Novel Dilated Separable Convolution Networks for Efficient Video Salient Object Detection in the Wild

Hemraj Singh, Mridula Verma, *Member, IEEE*, and Ramalingaswamy Cheruku, *Member, IEEE*

***Abstract*— Appearance and motion are essential features in video salient object detection (VSOD) tasks. Most of the existing approaches use local features and thus fail to understand both the appearance and motion-specific semantics at the global level. Hence, these methods are unable to perform in unconstrained scenarios where multiple challenges, such as partial occlusion, motion blur, noise, and clutter background, exist. Moreover, these approaches require a large number of computational resources due to their complex structures, which limits their applicability to real-world deployment. To resolve these issues and to achieve a balance between accuracy and computational complexity, in this article, a dilation separable convolution network (DSCNet) is proposed, which is equipped with dilation attention fusion module (DAFM), bidirectional cross-modality fusion module (BCFM), and saliency prediction module (SPM) to extract enhanced multiscaled motion and appearance features without increasing the model complexity. Furthermore, a bidirectional separable convolution network (BSCNet) equipped with a separable convolution module (SCM) and a FlowNet2.0 is proposed to use multiscale contextual information across appearance cues and generate enhanced multiscaled motion maps. For faster and better training of the DSCNet model, we propose a novel stochastic-gradient-based firefly algorithm (SGFA), which adaptively balances the exploration and exploitation in multiscaled, cross-modal embedded subspaces. With the help of the proposed SGFA algorithm, the DSCNet+ model is constructed on top of DSCNet, which further improves the results in terms of the training speed and other evaluation metrics. The proposed models are evaluated on six benchmark datasets, and a detailed comparative study is provided with 16 state-of-the-art (SOTA) models. One of the major highlights of the work is the significant performance of the proposed models on the most difficult DAVSOD-Diff dataset, which best reflects the challenging real-world scenarios.**

***Index Terms*— Bidirectional cross-modality fusion, dilation attention fusion, dilation separable convolution network (DSCNet), firefly optimizer, multiscale cross-modal features, video salient object detection (VSOD).**

I. Introduction

VIDEO salient object detection (VSOD) aims to automatically discover salient objects and localize the visually attractive object regions from the video sequence that captures human attention. Real-world applications of VSOD include robotic manipulation [1], autonomous cars [2], medical image processing [3], traffic management [4], surveillance system [5], drones [6], smart home [7], and many more. Instruments and systems, which infer salient target objects in the videos, use various computer vision applications such as video thumbnailing [8], surface defect detection [9], and instant retrieval [10]. In practical scenarios, a number of challenges appear due to the unconstrained environment, such as low-light scenario, occlusion, motion blur, noise, cluttered background, and crowd (a few examples shown in Fig. 1). Therefore, robust algorithms are needed to detect objects efficiently in challenging real-world scenarios.

Traditional approaches make use of the simplex method using either appearance-based (e.g., color frame [12]) or motion-based (e.g., optical flow [5], [13] or pixel orientation [14]) features in consecutive frames [15], [16]. However, uncoordinated knowledge across these modalities usually leads to poor performance in unconstrained environments. For instance, one of the biggest disadvantages of considering the VSOD task as a moving object detection (MOD) task by just using motion-based features may lead to the possibility of missing targets when the object is static or moving very slowly. Sophisticated appearance-based approaches for VSOD algorithms, such as [9] and [17], generally provide a detailed target description. However, the results are unstable due to the absence of prior knowledge about primary objects.

Motion–appearance-based schemes [1], [12], [18] are able to handle these limitations and extract better semantic features using knowledge from both the modalities. Appearance features suffer from lack of motion descriptions, and thus motion cues assist in choosing the best candidate regions for appearance features. Detection performance has significantly improved with optical flow-driven VSOD methods [12], [15], [16], the leading motion–appearance scenario. However, it is difficult to handle static foreground objects, when appearance modeling is abandoned and VSOD is changed into a foreground motion prediction that is entirely dependent on optical flow information. If optical flow estimation fails, the motion features of the primary video objects are appropriately invalidated. In this scenario, detection accuracy is presumably harmed by the nonselective fusion of appearance and motion features.

Thus, it is essential to use a modality transmission scheme in place of embedding them separately one by one. Inspired by

Manuscript received 7 January 2023; revised 6 June 2023; accepted 19 July 2023. Date of publication 7 August 2023; date of current version 24 August 2023. The Associate Editor coordinating the review process was Dr. Salvatore Graziani. *(Corresponding author: Ramalingaswamy Cheruku.)*

Hemraj Singh and Ramalingaswamy Cheruku are with the Department of Computer Science and Engineering, National Institute of Technology, Warangal, Hanamkonda, Telangana 506004, India (e-mail: 720079@student.nitw.ac.in; rmlswamy@nitw.ac.in).

Mridula Verma is with the Institute for Development and Research in Banking Technology, Hyderabad, Telangana 500028, India (e-mail: vmridula@idrbt.ac.in).

Digital Object Identifier 10.1109/TIM.2023.3302911

Fig. 1. Challenging scenes from the DAVIS [11] and DAVSOD-Diff [1] datasets. (a) Low light. (b) Occlusion. (c) Motion blur. (d) Noise. (e) Clutter scene. (f) Motion crowd.

this, recently, bidirectional modality transmission schemes [1], [18], [19] from the field of computer networks have been use that explicitly associate the appearance and motion patterns across the feature extraction process. These strategies aggregate homogeneous spatiotemporal information to describe interactions between entangled spatial and temporal information and share the same underpinning correlative patterns of object perceptions, such as semantic structure, shape, and movement. The major challenge with such strategies is that they produce very large models. Their deployment on a resource-constrained environment, such as mobile and other edge devices, becomes difficult.

To handle all these challenges and achieve a balance among accuracy and the number of parameters, a dilation separable convolution network (DSCNet) is proposed, which is equipped with dilation attention fusion module (DAFM), effective bidirectional cross-modality fusion module (BCFM), and saliency prediction module (SPM) to extract efficient multiscale contextual information across appearance and motion cues for efficient salient object detection in videos. Our proposed framework is able to extract multiscale global context across embedding subspaces without increasing the number of parameters any further, and hence suitable for resource-constrained edge devices. Furthermore, a bidirectional separable convolution network (BSCNet) is proposed, which is equipped with a separable convolution module (SCM) and FlowNet2.0 to extract efficient multiscale motion features across the appearance cues. In addition, to improve the generalization performance of the optimizer, a new stochastic-gradient-based firefly algorithm (SGFA) is proposed, which adaptively balances the exploration and exploitation in multiscaled, cross-model embedded subspaces. The main contributions of our proposed models are given below.

1) A novel dilated separable convolution network (DSCNet) is proposed, which is equipped with 1) a DAFM, for extracting the attention-based multiscale motion and appearance features, with the help of atrous spatial pyramid pooling (ASPP) [20]; 2) BCFM, for extracting the fused cross-modal features; and 3) SPM, for efficient saliency map construction.
2) Furthermore, a novel BSCNet is proposed, which is equipped with an SCM and FlowNet2.0 [13] to extract lightweight multiscale contextual information across appearance cues and generate an enhanced multiscaled motion map.
3) For faster and better training, we propose a novel stochastic-gradient-descent-based firefly optimization technique, which improves the generalization performance of the optimizer.
4) With the help of extensive experimentation, we demonstrate that the proposed models perform superior on six benchmark datasets. DSCNet+ outplays the state-of-the-art (SOTA) unsupervised VSOD (UVSOD) model (i.e., MTG-Net [21]) on the DAVIS-diff dataset by 37.3% in terms of F-measure.

II. Related Work

This section discusses SOTA semi-supervised VSOD methods, unsupervised attention-based VSOD methods, and video object segmentation (VOS) methods.

A. Semi-Supervised VSOD

Recently, various VSOD tasks were addressed in a semi-supervised way [17], [22]. They use low-level handcrafted features for speculative detection inference such as saliency priors [23], object proposals [24], long sparse point orientation [25], optical flow [1], or super-pixels [22]. These conventional models have limited generalizable quality in dynamic and complex schemes due to the absence of semantic characteristics and high-level content understanding. Recently, RNN-based models [26] have come into existence as their more effective accomplishment of fascinating long-term dependencies of using deep learning. Semi-supervised VSOD formulates a recurrent model over time, exploiting spatial features combined with everlasting temporal context. Using motion features along with appearance features is a big challenge in these areas. Tokmakov et al. [17] proposed a motion-pattern-based model, which uses motion patterns from video. Yet, their model fails to segment objects into two adjacent frames since it correctly guides optical flow. Many works [24], [25] have been proposed to overcome these problems by fusing the spatiotemporal features with the help of parallel networks. Multistage processing methods are proposed in [27], which provide motion-based consistent features to detect the objects. A unified referring VOS network (URVOS) is presented in [28], which provides language expression of the entire video frame and detects the object.

B. Unsupervised Attention-Based VSOD

The unsupervised VSOD is related to the attention-based VSOD task, and its goal is to extract the attention-aware features from a video clip. Conventional models [29], [30] compute single-frame saliency using handcrafted static and motion features and performance spatiotemporal optimization for preserving consistency across successive frames. Furthermore, [21], [31] extract highly semantic spatiotemporal features to detect the object in an end-to-end fashion. Some deep learning models [32], [33], [34] proposed to extract motion features using optical flow or adjacent frames. A key-frame method is proposed in [35] to determine high-quality

video frames to categorize saliency objects from key-frame. In [36], a method to detect salient objects based on extracting spatial–temporal information from high-quality frames is proposed. To enhance the quality of temporal features, many researchers focused on various challenges such as limited labeled data, [21] or analyzing relative saliency [37] in VSOD. A shift-aware-ConvLSTM is proposed in [38] to extract features with high-quality annotations and an attention-consistent VSOD dataset. A dynamic context sensitive filtering network (DCFNet) is proposed [18], which uses an efficient bidirectional dynamic fusion technique to extract location-related similarities over consecutive frames. In [39], a stereoscopically attentive multiscale (SAM) module is proposed, which uses a stereoscopic attention mechanism to merge the characteristics of different scales adaptably. In similar lines, Gu et al. [24] proposed a constrained self-attention (CSA) module to extract motion features based on the prior movement of the objects.

C. Video Object Segmentation

VOS used video frames as inputs and extracts the spatio-temporal features without losing information in motion. Hu et al. [40] suggest a motion-guided cascaded refinement network for VOS to address this issue. Unsupervised video segmentation is crucial for numerous applications, such as object detection and compression. Fast motion, motion blur, and occlusions continue to be significant issues. A unique saliency estimation technique is designed in [41] to improve initial foreground–background estimations and address these problems across diffusion time in unsupervised video segmentation.

In [37], an interactively constrained encoding (ICE) module is designed to optimize the energy consumption of motion and appearance features in a graph. SegFlow is proposed by [42] for segmenting the object in both optical flow and pixel level in videos. A few-shot learning module is proposed in [7] to predict parametric features and reduce frame segmentation errors. Numerous CNN-based techniques [1], [10], [11] have been developed to segment an object; however, most of them are heavy models which consume a large amount of inference time. To solve these problems, [25] proposed a fast and accurate VOS algorithm, which uses the segmentation operation to handle problematic elements such as significant deformation, occlusion, and a cluttered background.

III. Proposed Method

A. Overview of Architecture

Consider a dataset having T video clips with k_t consecutive frames (where $t = 1, 2, \ldots, T$). We can define the appearance frames as $\{A_k^t\}_{t=1}^T$ and the corresponding annotation maps $\{G_k^t\}_{t=1}^T$. These frames are passed to the BSCNet, which is the combination of SCM and FlowNet2.0 [13], and T optical flow maps $M_k^t = f[A_k^t, A_k^{t+1}]$. Furthermore, the proposed DSCNet takes as inputs both the appearance frames $\{A_k^t\}_{t=1}^T$ and the corresponding motion frames $\{M_k^t\}_{t=1}^T$ and pass to the backbone ResNet-50 network to generate backbone features X_k^t and Y_k^t. Then, these backbone features X_k^t and Y_k^t passed to the DAFM to enhance the quality of the feature. The BCFM performed the cross-modality operation to extract the outer bound and inner bound region boundaries on filtering fused features P_k^t and motion features Q_k^t and generated the depthwise efficient features. Finally, the SPM is used to generate the saliency map $S_{A,M}^t$.

B. Bidirectional Separable Convolution Network

The BSCNet contains a stack of SCM, which can be used in a bidirectional way. BSCNet is used for fine-tuning the pretrained FlowNet2.0 model with ResNet-50 and extracting spatial–temporal features at multiple scales (refer Fig. 2). First, the two consecutive appearance frames (A_k^t, A_{k+1}^t) are passed to the ResNet-50 for generating the backbone features X_k^t and X_{k+1}^t. These backbone features are passed to the SCM. The SCM detects edges based on sparse matches and computes the geodesic distance to obtain dense matches of two neighboring frames. The dense matches are fused to generate multiscale enhanced spatial features. The SCM has four modules with the same configuration (a detailed explanation of SCM is given in Section III-B1. Furthermore, these features are passed to FlowNet2.0 to generate the optical flow motion maps.

1) Separable Convolution Module: As shown in Fig. 3, the configuration of SCM is given, which has channel attention (CA), dimension reduction (DR), bilinear interpolation (BI), and upsampling (UP) modules. The backbone appearance features are passed to the separable convolution (SConv) layer, which has $f \times f$ filters to extract the multiscale spatial features. The extracted multiscale spatial features are passed to the CA to extract channelwise features and enhance feature quality. The CA is configured with SConv with $c \times c$ filters, adaptive average pooling (AP), and Sigmoid operation. The SConv with $c \times c$ filter extracts multiscale spatial features, followed by AP, which provides the balance between input and output channel filters. Furthermore, the Sigmoid operation is used to normalize the features between 0 and 1. After that, DR is used to reduce the feature dimensions, which has an SConv layer with $f \times f$ filter. Then, BI performs a resampling operation to predict the pixel value using the distance-weighted average of four adjacent pixel values. Furthermore, the UP operation is performed to maintain the dimension of the feature vectors. Next, SCM computes deep spatial features without loss of resolution and generates a sequence of multiscale spatial features X_k^t and $X_{k+1}^t \in \mathbb{R}^{h \times w \times d}$ (where h is the height, w is the width, and d is the channel). Next, these features are passed to FlowNet2.0 to generate the motion map M_k^t.

C. Dilation Separable Convolutional Network (DSCNet)

The DSCNet architecture is shown in Fig. 4. It contains three modules: 1) DAFM, which is a combination of an SConv layer and ASPP module with different dilation rates; 2) BCFM, to extract and fuse two cross-modality features in a bidirectional way; and 3) SPM, to extract high-level strong features to low-level weak features. As shown in Fig. 4, the appearance A_k^t and motion M_k^t features are first passed to the backbone network ResNet-50, which generates the backbone appearance X_k^t and motion Y_k^t features. These

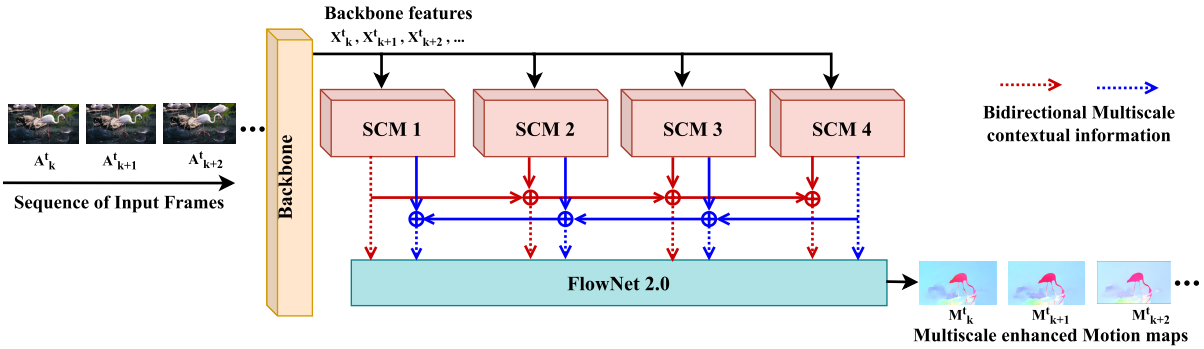

Fig. 2. Architecture diagram of BSCNet, which uses SCM and FlowNet2.0 to extract lightweight multiscale contextual information across appearance cues and generate enhanced multiscaled motion maps. ResNet-50 is used as the backbone.

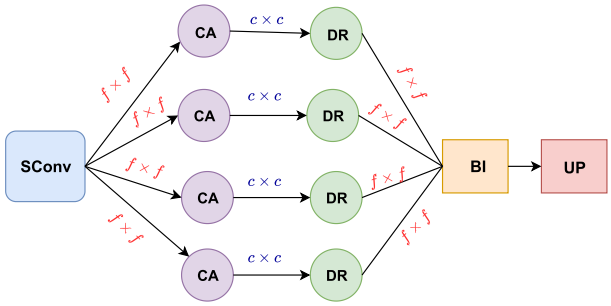

Fig. 3. Architecture diagram of SCM, which consists of CA, CA, BI, and UP.

backbone features are passed to DAFM (Section III-C1), which extracts spatio-temporal-based discriminative features and generates the fused features Z^t_k. These fused features are then passed to BCFM (Section III-C2) to extract spatial and temporal features and discriminate them into fused appearance P^t_k and motion features Q^t_k. Finally, with the help of the SPM (Section III-C4), the saliency map $S^t_{A,M}$ is generated.

1) Dilation Attention Fusion Module: The DAFM, shown in Fig. 5, is designed using a depthwise SConv (DSConv) layer combined with ASPP for extracting features at different dilation rates. Due to DSConv and ASPP, the computational complexity of the DAFM is significantly reduced in comparison to the existing aggregation methods [43], [44]. In addition, the aggregation can concentrate on the features of prominent objects or locations rather than the general feature maps with the help of our contextual attention.

DAFM uses backbone appearance X^t_k and motion features Y^t_k, which come from branches of ResNet-50. These features are passed to the depthwise SConv layer (DSConv) with $f \times f$ filter to extract the multiscale spatial and temporal features at multiple dilation rates. Then, the adaptive AP (AAP) operation is performed to generate the elementwise attention-based discriminative features vectors R^t_X and R^t_Y. Furthermore, the SConv operation is performed with $c \times c$ filter to generate spatial–temporal based discriminative global features using multiscale learnable parameter ω_{SConv}. Max-pooling and PReLU operations are performed to reduce the dimension of features and to handle the nonlinearity of the features, respectively. The depthwise spatial attention features Z^t_X and channelwise attention features Z^t_Y are obtain by the following operations:

$$Z^t_X = X^t_k \otimes \text{SConv}\left(R^t_X; \omega_{\text{SConv}}\right) \tag{1}$$

$$Z^t_Y = Y^t_k \otimes \text{SConv}\left(R^t_Y; \omega_{\text{SConv}}\right) \tag{2}$$

where $\otimes$ is an elementwise multiplication operation. Fused features Z^t_k are obtained by performing elementwise addition operation ($\oplus$) on Z^t_X and Z^t_Y followed by a Sigmoid operation (σ) as follows:

$$Z^t_k = \sigma\left(Z^t_X \oplus Z^t_Y\right). \tag{3}$$

2) Bidirectional Cross-Modality Fusion Module: The BCFM (shown in Fig. 6) aims to develop more precise representations of the motion and appearance features while reducing background noise and enhancing the saliency information. BCFM demonstrates that resampling features at several scales are useful for properly and quickly categorizing regions of arbitrary scales. Multiple scales of information are efficiently captured by BCFM using various dilation rates. However, we find that the number of valid filter weights (i.e., the weights that are applied to the valid feature region instead of padded zeros) is minimal as the sampling rate increases. Appearance and motion-level features are used to solve this issue and provide global context information to the model. According to Fig. 6, a DSConv with $f \times f$ filter and learnable parameter ω_{DSConv} is used to extract spatial and temporal features from Z^t_k and Y^t_k followed by batch normalization. The multiscale appearance and motion features are calculated using the following equation:

$$P^t_k = \text{DSConv}\left(Z^t_k; \omega_{\text{DSConv}}\right); \quad Q^t_k = \text{DSConv}\left(Y^t_k; \omega_{\text{DSConv}}\right). \tag{4}$$

Furthermore, a bilinear UP operation is performed on appearance features P^t_k and the motion features Q^t_k to persist the size of the features. An elementwise multiplication operation is performed between fused appearance and motion features to generate the cross-modality features. Next, a Sigmoid operation is performed for recentering and rescaling the features. Finally, global AP (GAP) with one convolution layer $c \times c$ filter is used to extract the spatio-temporal attention features and minimize overfitting by reducing the number of parameters. The BCFM suppresses robust features in a

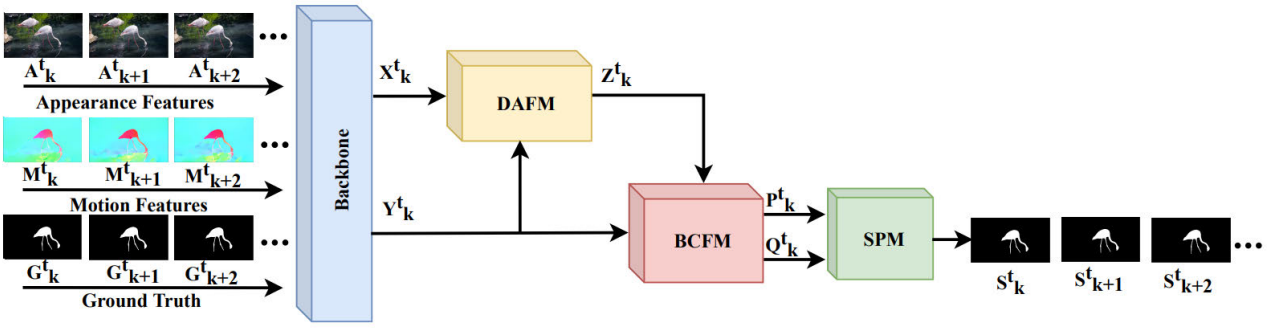

Fig. 4. Architecture of the proposed DSCNet uses a DAFM to extract the discriminative features from motion and appearance. Then, BCFM extracts cross-modality features and forwards them to the SPM to generate a saliency map.

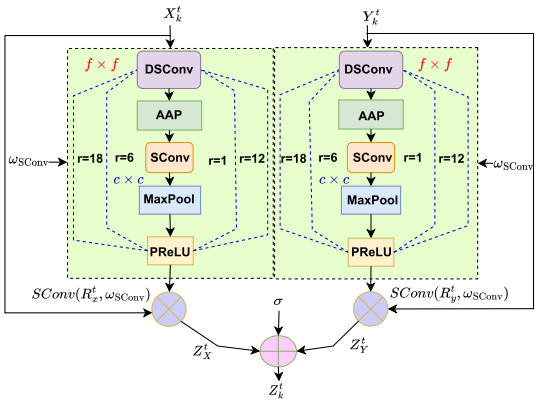

Fig. 5. Illustration of DAFM to extract the appearance and motion features and fuse these features.

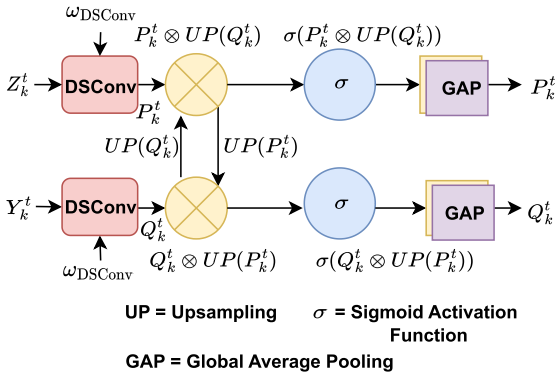

Fig. 6. BCFM to extract the motion and appearance features from fused appearance and motion features in a bidirectional way.

top-down approach and broadcasts high-level strong features to low-level weak features. Finally, the spatial attention features are computed as follows:

$$P_k^t = \mathrm{GAP}\left[\sigma\left(P_k^t \otimes \mathrm{UP}\left(Q_k^t\right)\right)\right] \tag{5}$$

where UP is a bilinear UP operation. The temporal attention features are derived as

$$Q_k^t = \mathrm{GAP}\left[\sigma\left(Q_k^t \otimes \mathrm{UP}\left(P_k^t\right)\right)\right]. \tag{6}$$

An SGFA is proposed for effective model training. The same DSCNet architecture is again trained using SGFA and named DSCNet+.

3) Stochastic-Gradient-Based Firefly Algorithm: The firefly optimization algorithm [45] is a multimodal meta-heuristic algorithm inspired by nature based on its flashing characteristic. Almost all the species of fireflies release distinctive small rhythmic flashes, which is called the bioluminescence process. The firefly algorithm has three main steps.

1) *Initialization Step:* The variables used in the optimization function and the corresponding value of the optimization function are considered when creating the solution search space at the initialization step. The initialization stage's randomness creates an unbalanced relationship between exploration and exploitation in the initial solution space, slowing down the algorithm's local and global convergence rates and resulting in a lower quality solution.
2) *Firefly Position Update Step:* The firefly positions are updated during the stage of altering firefly positions to find new solutions to the specified problem. This element is known as the randomization factor. During the position changing of fireflies, the movements are decided by randomization factor. The quality of the solution deteriorates if these values are not managed properly

$$\omega_i = \omega_i + \beta_0 e^{-\gamma d_{i,j}^2}(\omega_j - \omega_i) + \eta\left(\mathrm{rand} - \frac{1}{2}\right) \tag{7}$$

where, $\eta_{t+1} = \eta_t \delta$ is the randomization parameter with the initial randomness scaling factor η_0 set as 1, "rand" is the random number drawn from Gaussian distribution between [0, 1], and $\delta \in [0.95, 0.97]$ is the cooling factor, p_i and p_j are the positions of the ith and jth fireflies, respectively, $\beta_0 e^{-\gamma d_{i,j}^2}$ is the attractiveness at distance $d_{i,j}$, where $d_{i,j} = \|p_i - p_j\|$ defines the distance between ith and jth fireflies, $\|\cdot\|$ is the l_2-norm, and $\gamma = (1/\sqrt{\eta})$ is a scaling factor.
3) *Termination Step:* The algorithm ends at the termination phase.

To maintain the balance between exploration and exploitation of the solution search space, the SGD-optimized weights are used in the firefly initialization step. Thus, SGFA-provided

Algorithm 1 Stochastic Gradient Firefly Algorithm

Input : Define initial values of firefly parameters: β_0, γ, η, SGD weights as x, and number of fireflies N

Output: Optimal model weights (BS) and its fitness

Evaluate light intensity l at x of firefly by f(x) using 9.
//Initialization
while $t < MaxGen$ **do**
  for $i \in N$ **do**
    for $j \in N$ **do**
      Update **if** $l_i < l_j$ **then**
        Move firefly i toward firefly j according to 7
      end
      Calculate attractiveness variance with distance $d_{i,j}$ using $\beta = \beta_0 e^{-\gamma d_{i,j}^2}$ with reduced η by factor δ.
    end
  end
  Update firefly light intensity l_i
  Rank the fireflies and accept the new solution (BS) as the current best with fitness.
  Update iteration counter $t = t + 1$
end
return (BS)

optimal weights are two-step fine-tuned. At first, weights are fine-tuned using SGD as follows:

$$\omega_{i+1} = \omega_i - \alpha \times \frac{\partial f}{\partial \omega} \tag{8}$$

where α is the learning rate and $(\partial f/\partial \omega)$ is the gradient of the BCE loss function. Next, SGD weights are used in the FF initialization step followed by the position update step. In the FF algorithm, firefly's particles have been encoded as model weights and the BCE loss function as the fitness function. The BCE loss function is given (9). For better model weights, the loss function needs to be minimized. The detailed explanation of how SGFA is used for VSOD is stated in Algorithm 1

$$\text{BCE} = -\left\{G_k^t \log\left(S_{A,M}^t\right) + \left(1 - G_k^t\right)\log\left(1 - S_{A,M}^t\right)\right\} \tag{9}$$

where $S_{A,M}^t$ is the saliency map of appearance and motion features, and G_k^t is the ground truth.

4) Saliency Prediction Module: Generally, the appearance of salient objects does not isolate. Salient objects frequently coexist with other items and are constantly surrounded by a background to connect their neighbors' pixels. These contexts give essential information to distinguish the salient object from the background. In addition, salient objects frequently take up a significant portion of the image and catch people's attention. Given these facts, contextual attention is added to feature fusion and network learning processes. It compels the backbone network to concentrate on the most essential objects or regions and reduce the detrimental effects of the background. The upper layer of our proposed models is used

Algorithm 2 Dilation SConv Net (DSCNet).

Input: A_k^t: Appearance frames, M_k^t: motion frames, and G_k^t: ground truth

Output: $S_{A,M}^t$: Saliency prediction map.

BSCNet is used to calculate motion frames M_k^t using consecutive appearance frames (X_{k+1}^t, X_k^t).
DSCNet is used as input as A_k^t, M_k^t, and G_k^t and passed to ResNet-50.
ResNet-50 backbone network generated backbone appearance and motion features (X_k^t, Y_k^t).
The backbone appearance and motion features passed to the DAFM to generate the fused appearance and motion features Z_k^t using *Eq.* 1–3.
The fused and motion features passed to the BCFM to extract the attention-based spatial and temporal features using *Eq.* 4–6.
To update the model weight parameter, Adam optimizer (SGFA in case of DSCNet+) is used, which minimizes BCE loss function using *Eq.* 7–9.
At last, SPM is used to generate the saliency map using *Eq.* 10–12 respectively.

to generate the attention mask from high-level to low-level features. It captures more extensive context areas and contains more detailed object information.

As shown in Fig. 7, SPM uses three $c \times c$ convolution layer to extract global high-level spatio-temporal features. Parallelly, both the motion–appearance features are passed to the convolution block to transform into linear vectors. To retain the quality of the features, they are passed to the corresponding parallel node before applying the PReLU operation; otherwise, the characteristics of the features are going to change. Then, nonlinearity is handled by the PReLU activation function. Furtherm0re, these three features are going to add. The enhanced spatio-temporal high-level feature is generated by the below equation

$$P_k^t = \sum_{i=1}^{3} \text{PReLU}_i\left(\text{Conv}_i\left(P_k^t\right)\right) \tag{10}$$

$$Q_k^t = \sum_{i=1}^{3} \text{PReLU}_i\left(\text{Conv}_i\left(Q_k^t\right)\right). \tag{11}$$

Finally, after generating the high-level features' output, $c \times c$ convolution layers are used with single filters followed by the Sigmoid activation function to display the saliency map $(S_{A,M}^t)$ at frame t

$$S_{A,M}^t = \sigma\left(\text{Conv}\left(P_k^t, Q_k^t\right)\right). \tag{12}$$

IV. Experiments and Result Analysis

The section discusses the experimental setup, datasets, evaluation metrics, training performance, testing performance, computational complexity measure, comparative analysis, and parameter tuning on datasets.

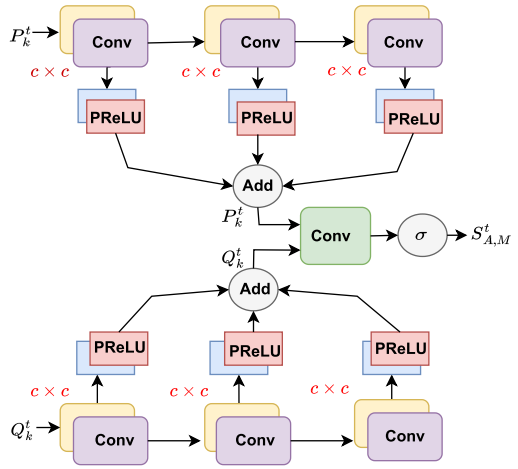

Fig. 7. Illustration of SPM to transform the features from high to low level and retain the global information.

A. Experimental Setup

All the implementations are performed on a 64-bit Ubuntu 18.04 system. The GPU configuration is 32-GB RAM, 16-GB P5000/PCIe/SSE2 GPU. Anaconda 3.7 and PyTorch [46] version 1.10.0 with CUDA 10.0 by NVIDIA Quadro P5000/PCIe/SSE2 is installed on the GPU machine. All the input frames are resized to 352 × 352. For optimizing the network, an Adam optimizer algorithm for DSCNet (SGFA for DSCNet+) is used with a weight decay of $5e^{-4}$ and the learning rate $\alpha = 1e^{-4}$. The multiscale training is used in the experimentation with weight parameters (0.25, 0.50, 0.95, and 1).

B. Datasets

The proposed models (DSCNet and DSCNet+) are evaluated on six benchmark VSOD datasets: 1) DAVIS-16[1] [54] is one of the popular datasets, which has 50 high-quality and densely annotated video sequences (30 training videos and 20 testing videos); 2) MCL[2] [55] has 24 videos; 3) FBMS[3] [56] consists of 59 videos of natural scenes (29 videos for training and 30 videos for testing); 4) SegTrack-V2[4] [27] is the advanced VSOD dataset having 13 video clips; 5) DAVSOD-Easy[5] [38] has 61 video clips for training and 35 clips for testing; and 6) DAVSOD-Difficult-20 [38] has 20 video clips for testing. We are using the DAVIS-16, FBMS, and DAVSOD-19 datasets for training purposes and the DAVIS-16, FBMS, DAVSOD-Easy, DAVSOD-Difficult-20, MCL, and SegTrack-V2 datasets for testing purposes.

C. Performance Metrics

The three standard metrics are used to measure the performance of the proposed models, structure measure (S_α, $\alpha = 0.5$), mean absolute error (MAE), and F-measure (F_β) [1], [5], [24].

[1]https://davischallenge.org/
[2]https://mcl.usc.edu/mcl-jcv-dataset/
[3]https://lmb.informatik.uni-freiburg.de/resources/datasets/
[4]https://web.engr.oregonstate.edu/~lif/SegTrack2/dataset.html
[5]https://github.com/DengPingFan/DAVSOD#statistics-of-davsod

TABLE I
PARAMETER TUNING OF THE DSCNET AND DSCNET+ MODELS

S.No.	Parameter	Value	Explanation
1	β_0	0.3	Initial brightness
2	N	30	Initial population size.
3	η	0.4	Randomness of the fliers.
4	MaxGen	1500	Maximum number of generations.
5	f × f, c × c	3 × 3, 1 × 1	Convolution filter.
6	h × w × c	352 × 352 × 3	Input frame features.
7	α	$1e^{-4}$	learning rate.

1) **S-measure** calculates the structural similarity between the saliency map and ground truth. It examines object-aware (S_o) and region-aware (S_r) structure similarities

$$S = \alpha \times S_o + (1 - \alpha) \times S_r \tag{13}$$

where, α is set to 0.5.

2) **F-measure** is a combination of precision and recall for calculating weighted harmonic mean

$$F_\beta = \frac{(1 + \beta^2)\text{Precision} \times \text{Recall}}{\beta^2\text{Precision} + \text{Recall}} \tag{14}$$

where, β is 0.3.

3) **MAE** calculates the average pixelwise absolute error between saliency map $S \in [0, 1]^{W \times H}$ and ground truth $G \in [0, 1]^{W \times H}$

$$\text{MAE} = \frac{1}{W \times H} \sum_{i=1}^{W} \sum_{j=1}^{H} |G(i, j) - S(i, j)|. \tag{15}$$

D. Training Performance

As discussed earlier, the multiscale lightweight motion map is first constructed using BSCNet. These enhanced motion features, along with the appearance features and ground truth, are passed to the ResNet-50 backbone network and then the proposed models (DSCNet and DSCNet+) for training. The training datasets are constituted of 6500 appearance and motion frames [2373 frames from DAVIS (30 videos) + 600 frames for FBMS (29 videos) + 3527 frames for DAVSOD-Easy (26 videos)]. These training datasets are used to train the DSCNet with the Adam optimizer (SGFA optimizer in case of DSCNet+) to minimize the BCE loss. The whole process is shown in Algorithm 2. The fine-tuned hyperparameter values are given in Table I. To control the vanishing gradient and overfitting problems, batch normalization and nonlinearity activation function (PReLU) is used to train the spatial branch. The proposed models take approximately 10 hours to perform 100 epochs with six batch sizes.

E. Parameter Tuning

The parameters are tuned with different values to check the output results in terms of F-measure and MAE. The greedy search technique is used to tune the parameters of the optimization algorithm. The values and ranges of tuning parameters are given in terms of brightness β_0, randomness η, population size (N), maximum generation (MaxGen), and learning rate α of the SGFA algorithm. The range considered for hyperparameter tuning of brightness is taken as (0.1–0.4),

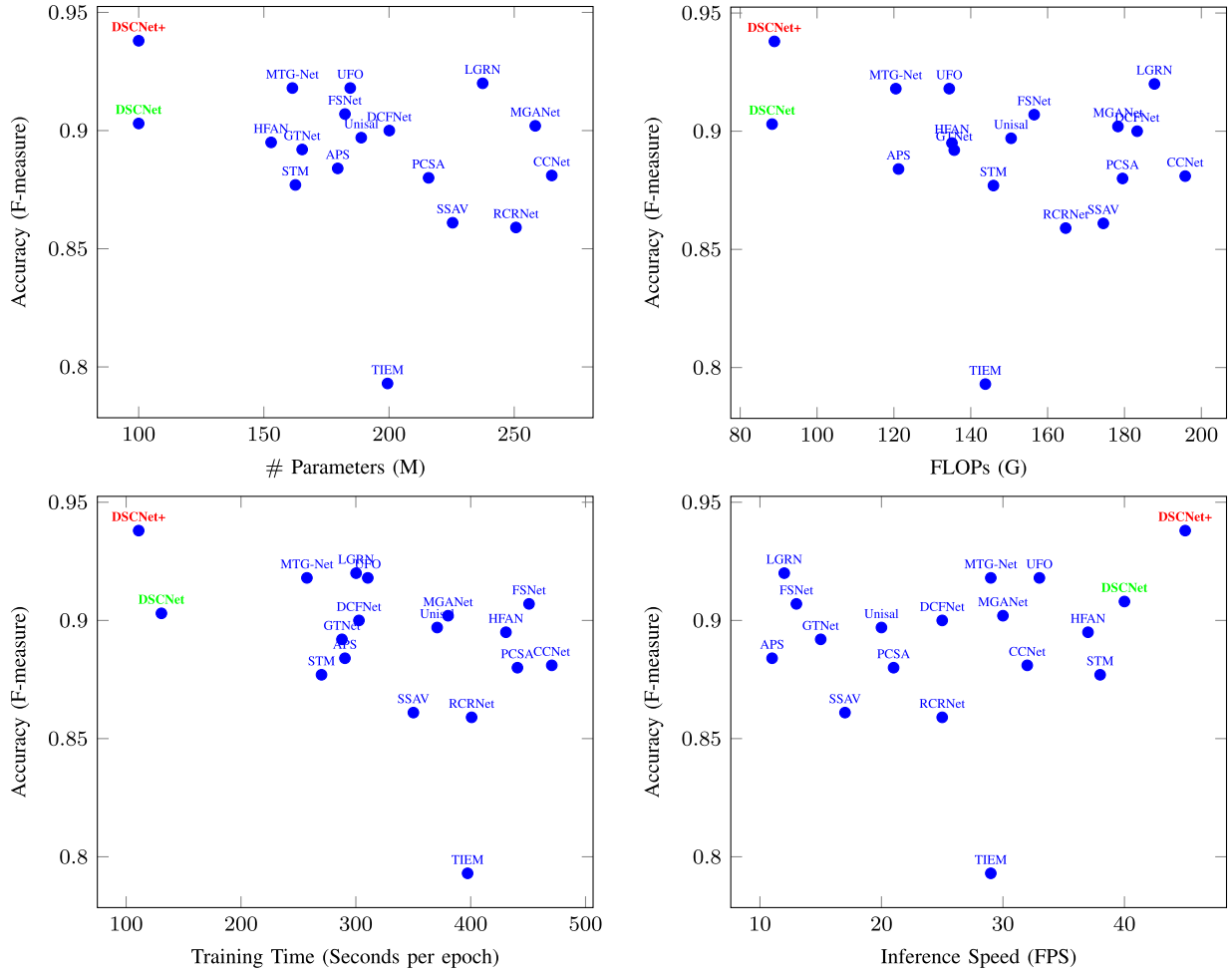

Fig. 8. Performance comparison of our proposed models (DSCNet and DSCNet+) and SOTA models on the DAVIS dataset in terms of accuracy versus number of parameters [in millions (M)], FLOPs, training time per epoch, and inference speed (FPS).

TABLE II

COMPARISONS BETWEEN THE PROPOSED MODELS (DSCNET AND DSCNET+) AND 16 SOTA MODELS ON SIX DATASETS. THE TOP THREE RESULTS ARE HIGHLIGHTED IN RED, GREEN, AND BLUE, AND ARS IS AVERAGE RANKING SCORE

	# Param	FLOPs	Speed	DAVIS			FBMS			DAVSOD			SegTrack-V2			MCL			DAVSOD-Diff			
Model Yr. Ref.	**(M)**	**(G)**	**(FPS)**	S_α	F_β	**MAE**	S_α	F_β	**MAE**	S_α	F_β	**MAE**	S_α	F_β	**MAE**	S_α	F_β	**MAE**	S_α	F_β	**MAE**	**ARS**
MGANet$_{19}$ [12]	258.4	178.3	30	0.913	0.902	0.022	0.907	0.910	0.027	0.687	0.678	0.083	0.837	0.698	0.053	0.774	0.759	0.096	0.450	0.429	0.143	13.8
RCRNet$_{19}$ [47]	250.7	164.7	25	0.884	0.859	0.029	0.870	0.861	0.039	0.695	0.683	0.089	0.812	0.694	0.032	0.763	0.769	0.056	0.644	0.444	0.097	4.3
Unisal$_{20}$ [48]	188.9	150.5	20	0.901	0.897	0.039	0.895	0.870	0.037	0.753	0.759	0.072	0.783	0.739	0.062	0.732	0.725	0.089	0.479	0.428v	0.127	10.2
PCSA$_{20}$[24]	215.8	179.5	25	0.902	0.880	0.022	0.866	0.831	0.041	0.741	0.655	0.086	0.865	0.810	0.025	0.747	0.827	0.067	0.457	0.412	0.137	14.3
CCNet$_{20}$ [43]	265.0	195.8	32	0.907	0.881	0.023	0.900	0.883	0.035	0.761	0.671	0.084	0.774	0.731	0.064	0.852	0.832	0.048	0.486	0.460	0.100	5.3
SSAV$_{20}$ [38]	225.4	174.5	17	0.893	0.861	0.028	0.879	0.865	0.040	0.724	0.603	0.092	0.851	0.801	0.025	0.856	0.798	0.073	0.619	0.399	0.114	9.6
FSNet$_{21}$ [1]	182.4	156.5	13	0.920	0.907	0.020	0.890	0.888	0.041	0.773	0.685	0.072	0.833	0.698	0.038	0.864	0.821	0.023	0.662	0.487	0.099	3.3
GTNet$_{21}$ [49]	165.3	135.7	15	0.912	0.892	0.022	0.887	0.865	0.022	0.760	0.673	0.074	0.810	0.753	0.047	0.853	0.819	0.027	0.553	0.453	0.109	6
LGRN$_{21}$ [50]	237.4	187.8	12	0.923	0.920	0.017	0.889	0.879	0.049	0.765	0.699	0.063	0.759	0.685	0.046	0.843	0.815	0.028	0.445	0.428	0.120	12.5
TIEM$_{21}$ [51]	199.4	143.8	29	0.846	0.793	0.038	0.803	0.792	0.073	0.705	0.605	0.103	0.819	0.762	0.033	0.695	0.601	0.096	0.459	0.334	0.118	13.3
APS$_{21}$ [44]	179.5	121.2	11	0.894	0.884	0.039	0.803	0.796	0.059	0.715	0.695	0.092	0.815	0.725	0.056	0.675	0.597	0.094	0.461	0.415	0.102	9.6
DCFNet$_{21}$ [18]	200.1	183.3	25	0.914	0.900	0.016	0.845	0.838	0.046	0.741	0.666	0.074	0.883	0.839	0.015	0.774	0.765	0.065	0.432	0.396	0.143	17.8
MTG-Net$_{22}$ [21]	161.4	120.5	29	0.925	0.918	0.015	0.901	0.890	0.033	0.766	0.756	0.045	0.893	0.849	0.014	0.843	0.832	0.034	0.484	0.439	0.117	8
STM$_{22}$ [11]	162.6	145.9	38	0.897	0.877	0.020	0.894	0.883	0.032	0.777	0.708	0.065	0.886	0.850	0.014	0.786	0.773	0.054	0.453	0.406	.132	15
UFO$_{22}$ [52]	184.5	134.4	33	0.925	0.918	0.015	0.901	0.890	0.033	0.766	0.756	0.045	0.893	0.849	0.013	0.843	0.832	0.034	0.478	0.421	0.129	11.5
HFAN$_{22}$ [53]	152.9	135.1	37	0.900	0.895	0.023	0.878	0.863	0.031	0.745	0.737	0.048	0.834	0.818	0.031	0.858	0.821	0.039	0.448	0.410	0.129	14.5
DSCNet(Prop.)	**100.0**	**88.3**	**40**	0.948	0.903	0.027	0.890	0.898	0.032	0.780	0.752	0.052	0.840	0.744	0.035	0.866	0.835	0.033	0.752	0.552	0.075	2
DSCNet+(Prop.)	**100.0**	**88.3**	**45**	0.953	0.938	0.014	0.912	0.908	0.016	0.799	0.765	0.040	0.901	0.878	0.028	0.872	0.842	0.022	0.772	0.573	0.071	1

TABLE III

COMPARISON STUDY OF DIFFERENT TYPES OF OPTIMIZATION ALGORITHM WITH THE PROPOSED MODEL (DSCNET+)

	Different Optimizer				FLOPs	DAVIS		MCL		FBMS		SegTrack-V2		DAVSOD		DAVSOD-Diff	
No.	**ADAM**	**SGD**	**ADAM+FOA**	**SGFA**	**(G)**	S_α	**MAE**	S_α	**MAE**	S_α	**MAE**	S_α	**MAE**	S_α	**MAE**	S_α	**MAE**
1	✓				259.3	0.862	0.043	0.835	0.043	0.866	0.042	0.860	0.042	0.760	0.052	0.560	0.125
2		✓			270.3	0.896	0.039	0.847	0.031	0.872	0.037	0.871	0.036	0.768	0.047	0.587	0.099
3			✓		224.8	0.901	0.036	0.859	0.030	0.889	0.036	0.883	0.035	0.769	0.043	0.639	0.089
4				✓	**100.8**	**0.953**	**0.014**	**0.872**	**0.022**	**0.912**	**0.016**	**0.901**	**0.028**	**0.799**	**0.040**	**0.772**	**0.071**

attractiveness is taken as (0.2–0.6), and population size N is (20–40). As the population size and learning rate increase, the maximum generation size of the proposed model is increasing, but performance is going to go down. So, to maintain

Fig. 9. Performance comparison of the proposed model (DSCNet+) and SOTA models. Where IF is the input frame, GT is the ground truth, MF is the motion frame, four is FSNet [1], the fifth is GTNet [49], sixth is LGRN [50], seventh is TIEM [51], and the proposed model (DSCNet+).

TABLE IV
PARAMETER TUNING OF SGFA OPTIMIZATION ALGORITHM IN THE PROPOSED MODEL (DSCNET+)

	Parameter Tuning					DAVIS		MCL		FBMS		SegTrack-V2		DAVSOD		DAVSOD-Diff	
No.	β_0	η	**Population**	**MaxGen**	α	S_α	**MAE**	S_α	**MAE**	S_α	**MAE**	S_α	**MAE**	S_α	**MAE**	S_α	**MAE**
1	0.1	0.2	20	500	0.01	0.891	0.024	0.847	0.032	0.864	0.028	0.865	0.038	0.763	0.054	0.495	0.116
2	0.2	0.3	25	100	0.001	0.918	0.021	0.861	0.028	0.887	0.024	0.883	0.035	0.771	0.048	0.531	0.091
3	**0.3**	**0.4**	**30**	**1500**	**0.0001**	**0.953**	**0.014**	**0.872**	**0.022**	**0.912**	**0.016**	**0.901**	**0.028**	**0.799**	**0.040**	**0.772**	**0.071**
4	0.4	0.6	40	2000	0.00001	0.936	0.020	0.857	0.026	0.894	0.022	0.885	0.034	0.778	0.047	0.627	0.099

the performance and iteration size, best combinations of the parameter are shown in Table IV.

F. Testing Performance

The performance of the proposed models (DSCNet and DSCNet+) is tested on the DAVIS-16 [54] test dataset that contains 20 videos, the FBMS [56] test dataset that contains 30 videos, the DAVSOD-Easy [38] test dataset that contains 35 videos, the DAVSOD-Difficult-20 [38] test datasets that contain 20 videos, the MCL [55] test dataset that contains nine videos, and the SegTrack-V2 [27] test dataset that contains 13 videos. For testing, no preprocessing and postprocessing techniques are used. The performance of the proposed models (DSCNet and DSCNet+) is estimated in terms of S_α, F_β, and MAE. Furthermore, the performance of the proposed models (DSCNet and DSCNet+) is estimated in terms of complexity and computational measures such as the number of hyperparameters, FLOPs, and speed. The performance results are shown in Table II. We have shown the rankwise list of the best-performing approaches in terms of average ranking score of F-measure, S-measure, and MAE, on the DAVSOD-Diff dataset.

G. Computation Complexity Measures

The main aim of this article is to design such a model, which gives a robust solution without increasing the network's computational complexity for VSOD. We compared the computational costs of several techniques in terms of the number of model parameters (# Param), the number of floating-point operations (FLOPs), and inference speed [frames per second (FPS)]. The results are shown in Fig. 8. Here, the number of parameters is measured in million (M), and the GPU memory usage is measured in gigabytes (G) FLOPs. The inference speed is measured by per second number of frames processed, and runtime is measured in seconds per epoch processed.

H. Comparative Analysis

The proposed models; (DSCNet and DSCNet+) test performance is compared with 16 SOTA VSOD models (refer Table II) in terms of S_α, F_β, and MAE. Also, the proposed

TABLE V
ABLATION STUDIES FOR THE COMPONENTS' SETTING OF DSCNET+

	Component Setting			DAVIS		MCL		FBMS		SegTrack-V2		DAVSOD		DAVSOD-Diff	
No.	SPM	DAFM	BCFM	S_α	MAE	S_α	MAE	S_α	MAE	S_α	MAE	S_α	MAE	S_α	MAE
1				0.853	0.039	0.678	0.069	0.783	0.063	0.738	0.069	0.723	0.056	0.334	0.143
2	✓			0.857	0.038	0.748	0.063	0.785	0.059	0.721	0.075	0.736	0.064	0.335	0.142
3		✓		0.859	0.040	0.758	0.069	0.763	0.053	0.700	0.075	0.733	0.068	0.338	0.139
4			✓	0.867	0.043	0.754	0.064	0.789	0.067	0.753	0.060	0.760	0.046	0.337	0.137
5	✓	✓		0.868	0.054	0.769	0.058	0.758	0.063	0.753	0.048	0.727	0.056	0.443	0.130
6		✓	✓	0.865	0.044	0.743	0.059	0.773	0.053	0.748	0.057	0.766	0.053	0.458	0.123
7	✓		✓	0.895	0.038	0.818	0.029	0.863	0.035	0.748	0.037	0.766	0.043	0.535	0.098
8	✓	✓	✓	**0.953**	**0.014**	**0.872**	**0.022**	**0.912**	**0.016**	**0.901**	**0.028**	**0.799**	**0.040**	**0.772**	**0.071**

TABLE VI
ABLATION STUDIES FOR THE DESIGN CHOICE OF DSCNET+

	Parameters		DAVIS		MCL		FBMS		SegTrack-V2		DAVSOD		DAVSOD-Diff	
No.	# filters	Dilation rates	S_α	MAE	S_α	MAE	S_α	MAE	S_α	MAE	S_α	MAE	S_α	MAE
1	2	1	0.853	0.039	0.678	0.069	0.783	0.063	0.738	0.069	0.723	0.056	0.334	0.143
2	4	1,2	0.857	0.038	0.748	0.063	0.785	0.059	0.721	0.075	0.736	0.064	0.335	0.142
3	8	1,2,3	0.859	0.040	0.758	0.069	0.763	0.053	0.700	0.075	0.733	0.068	0.338	0.139
4	16	1,2,3	0.867	0.043	0.754	0.064	0.789	0.067	0.753	0.060	0.760	0.046	0.337	0.137
5	**32**	**1,2,3, 4**	**0.953**	**0.014**	**0.872**	**0.022**	**0.912**	**0.016**	**0.901**	**0.028**	**0.799**	**0.040**	**0.772**	**0.071**
6	64	4, 3,	0.868	0.054	0.769	0.058	0.758	0.063	0.753	0.048	0.727	0.056	0.343	0.130
7	128	4,3,2	0.865	0.044	0.743	0.059	0.773	0.053	0.748	0.057	0.766	0.053	0.358	0.123
8	256	4, 3, 2, 1	0.895	0.038	0.818	0.029	0.863	0.035	0.748	0.037	0.766	0.043	0.435	0.098

models are compared with the SOTA models regarding complexity and computational measures. These comparative results are shown in Table II. The table shows that the proposed models outperform 16 SOTA models on the DAVIS, DAVSOD, MCL, and DAVSOD-Difficult datasets regarding complexity and computations. The proposed models generate saliency maps accurately in less time. The comparison between the accuracy and efficiency of the proposed models (DSCNet and DSCNet+) with various SOTA approaches are shown in Fig. 8. From Fig. 8, we see that the proposed models use significantly fewer parameters and FLOPs and are substantially faster than previous SOTA approaches to achieve better accuracy. Using the SConv, performing multiscale dilation operations with the SGFA optimizer, and fusing the feature in bidirectional ways cause the reduction of the model parameters. This reduction of model parameters increases the inference speed and decreases the training time. Thus, it can be concluded that the proposed models (DSCNet and DSCNet+) perform a great trade between accuracy, the number of parameters, the number of FLOPs, and speed. The proposed models (DSCNet and DSCNet+) are located in the top-left corner of the subfigures of F-measure versus # parameters, F-measure versus FLOPs, and F-measure versus training time in seconds per epoch. The inference speed F-measure versus FPS is shown in the right top corners. The performance of the proposed model (DSCNet+) is also performed based on the different types of the optimizer as shown in Table III, which shows that when we use SGFA, the result increases when compared with other optimizers.

1) Qualitative Comparison: The proposed models (DSCNet and DSCNet+) are visually compared with four SOTA methods in Fig. 9 under various difficult scenes. DSCNet+ can separate salient objects with coherent borders in a variety of difficult situations, including cluttered background with low light (fourth and fifth rows), noise between foreground and background (first rows), occluding objects with motion blur (third and fifth rows), motion blur with illumination scenarios (first, fourth, and fifth rows), and perplexing natural scenarios with deformation (first, second, sixth, seventh, and eighth rows). Given the simplicity and effectiveness of DSCNet+, it supports real-world VSOD applications. Our proposed models outperform the above scenes and demonstrate great flexibility.

I. Evaluation on DAVSOD-Diff

The DAVSOD-Diff [1] dataset is the most challenging dataset, containing multiple instances to represent the wild/unconstrained scenarios. Hence, the results we obtain on this dataset present the significance of our contribution. From Table II, it can be observed that the performances of all the other models drastically decrease; however, our proposed models are able to maintain the performance. In comparison to the (previous), best FSNet model [1], our proposed models increase S_α by 16.62%, F_β by 17.65%, and MAE by 28.3%. Compared with the second-best SSAV model, our proposed models increase S_α by 24.71%, F_β by 43.60%, and MAE by 37.71%. Similarly, in comparison to the third-best RCRNet model [47], our proposed models increase S_α by 19.87%, F_β by 29.05%, and MAE by 26.80%. In contrast to the recently proposed models, RCRNet, which uses pseudolabels, and SSAV, which uses a validation set, our model does not use additional training data. In addition, recent findings show that "human visual attention" should be an underlying method that drives VSOD. The quantitative comparison of SOTA and proposed models is given in Table II, where the average ranking score is calculated using the S-measure, F-measure, and MAE results for all the models on the DAVSOD-Diff dataset.

J. Ablation Study

We perform an ablation study to show how well the proposed models' (DSCNet and DSCNet+) components and parameter configurations work. The experimental settings are the same as those in the training Section IV-D. The ablation

TABLE VII

ABLATION STUDIES FOR EXTRACTING THE MOTION MAP WITH SCM AND FLOWNET2.0 IN DSCNET AND DSCNET+

	Component Setting				DAVIS		MCL		FBMS		SegTrack-V2		DAVSOD		DAVSOD-Diff	
No.	SCM	FlowNet2.0	DSCNet	DSCNet+	S_α	MAE	S_α	MAE	S_α	MAE	S_α	MAE	S_α	MAE	S_α	MAE
1			✓		0.869	0.034	0.837	0.041	0.879	0.040	0.801	0.044	0.746	0.068	0.538	0.134
2				✓	0.873	0.031	0.840	0.037	0.881	0.039	0.810	0.042	0.749	0.066	0.551	0.131
3		✓	✓		0.879	0.029	0.844	0.032	0.884	0.038	0.817	0.038	0.753	0.064	0.560	0.135
4		✓		✓	0.888	0.026	0.849	0.029	0.885	0.037	0.823	0.035	0.760	0.061	0.567	0.137
5	✓		✓		0.895	0.022	0.854	0.027	0.887	0.036	0.829	0.039	0.766	0.058	0.579	0.135
6	✓			✓	0.903	0.020	0.859	0.026	0.889	0.034	0.835	0.037	0.770	0.056	0.627	0.137
7	✓	✓	✓		0.948	0.017	0.866	0.024	0.890	0.032	0.840	0.035	0.780	0.052	0.676	0.075
8	✓	✓		✓	**0.953**	**0.014**	**0.872**	**0.022**	**0.912**	**0.016**	**0.901**	**0.028**	**0.799**	**0.040**	**0.772**	**0.071**

TABLE VIII

ABLATION STUDIES OF VARIOUS SCM MODULE FUSION WITH FLOWNET2.0 OF THE PROPOSED MODELS (DSCNET AND DSCNET+). HERE, N IS THE NUMBER OF SCM MODULES

	DAVIS		MCL		FBMS		SegTrack-V2		DAVSOD		DAVSOD-Diff	
SCM	S_α	**MAE**	S_α	**MAE**	S_α	**MAE**	S_α	**MAE**	S_α	**MAE**	S_α	**MAE**
N=0	0.887	0.030	0.844	0.034	0.839	0.034	0.848	0.035	0.760	0.049	0.537	0.083
N=1	0.889	0.028	0.859	0.028	0.858	0.027	0.853	0.033	0.767	0.056	0.543	0.087
N=2	0.895	0.023	0.863	0.031	0.873	0.023	0.878	0.032	0.776	0.050	0.558	0.091
N=3	0.918	0.019	0.869	0.029	0.897	0.019	0.888	0.031	0.786	0.043	0.635	0.087
N=4	**0.953**	**0.014**	**0.872**	**0.022**	**0.912**	**0.016**	**0.901**	**0.028**	**0.799**	**0.040**	**0.772**	**0.071**
N=5	0.925	0.018	0.868	0.026	0.888	0.030	0.789	0.038	0.769	0.043	0.635	0.098
N=6	0.917	0.022	0.855	0.029	0.863	0.035	0.768	0.037	0.766	0.042	0.635	0.097

Fig. 10. Failure case of our proposed (DSCNet and DSCNet+) models. Where IF is the input frame, GT is the ground truth, and the proposed models.

study findings for the suggested module components are displayed in Table V. These fundamental components are one-by-one included in the proposed (DSCNet and DSCNet+) models to provide effective and efficient multiscale learning. According to Table V, the performance is progressively improved as each component is added to the framework. In addition, the comparison between No. 1 and No. 8 shows the proposed solution's superiority to the baseline, where the performance gap is completely attributable to our contributions because the two models are each trained from scratch. The findings of the ablation investigation are presented in Table VI for various network configurations. It is intriguing to learn that the proposed DSCNet+ is resilient to minor separable configuration changes. From Table VII, we see that the component setting of BDCNet shows the important contribution of increasing the performance of DSCNet and DSCNet+. Rows 1 and 2 show without motion, which does not perform well. As the component is added to DSCNet and DSCNet+, the performance increases. Furthermore, from Table VIII, as the component (SCM) is added to FlowNet2.0, the performance of DSCNet and DSCNet+ is increased. FlowNet2.0 depends on the component of SCM. As the SCM is added to FlowNet2.0 and fused the feature in bidirectional, the quality of the motion map increases. Due to that, the overall performance of the proposed models (DSCNet and DSCNet+) is increased.

K. Failure Case

We choose a few illustrative failure scenarios and compare the outcomes of our proposed models. As shown in Fig. 10, the DSCNet model is unable to distinguish the border between an object and its background in a crowded scenario along with the complicated illumination (rows 1–3), whereas it is correctly detected by DSCNet+. Moreover, the DSCNet model fails to discern depth information because it cannot discriminate salient objects when the foreground has a huge nonsalient object and the target object is in the background (rows 1 and 2). Due to memory or computational constraints, our model cannot input exceptionally lengthy frames to assess whether the current item is salient in the last scenario with long-term temporal dependencies (rows 4 and 5).

V. CONCLUSION

In this article, we handle the problem of salient object detection in videos containing unconstrained scenarios. The contribution of this article is multifold. First, we propose a novel BSCNet, which uses SCM and FlowNet2.0 to extract lightweight multiscale contextual information across appearance cues and generate enhanced multiscaled motion maps. Second, a novel DSCNet model is proposed for detecting

salient objects effectively with the help of enhanced multi-scaled motion and appearance features. DSCNet is made up of multiple novel components, namely, the DAFM, BCFM, and SPM, which collectively handle multiple challenges, such as partial occlusion, motion blur, noise, and clutter background, present in a real-time environment. Third, for faster and better training of the DSCNet model, we propose a novel SGFA, which adaptively balances the exploration and exploitation in multiscaled, cross-modal embedded subspaces. The model we train with the proposed SGFA algorithm creates DSCNet+ on top of DSCNet, which further improves the results in terms of the training speed and other evaluation metrics. The performance of both the proposed models, DSCNet and DSCNet+, is evaluated on six publicly available benchmark datasets. With the help of extensive experimentation and comparative study, we conclude that our proposed models outperforms 16 SOTA models in terms of S-measure, F-measure, MAE, number of parameters, FLOPs, and FPS. One of the major highlights of the work is the significant performance of the proposed models on the most difficult DAVSOD-Diff dataset.

REFERENCES

[1] G.-P. Ji, K. Fu, Z. Wu, D.-P. Fan, J. Shen, and L. Shao, "Full-duplex strategy for video object segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 4902–4913.

[2] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, "1 year, 1000 km: The Oxford RobotCar dataset," *Int. J. Robot. Res.*, vol. 36, no. 1, pp. 3–15, Jan. 2017.

[3] G.-P. Ji et al., "Progressively normalized self-attention network for video polyp segmentation," 2021, *arXiv:2105.08468*.

[4] M. Ding, Z. Wang, B. Zhou, J. Shi, Z. Lu, and P. Luo, "Every frame counts: Joint learning of video segmentation and optical flow," in *Proc. Conf. AAAI Artif. Intell.*, vol. 34, no. 7, 2020, pp. 10713–10720.

[5] H. Song, W. Wang, S. Zhao, J. Shen, and K.-M. Lam, "Pyramid dilated deeper ConvLSTM for video salient object detection," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Oct. 2018, pp. 744–760.

[6] M. Lan, Y. Zhang, Q. Xu, and L. Zhang, "E3SN: Efficient end-to-end Siamese network for video object segmentation," in *Proc. 29th Int. Joint Conf. Artif. Intell.*, Jul. 2020, pp. 701–707.

[7] G. Bhat et al., "Learning what to learn for video object segmentation," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, Nov. 2020, pp. 777–794.

[8] A. K. Gupta, A. Seal, P. Khanna, E. Herrera-Viedma, and O. Krejcar, "ALMNet: Adjacent layer driven multiscale features for salient object detection," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–14, 2021.

[9] G. Song, K. Song, and Y. Yan, "EDRNet: Encoder–decoder residual network for salient object detection of strip steel surface defects," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 12, pp. 9709–9719, Dec. 2020.

[10] F. Huo, X. Zhu, Q. Zhang, Z. Liu, and W. Yu, "Real-time one-stream semantic-guided refinement network for RGB-thermal salient object detection," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–12, 2022.

[11] X. Zhao et al., "Motion-aware memory network for fast video salient object detection," 2022, *arXiv:2208.00946*.

[12] H. Li, G. Chen, G. Li, and Y. Yu, "Motion guided attention for video salient object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 7273–7282.

[13] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox, "FlowNet 2.0: Evolution of optical flow estimation with deep networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1647–1655.

[14] Z. Teed and J. Deng, "Raft: Recurrent all-pairs field transforms for optical flow," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, Nov. 2020, pp. 402–419.

[15] Y. Li, Z. Shen, and Y. Shan, "Fast video object segmentation using the global context module," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, Nov. 2020, pp. 735–750.

[16] F. Lin, Y. Chou, and T. Martinez, "Flow adaptive video object segmentation," *Image Vis. Comput.*, vol. 94, Feb. 2020, Art. no. 103864.

[17] P. Tokmakov, K. Alahari, and C. Schmid, "Learning video object segmentation with visual memory," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 4481–4490.

[18] M. Zhang et al., "Dynamic context-sensitive filtering network for video salient object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 1533–1543.

[19] X. Tian, K. Xu, X. Yang, L. Du, B. Yin, and R. W. H. Lau, "Bi-directional object-context prioritization learning for saliency ranking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 5872–5881.

[20] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," 2017, *arXiv:1706.05587*.

[21] D. Min, C. Zhang, Y. Lu, K. Fu, and Q. Zhao, "Mutual-guidance transformer-embedding network for video salient object detection," *IEEE Signal Process. Lett.*, vol. 29, pp. 1674–1678, 2022.

[22] T. Zhou, S. Wang, Y. Zhou, Y. Yao, J. Li, and L. Shao, "Motion-attentive transition for zero-shot video object segmentation," in *Proc. AAAI*, vol. 34, no. 7, 2020, pp. 13066–13073.

[23] S. W. Oh, J.-Y. Lee, K. Sunkavalli, and S. J. Kim, "Fast video object segmentation by reference-guided mask propagation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7376–7385.

[24] Y. Gu, L. Wang, Z. Wang, Y. Liu, M.-M. Cheng, and S.-P. Lu, "Pyramid constrained self-attention network for fast video salient object detection," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, vol. 34, no. 7, 2020, pp. 10869–10876.

[25] J. Cheng, Y.-H. Tsai, W.-C. Hung, S. Wang, and M.-H. Yang, "Fast and accurate online video object segmentation via tracking parts," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7415–7424.

[26] W. Wang, J. Shen, X. Lu, S. C. H. Hoi, and H. Ling, "Paying attention to video object pattern understanding," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 7, pp. 2413–2428, Jul. 2021.

[27] F. Li, T. Kim, A. Humayun, D. Tsai, and J. M. Rehg, "Video segmentation by tracking many figure-ground segments," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 2192–2199.

[28] S. Seo, J.-Y. Lee, and B. Han, "URVOS: Unified referring video object segmentation network with a large-scale benchmark," in *Proc. 16th Eur. Conf. Comput. Vis. (ECCV)*. Glasgow, U.K.: Springer, Aug. 2020, pp. 208–223.

[29] F. Perazzi, A. Khoreva, R. Benenson, B. Schiele, and A. Sorkine-Hornung, "Learning video object segmentation from static images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3491–3500.

[30] P. Voigtlaender, Y. Chai, F. Schroff, H. Adam, B. Leibe, and L.-C. Chen, "FEELVOS: Fast end-to-end embedding learning for video object segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9473–9482.

[31] W. Wang, J. Shen, J. Xie, and F. Porikli, "Super-trajectory for video segmentation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1680–1688.

[32] F. Perazzi, O. Wang, M. Gross, and A. Sorkine-Hornung, "Fully connected object proposals for video segmentation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3227–3234.

[33] Y. Xu, D. Song, and A. Hoogs, "An efficient online hierarchical supervoxel segmentation algorithm for time-critical applications," in *Proc. Brit. Mach. Vis. Conf.*, 2014, pp. 1–12.

[34] W. Wang et al., "Learning unsupervised video object segmentation through visual attention," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3059–3069.

[35] N. Ballas, L. Yao, C. Pal, and A. Courville, "Delving deeper into convolutional networks for learning video representations," 2015, *arXiv:1511.06432*.

[36] M. Siam et al., "Video object segmentation using teacher–student adaptation in a human robot interaction (HRI) setting," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 50–56.

[37] Z. Chen, C. Guo, J. Lai, and X. Xie, "Motion-appearance interactive encoding for object segmentation in unconstrained videos," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 6, pp. 1613–1624, Jun. 2020.

[38] D.-P. Fan, W. Wang, M.-M. Cheng, and J. Shen, "Shifting more attention to video salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 8546–8556.

[39] Y. Liu, X.-Y. Zhang, J.-W. Bian, L. Zhang, and M.-M. Cheng, "SAM-Net: Stereoscopically attentive multi-scale network for lightweight salient object detection," *IEEE Trans. Image Process.*, vol. 30, pp. 3804–3814, 2021.

[40] P. Hu, G. Wang, X. Kong, J. Kuen, and Y.-P. Tan, "Motion-guided cascaded refinement network for video object segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1400–1409.

[41] Y.-T. Hu, J.-B. Huang, and A. G. Schwing, "Unsupervised video object segmentation using motion saliency-guided spatio-temporal propagation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 786–802.

[42] J. Cheng, Y.-H. Tsai, S. Wang, and M.-H. Yang, "SegFlow: Joint learning for video object segmentation and optical flow," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 686–695.

[43] Z. Wang, J. Li, and Z. Pan, "Cross complementary fusion network for video salient object detection," *IEEE Access*, vol. 8, pp. 201259–201270, 2020.

[44] X. Zhao, Y. Pang, J. Yang, L. Zhang, and H. Lu, "Multi-source fusion and automatic predictor selection for zero-shot video object segmentation," in *Proc. 29th ACM Int. Conf. Multimedia*, Oct. 2021, pp. 2645–2653.

[45] X-S. Yang, "Firefly algorithm, stochastic test functions and design optimisation," *Int. J. Bio-Inspired Comput.*, vol. 2, no. 2, pp. 78–84, 2010.

[46] A. Paszke et al., "PyTorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, Dec. 2019, pp. 8026–8037.

[47] P. Yan et al., "Semi-supervised video salient object detection using pseudo-labels," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 7283–7292.

[48] R. Droste, J. Jiao, and J. A. Noble, "Unified image and video saliency modeling," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 419–435.

[49] Y. Jiao et al., "Guidance and teaching network for video salient object detection," 2021, *arXiv:2105.10110*.

[50] Y. Tang, Y. Li, and G. Xing, "Video salient object detection via adaptive local–global refinement," 2021, *arXiv:2104.14360*.

[51] W. Zhao, J. Zhang, L. Li, N. Barnes, N. Liu, and J. Han, "Weakly supervised video salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 16821–16830.

[52] Y. Su, J. Deng, R. Sun, G. Lin, and Q. Wu, "A unified transformer framework for group-based segmentation: Co-segmentation, co-saliency detection and video salient object detection," 2022, *arXiv:2203.04708*.

[53] G. Pei, F. Shen, Y. Yao, G.-S. Xie, Z. Tang, and J. Tang, "Hierarchical feature alignment network for unsupervised video object segmentation," 2022, *arXiv:2207.08485*.

[54] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung, "A benchmark dataset and evaluation methodology for video object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 724–732.

[55] H. Wang et al., "MCL-JCV: A JND-based H.264/AVC video quality assessment dataset," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2016, pp. 1509–1513.

[56] T. Brox, J. Malik, and P. Ochs, "Freiburg-berkeley motion segmentation dataset (FBMS-59)," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2010, p. 9.

Hemraj Singh received the Diploma degree in computer science and engineering from BTEUP, Lucknow, UP, India, in 2013, the B.Tech. degree in computer science and engineering from AKTU University, Lucknow, Uttar Pradesh, in 2017, and the M.Tech. degree in artificial intelligence from NIT Uttarakhand, Srinagar, India, in 2020. He is currently pursuing the Ph.D. degree with the Department of Computer Science and Engineering, National Institute of Technology, Warangal, Hanamkonda, Telangana, India.

His research interests include computer vision and image processing, video processing, artificial intelligence, machine learning, and deep learning.

Mridula Verma (Member, IEEE) received the M.Tech. degree in CSE from the Indian Institute of Technology, Roorkee, India, in 2009, and the Ph.D. degree in CSE from the Indian Institute of Technology (BHU), Varanasi, India, in 2017.

She is an Assistant Professor and the Head of the Artificial Intelligence and Machine Learning Laboratory, Institute for Development and Research in Banking Technology (IDRBT), Hyderabad, India. Her research interests include practical machine learning, federated learning, privacy-preserving machine learning, and financial NLP.

Ramalingaswamy Cheruku (Member, IEEE) received the B.Tech. degree in CSE from JNT University, Kakinada Campus, Kakinada, India, in 2008, the M.Tech. degree in CSE from ABV-IIIT at Gwalior, Gwalior, India, in 2011, and the Ph.D. degree in CSE from NIT Goa, Ponda, Goa, India, in 2018.

He is currently an Assistant Professor with the Department of CSE, National Institute of Technology, Warangal, Hanamkonda, Telangana, India. He has published more than 30 journal articles and 20 conference papers in reputed venues.