momentum and weight decay are set to 0.9 and 0.0005 respectively. The length of the training video frames H is set to 5.

Data Augmentation. We use data augmentation by mirror reflection, rotation (four rotation angles, 0°, 90°, 180°, and 270°) and image cropping to relieve overfitting. For each training sample instance (static images or video frames), we crop out the most top, bottom, left, and right slices of the border. Meanwhile, considering the video sequences may have different frame rates, we utilize different sampling steps in time axis to increase training samples. Specifically, for each video training iteration, we first randomly select a video frame from video data as the first frame in the training batch. Then we pick up following frames with a certain sampling step (={1,2,3,4,5,6}) until obtaining a training batch of 5 frames. The total training time is about 40 hours on one GTX 1080Ti GPU (11G Memory).

4 Experiments

In this section, two sets of experiments are first performed. One is for examining the performance of the proposed model for the main purpose, video salient object detection (§ 4.1). The other one is for evaluating the effectiveness of the proposed model on unsupervised video object segmentation, as salient object detection has been shown as an essential preprocessing step for unsupervised segmentation task (§ 4.2). After that, in § 4.3, we present runtime analysis. Finally, an ablation study is performed to gain a deeper insight into the proposed model (§ 4.4).

For video salient object detection, we evaluate the performance on three public datasets, *i.e.*, Densely Annotated VIdeo Segmentation (DAVIS) [31], Freiburg-Berkeley Motion Segmentation (FBMS) [2] and the Video Salient object detection (ViSal) [43]. **DAVIS** consists of 50 high-quality videos, totaling 3455 frames with fully annotated pixel-level ground truths. We utilize its training set that consists of 30 videos, totaling 2079 frames, to train our model. We test the models on the test set, which contains 20 videos, totaling 1376 frames. **FBMS** contains 59 natural video sequences, in which 29 are for training and 30 are for testing. We report the performance of our method on the test set. **ViSal** is the first dataset specially designed for video salient object detection and includes 17 challenging video clips. The length of videos in ViSal ranges from 30 to 100 frames, and totally 193 frames are manually annotated. The whole ViSal dataset is used for evaluation. For unsupervised video object segmentation task, we perform experiments on the test sets of DAVIS and FBMS datasets, which are the most popular benchmarks currently.

4.1 Performance on Video Salient Object Detection

We compared our model with 18 famous saliency methods, including 11 image salient object detection models: Amulet [51], SRM [36], UCF [52], DSS [16], MSR [23], NLDF [29], DCL [25], DHS [26], ELD [22], RFCN [35], KSR [37]; and

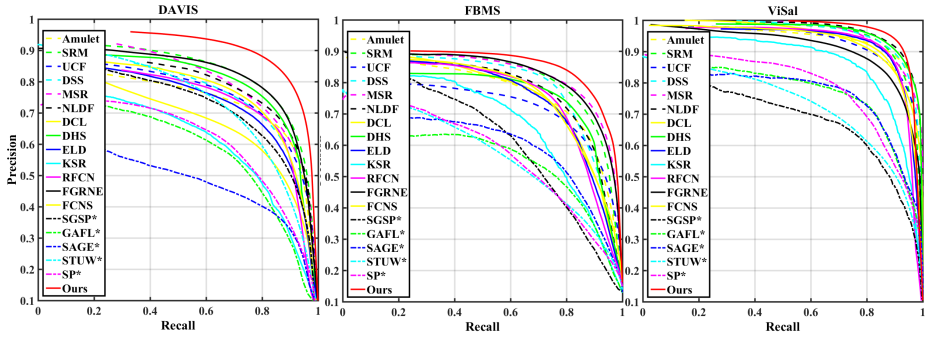

Fig. 4. Quantitative comparison against 18 saliency methods using PR curve on DAVIS [31], FBMS [2] and ViSal [43] datasets. Please see § 4.1 for more details.

Table 1. Quantitative comparison results against 18 saliency methods using MAE and maximum F-measure on DAVIS [31], FBMS [2] and ViSal [43]. The best scores are marked in **bold**. See § 4.1 for more details.

			DAVIS		FBMS		ViSal	
	Methods	Year	MAE↓	F^{max}↑	MAE↓	F^{max}↑	MAE↓	F^{max}↑
Image Saliency Models	Amulet [51]	ICCV'17	0.082	0.699	0.110	0.725	0.032	0.894
	SRM [36]	ICCV'17	0.039	0.779	0.071	0.776	0.028	0.890
	UCF [52]	ICCV'17	0.107	0.716	0.147	0.679	0.068	0.870
	DSS [16]	CVPR'17	0.062	0.717	0.083	0.764	0.028	0.906
	MSR [23]	CVPR'17	0.057	0.746	**0.064**	0.787	0.031	0.901
	NLDF [29]	CVPR'17	0.056	0.723	0.092	0.736	0.023	0.916
	DCL [25]	CVPR'16	0.070	0.631	0.089	0.726	0.035	0.869
	DHS [26]	CVPR'16	0.039	0.758	0.083	0.743	0.025	0.911
	ELD [22]	CVPR'16	0.070	0.688	0.103	0.719	0.038	0.890
	KSR [37]	ECCV'16	0.077	0.601	0.101	0.649	0.063	0.826
	RFCN [35]	ECCV'16	0.065	0.710	0.105	0.736	0.043	0.888
Video Saliency Models	FGRNE [24]	CVPR'18	0.043	0.786	0.083	0.779	0.040	0.850
	FCNS [44]	TIP'18	0.053	0.729	0.100	0.735	0.041	0.877
	SGSP* [27]	TCSVT'17	0.128	0.677	0.171	0.571	0.172	0.648
	GAFL* [43]	TIP'15	0.091	0.578	0.150	0.551	0.099	0.726
	SAGE* [42]	CVPR'15	0.105	0.479	0.142	0.581	0.096	0.734
	STUW* [8]	TIP'14	0.098	0.692	0.143	0.528	0.132	0.671
	SP* [28]	TCSVT'14	0.130	0.601	0.161	0.538	0.126	0.731
	Ours	ECCV'18	**0.030**	**0.849**	0.069	**0.815**	**0.022**	**0.917**

* Non-deep learning model.

7 video salient object detection approaches: SGSP [27], GAFL [43], SAGE [42], STUW [8], SP [28], FCNS [44], and FGRNE [24]. Note that FCNS and FGRNE are deep learning based video salient object detection models.

For quantitative evaluation, we employ three widely used metrics, namely PR curve, F-measure and MAE score. We refer readers to [44] for more details. Fig. 4 plots the PR curves on the test sets of DAVIS and FBMS datasets, as well as the whole ViSal dataset. It can be observed that our model outperforms other competitors. The maximum F-measures and MAE scores on above three datasets are reported in Table 1. Overall, our model achieves the best performance over three datasets using all the evaluation metrics. Fig. 5 presents some visual comparison results on three example video sequences: *horsejump-high* (from DAVIS), *tennis* (from FBMS), and *bird* (from ViSal). As seen, our model consistently produces accurate salient object estimations with various challenging scenes.

4.2 Performance on Unsupervised Video Object Segmentation

Video salient object detection model produces a sequence of probability maps that highlight the most visually important object(s). As demonstrated in [42], such salient object estimation could offer meaningful cue for unsupervised video primary object segmentation, which seeks to a binary foreground/background

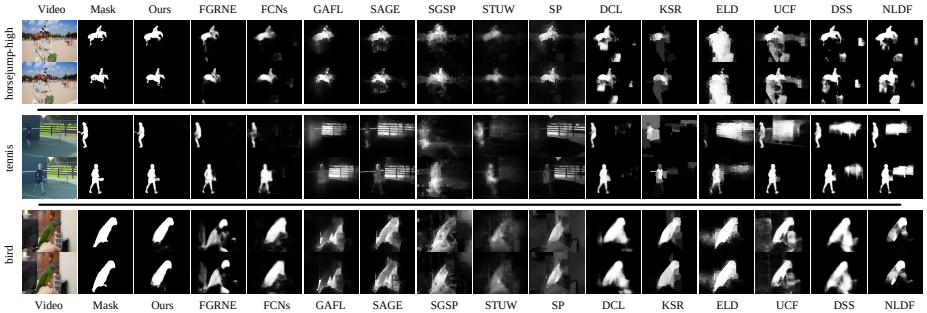

Fig. 5. Qualitative comparison against other top-performing saliency methods with groundtruths on three example video sequences. Zoom-in for details.

Table 2. Comparison with 7 representative unsupervised video object segmentation methods on the test sets of DAVIS and FBMS datasets. The best scores are marked in **bold**. See § 4.2 for details.

Dataset	Metric	Method								
		ARP*[20]	LVO[34]	FSEG[19]	LMP[33]	SFL*[5]	FST*[30]	SAGE*[42]	Ours	Ours+
DAVIS	$\mathcal{J}\uparrow$	76.2	75.9	70.7	70.0	67.4	55.8	41.5	74.3	**77.2**
	$\mathcal{F}\uparrow$	70.6	72.1	65.3	65.9	66.7	51.1	36.9	72.8	**74.5**
FBMS	$\mathcal{J}\uparrow$	59.8	65.1	68.4	35.7	55.0	47.7	61.2	72.3	**74.0**

* Non-deep learning model.

classification of each pixel. Thus video salient object detection can be used as a pre-processing step for unsupervised video segmentation. For better demonstrating the advantages of the proposed video saliency model, we extend our model for unsupervised video object segmentation and test it on DAVIS and FBMS datasets in segmentation settings.

Given an input frame $\mathbf{I}_t$ and corresponding saliency estimation $\mathbf{S}_t$, we formulate the segmentation task as an energy function minimization problem. The segmentation energy function is based on fully connected CRF model [21], where the foreground (or background) label assignment probability in binary term is $\mathbf{S}_t$ (or $1-\mathbf{S}_t$). The pairwise potential is defined as [21]. With the publicly available implementation of [21], our model takes about $0.5 \sim 1$ second per frame to generate a segmentation mask.

We compare our segmentation results with 7 representative unsupervised video segmentation methods [20, 34, 19, 33, 5, 30, 42], on the test sets of DAVIS and FBMS datasets. Following the experimental settings of DAVIS dataset, we employ the intersection-over-union metric ($\mathcal{J}$) and contour accuracy ($\mathcal{F}$) metrics for quantitative evaluation. For FBMS dataset, we adopted intersection-over-union score, as done by previous methods. Table 2 summarizes the results, where *Ours* indicates the results obtained via a simple thresholding strategy ($= 0.42$, validated on the training set of DAVIS) and *Ours+* indicates the results obtained via the CRF optimization. It can be observed that our model is able to produce more accurate segmentation results. This is mainly because the proposed PDB-

Table 3. Runtime comparison with 6 existing video saliency methods.

Method	SGSP[27]	SAGE[42]	GAFL[43]	STUW[8]	SP[28]	FCNS[44]	Ours
Time(s)	1.70*(+)	0.88*(+)	1.04*(+)	0.78*(+)	6.05*(+)	0.47	**0.05**

* CPU time.
(+) indicates extra computation of optical flow. For reference, LDOF [1] takes about 49.64s per frame, Flownet v2.0 [17] takes about 0.05s per frame.

ConvLSTM offers a powerful spatiotemporal learning framework that captures multi-scale features efficiently.

4.3 Runtime Analysis

In Table 3, we report the runtime comparison results with other 6 video saliency models, namely SGSP, SAGE, GAFL, STUW, SP, FCNS. For all the methods, we exclude their computation time of optical flow and the I/O time. All timings are measured on the same computer configuration Intel Core i7-6700 @3.4GHz and GTX 1080Ti GPU. SGSP, SAGE, STUW and SP are run on CPU and take optical flow as extra input. FCNS needs to calculate static saliency first. In comparison, our model extracts spatial features for each input frame independently, and leans temporal dynamics via the efficient PDB-ConvLSTM module without optical flow. Additionally, our model does not need any pre-/post-processing. For a 353×353 input frame, our model achieves the fastest speed of 20 fps.

4.4 Ablation Study

PDC Module. In order to analyze the effect of PDC module, we derive four variants, each of which only adopts one single dilate rate, *i.e.*, r is set to $2, 4, 8$, or 16. We also replace our PDC module with ASPP [4], which adopts element-wise sum operation, instead of concatenation, over all the features from different scales. The experimental results are summarized in Table 4. We can observe performance drops when only considering single scale and features extracted from different scales would have different impacts on the final results. Confusing multi-scale features (baseline: r=$\{2, 4, 8, 16\}$) brings the best performance. The results also demonstrate that the proposed PDC module is more favored, compared with ASPP, since PDC module lets the network automatically learn the importance of each scale via concatenation operation.

Table 4. Ablation study for PDC module on DAVIS and FBMS datasets.

Dataset	Metric	PDC Module					ASPP [4]
		$r=2$	$r=4$	$r=8$	$r=16$	$r=\{2,4,8,16\}$	
DAVIS	$F^{max}\uparrow$	0.703	0.704	0.715	0.708	**0.774**	0.769
	MAE $\downarrow$	0.079	0.077	0.074	0.074	0.047	**0.045**
FBMS	$F^{max}\uparrow$	0.707	0.702	0.714	0.716	**0.744**	0.730
	MAE $\downarrow$	0.110	0.109	0.107	0.108	**0.103**	0.111

Table 5. Ablation study for PDB-ConvLSTM on DAVIS and FBMS datasets.

Dataset	Metric	FC-LSTM	ConvLSTM	B-ConvLSTM	DB-ConvLSTM	PDB-ConvLSTM
DAVIS	F^{max} ↑	0.705	0.783	0.786	0.809	**0.849**
	MAE ↓	0.056	0.043	0.039	0.036	**0.030**
FBMS	F^{max} ↑	0.672	0.755	0.757	0.799	**0.815**
	MAE ↓	0.121	0.096	0.094	0.072	**0.069**

PDB-ConvLSTM module. Four baselines are used to discuss the contribution of our PDB-ConvLSTM module in spatiotemporal information learning, namely Fully Connected LSTM (FC-LSTM), Convolutional LSTM (ConvLSTM), Bidirectional ConvLSTM (B-ConvLSTM), and Deeper Bidirectional ConvLSTM (DB-ConvLSTM). In comparison, we replace our PDB-ConvLSTM module with above variants and report the corresponding performance using maximum F-score and MAE over the DAVIS and FBMS datasets. The results are summarized in Table 5. It can be observed that, no surprisingly, FC-LSTM gains worst performance since it totally loses spatial details. DB-LSTM performs better than ConvLSTM and B-ConvLSTM due to its deeper fusion of bidirectional information. PDB-ConvLSTM further advances the performance by considering multi-scales.

Training Protocol. Now we assess the our training strategy, *i.e.*, using massive data from static images and video frames. On DAVIS dataset, we find *Our w. static data* (F^{max} ↑: 0.849, MAE ↓: 0.030) outperforms *Our w/o. static data* (F^{max} ↑: 0.753, MAE ↓: 0.049). This demonstrates that using static data to train the model can avoid the risk of over-fitting on relatively small amount of video data and improve the generalization ability of our model.

5 Conclusions

This paper proposed a deep video salient object detection model which consists of two essential components: PDC module and PDB-ConvLSTM module. In the PDC module, a set of parallel dilated convolutions are adopted for extracting multi-scale spatial features through different receptive fields. In the PDB-ConvLSTM module, conventional ConvLSTM is extended with deeper information extraction and parallel two dilated ConvLSTMs to extract sequential features at different scales. The proposed model leverages both labeled video data and also the massive amount of labeled static-image data for training, so as to increase its generalization to diverse videos. The proposed model generates high-quality saliency maps with a real-time processing speed of 20 fps. The experiments also demonstrate the proposed model is well applicable to unsupervised segmentation task and achieves most accurate segmentation results.

Acknowledgments. This research was supported in part by the Beijing Natural Science Foundation under Grant 4182056, the Fok Ying Tung Education Foundation under Grant 141067, and the Specialized Fund for Joint Building Program of Beijing Municipal Education Commission.

References

1. Brox, T., Malik, J.: Large displacement optical flow: Descriptor matching in variational motion estimation. IEEE TPAMI **33**(3), 500–513 (2011)
2. Brox, T., Malik, J.: Object segmentation by long term analysis of point trajectories. In: ECCV. pp. 282–295 (2010)
3. Chang, J., Wei, D., Fisher, J.W.: A video representation using temporal superpixels. In: IEEE CVPR. pp. 2051–2058 (2013)
4. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. IEEE TPAMI **40**(4), 834–848 (2018)
5. Cheng, J., Tsai, Y.H., Wang, S., Yang, M.H.: SegFlow: Joint learning for video object segmentation and optical flow. In: IEEE CVPR. pp. 686–695 (2017)
6. Cheng, M.M., Mitra, N.J., Huang, X., Torr, P.H.S., Hu, S.M.: Global contrast based salient region detection. IEEE TPAMI **37**(3), 569–582 (2015)
7. Cong, R., Lei, J., Fu, H., Huang, Q., Cao, X., Hou, C.: Co-saliency detection for rgbd images based on multi-constraint feature matching and cross label propagation. IEEE TIP **27**(2), 568–579 (2018)
8. Fang, Y., Wang, Z., Lin, W., Fang, Z.: Video saliency incorporating spatiotemporal cues and uncertainty weighting. IEEE TIP **23**(9), 3910–3921 (2014)
9. Fu, H., Cao, X., Tu, Z.: Cluster-based co-saliency detection. IEEE TIP **22**(10), 3766–3778 (2013)
10. Fu, H., Xu, D., Zhang, B., Lin, S., Ward, R.K.: Object-based multiple foreground video co-segmentation via multi-state selection graph. IEEE TIP **24**(11), 3415–3424 (2015)
11. Guo, F., Wang, W., Shen, J., Shao, L., Yang, J., Tao, D., Tang, Y.Y.: Video saliency detection using object proposals. IEEE TCYB (2018)
12. Guo, J., Chao, H.: Building an end-to-end spatial-temporal convolutional network for video super-resolutiong. In: AAAI. pp. 4053–4060 (2017)
13. He, K., Zhang, X., Ren, S., Sun, J.: Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE TPAMI **37**(9), 1904–1916 (2015)
14. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE CVPR. pp. 770–778 (2016)
15. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Computation **9**(8), 1735–1780 (1997)
16. Hou, Q., Cheng, M.M., Hu, X., Borji, A., Tu, Z., Torr, P.H.S.: Deeply supervised salient object detection with short connections. In: IEEE CVPR. pp. 5300–5309 (2017)
17. Ilg, E., Mayer, N., Saikia, T., Keuper, M., Dosovitskiy, A., Brox, T.: Flownet 2.0: Evolution of optical flow estimation with deep networks. In: IEEE CVPR. pp. 2462–2470 (2017)
18. Itti, L., Koch, C.: Computational modelling of visual attention. Nature Reviews Neuroscience **2**(3), 194–203 (2001)
19. Jain, S., Xiong, B., Grauman, K.: Fusionseg: Learning to combine motion and appearance for fully automatic segmention of generic objects in videos. In: IEEE CVPR (2017)
20. Koh, Y.J., Kim, C.S.: Primary object segmentation in videos based on region augmentation and reduction. In: IEEE CVPR. pp. 7417–7425 (2017)
21. Krahenbuhl, P., Koltun, V.: Efficient inference in fully connected CRFs with Gaussian edge potentials. In: NIPS (2011)

22. Lee, G., Tai, Y.W., Kim, J.: Deep saliency with encoded low level distance map and high level features. In: IEEE CVPR. pp. 660–668 (2016)
23. Li, G., Xie, Y., Lin, L., Yu, Y.: Instance-level salient object segmentation. In: IEEE CVPR. pp. 247–256 (2017)
24. Li, G., Xie, Y., Wei, T., Wang, K., Lin, L.: Flow guided recurrent neural encoder for video salient object detection. In: IEEE CVPR. pp. 3243–3252 (2018)
25. Li, G., Yu, Y.: Deep contrast learning for salient object detection. In: IEEE CVPR. pp. 478–487 (2016)
26. Liu, N., Han, J.: Dhsnet: Deep hierarchical saliency network for salient object detection. In: IEEE CVPR. pp. 678–686 (2016)
27. Liu, Z., Li, J., Ye, L., Sun, G., Shen, L.: Saliency detection for unconstrained videos using superpixel-level graph and spatiotemporal propagation. IEEE TCSVT **27**(12), 2527–2542 (2017)
28. Liu, Z., Zhang, X., Luo, S., Meur, O.L.: Superpixel-based spatiotemporal saliency detection. IEEE TCSVT **24**(9), 1522–1540 (2014)
29. Luo, Z., Mishra, A.K., Achkar, A., Eichel, J.A., Li, S., Jodoin, P.M.: Non-local deep features for salient object detection. In: IEEE CVPR. pp. 6593–6601 (2017)
30. Papazoglou, A., Ferrari, V.: Fast object segmentation in unconstrained video. In: IEEE ICCV. pp. 1777–1784 (2013)
31. Perazzi, F., Pont-Tuset, J., McWilliams, B., Gool, L.J.V., Gross, M.H., Sorkine-Hornung, A.: A benchmark dataset and evaluation methodology for video object segmentation. In: IEEE CVPR. pp. 724–732 (2016)
32. Shi, X., Chen, Z., Wang, H., Yeung, D.Y., Wong, W.K., Woo, W.C.: Convolutional LSTM network: A machine learning approach for precipitation nowcasting. In: NIPS (2015)
33. Tokmakov, P., Alahari, K., Schmid, C.: Learning motion patterns in videos. In: IEEE CVPR. pp. 531–539 (2017)
34. Tokmakov, P., Alahari, K., Schmid, C.: Learning video object segmentation with visual memory. In: IEEE ICCV (2017)
35. Wang, L., Wang, L., Lu, H., Zhang, P., Xiang, R.: Saliency detection with recurrent fully convolutional networks. In: ECCV. pp. 825–841 (2016)
36. Wang, T., Borji, A., Zhang, L., Zhang, P., Lu, H.: A stagewise refinement model for detecting salient objects in images. In: IEEE ICCV. pp. 4039–4048 (2017)
37. Wang, T., Zhang, L., Lu, H., Sun, C., Qi, J.: Kernelized subspace ranking for saliency detection. In: ECCV. pp. 450–466 (2016)
38. Wang, W., Shen, J., Li, X., Porikli, F.: Robust video object cosegmentation. IEEE TIP **24**(10), 3137–3148 (2015)
39. Wang, W., Shen, J.: Deep visual attention prediction. IEEE TIP **27**(5), 2368–2378 (2018)
40. Wang, W., Shen, J., Dong, X., Borji, A.: Salient object detection driven by fixation prediction. In: IEEE CVPR. pp. 1171—1720 (2018)
41. Wang, W., Shen, J., Guo, F., Cheng, M.M., Borji, A.: Revisiting video saliency: A large-scale benchmark and a new model. In: IEEE CVPR. pp. 4894–4903 (2018)
42. Wang, W., Shen, J., Porikli, F.: Saliency-aware geodesic video object segmentation. In: IEEE CVPR. pp. 3395–3402 (2015)
43. Wang, W., Shen, J., Shao, L.: Consistent video saliency using local gradient flow optimization and global refinement. IEEE TIP **24**(11), 4185–4196 (2015)
44. Wang, W., Shen, J., Shao, L.: Video salient object detection via fully convolutional networks. IEEE TIP **27**(1), 38–49 (2018)
45. Wang, W., Shen, J., Shao, L., Porikli, F.: Correspondence driven saliency transfer. IEEE TIP **25**(11), 5025–5034 (2016)

46. Wang, W., Shen, J., Xie, J., Fatih, P.: Super-trajectory for video segmentation. In: IEEE ICCV. pp. 1671–1679 (2017)
47. Wang, W., Shen, J., Yang, R., Porikli, F.: Saliency-aware video object segmentation. IEEE TPAMI **40**(1), 20–33 (2018)
48. Xu, C., Corso, J.J.: Evaluation of super-voxel methods for early video processing. In: IEEE CVPR. pp. 1202–1209 (2012)
49. Yang, C., Zhang, L., Lu, H., Ruan, X., Yang, M.H.: Saliency detection via graph-based manifold ranking. In: IEEE CVPR. pp. 3166–3173 (2013)
50. Yu, F., Koltun, V.: Multi-scale context aggregation by dilated convolutions. In: ICLR (2016)
51. Zhang, P., Wang, D., Lu, H., Wang, H., Ruan, X.: Amulet: Aggregating multi-level convolutional features for salient object detection. In: IEEE ICCV. pp. 202–211 (2017)
52. Zhang, P., Wang, D., Lu, H., Wang, H., Yin, B.: Learning uncertain convolutional features for accurate saliency detection. In: IEEE ICCV. pp. 212–221 (2017)
53. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: IEEE CVPR. pp. 6230–6239 (2017)