S ^{3} Net: Self-Supervised Self-Ensembling Network for Semi-Supervised RGB-D Salient Object Detection

Lei Zhu, Xiaoqiang Wang, Ping Li, *Member, IEEE*, Xin Yang, Qing Zhang, Weiming Wang, Carola-Bibiane Schönlieb, and C. L. Philip Chen, *Fellow, IEEE*

***Abstract*—RGB-D salient object detection aims to detect visually distinctive objects or regions from a pair of the RGB image and the depth image. State-of-the-art RGB-D saliency detectors are mainly based on convolutional neural networks but almost suffer from an intrinsic limitation relying on the labeled data, thus degrading detection accuracy in complex cases. In this work, we present a self-supervised self-ensembling network (S ^{3} Net) for semi-supervised RGB-D salient object detection by leveraging the unlabeled data and exploring a self-supervised learning mechanism. To be specific, we first build a self-guided convolutional neural network (SG-CNN) as a baseline model by developing a series of three-layer cross-model feature fusion (TCF) modules to leverage complementary information among depth and RGB modalities and formulating an auxiliary task that predicts a self-supervised image rotation angle. After that, to further explore the knowledge from unlabeled data, we assign SG-CNN to a student network and a teacher network, and encourage the saliency predictions and self-supervised rotation predictions from these two networks to be consistent on the unlabeled data. Experimental results on seven widely-used benchmark datasets demonstrate that our network quantitatively and qualitatively outperforms the state-of-the-art methods.**

***Index Terms*—RGB-D salient object detection, self-supervised learning, semi-supervised learning, and cross-model and cross-level feature aggregation.**

I. Introduction

Salient object detection (SOD) aims to distinguish the most visually distinctive objects from a single input image [1]–[8]. By acting as an effective pre-processing step, RGB-D SOD benefits diverse image processing and computer vision tasks *e.g.*, object segmentation [9] and tracking [10], video compression [11] and abstraction [12], image editing [13], and texture smoothing [14]. Although the existing methods [15], [16] achieve remarkable results, they usually rely on the individual RGB/color images or video sequences, but totally ignore the depth information, which is now easily obtained by using Kinect, RealSense, and modern smartphones (*e.g.*, iPhone X, Huawei Mate10, and Samsung Galaxy S10) [17], [18]. Hence, inferring saliency information from RGB-D inputs (D refers to the depth image) has attracted many research attention in the SOD area.

Traditional RGB-D detectors [19]–[21] mainly examined the hand-crafted priors, which degrades the detection performance, since the assumptions of these heuristic priors are not always correct. Later, RGB-D SOD detectors [18], [22]–[27] based on convolutional neural networks (CNNs) have been developed by learning the features from RGB and depth modalities and exploring the complementary information between them. These methods can befurther grouped as two main categories: one-stream detectors and two-stream detectors, according to the number of network streams for a multi-model fusion. Although achieving high accuracy on the benchmark datasets, these CNN-based detectors mainly relied on labeled training data to detect salient objects from a pair of RGB-D inputs, thereby tending to produce poor results in some complex situations. The reason behind is that the labeled data are collected in the limited applications and thus CNNs trained on only these labeled data suffer from limited capabilities to handle unseen photos. Compared with the labeled datasets, we could easily collect abundant unlabeled RGB-D images from diverse scenarios. Hence, it is highly desirable to leverage additional unlabeled data and the limited labeled data to improve the performance of RGB-D saliency detection.

In this work, we present a self-supervised self-ensembling network (S ^{3} Net) for boosting the RGB-D saliency detection by learning from both labeled and unlabeled data.

Manuscript received 2 April 2021; revised 21 August 2021 and 3 October 2021; accepted 30 October 2021. Date of publication 22 November 2021; date of current version 9 March 2023. This work was supported in part by the National Natural Science Foundation of China under Grants 61902275, 61802453, and 61802072, in part by Hong Kong Polytechnic University under Grants P0030419, P0030929, and P0035358, in part by the National Key Research and Development Program of China under Grant 2019YFB1703600, and in part by Hong Kong Metropolitan University under Research Grant 2020/1.12. The Associate Editor coordinating the review of this manuscript and approving it for publication was Prof. Houqiang Li. *(Corresponding author: Qing Zhang.)*

Lei Zhu is with the Hong Kong University of Science and Technology, and the Department of Applied Mathematics and Theoretical Physics (DAMTP), University of Cambridge, Cambridge CB3 0WA, U.K. (e-mail: zhulei9009@gmail.com).

Xiaoqiang Wang is with the College of Computer Science and Technology, Zhejiang University, Shatin 310058, China (e-mail: xq.wang@zju.edu.cn).

Ping Li is with the Department of Computing, Hong Kong Polytechnic University, Kowloon, Hong Kong SAR 00852, China (e-mail: p.li@polyu.edu.hk).

Xin Yang is with the Department of Computer Science, Dalian University of Technology, Dalian 116024, China (e-mail: xinyang@dlut.edu.cn).

Qing Zhang is with the School of Data and Computer Science, Sun Yat-Sen University, Guangzhou 510006, China (e-mail: zhangqing.whu.cs@gmail.com).

Weiming Wang is with the School of Science and Technology, Hong Kong Metropolitan University, Ho Man Tin, Hong Kong SAR 00852, China (e-mail: wmwang@ouhk.edu.hk).

Carola-Bibiane Schönlieb is with the Department of Applied Mathematics and Theoretical Physics (DAMTP), University of Cambridge, Cambridge CB30WA, U.K. (e-mail: cbs31@cam.ac.uk).

C. L. Philip Chen is with the School of Computer Science and Engineering, South China University of Technology, Guangzhou 510006, China, and with Navigation College, Dalian Maritime University, Dalian 116026, China, and also with the Faculty of Science, and Technology, University of Macau, Macau 999078, China (e-mail: philip.chen@ieee.org).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TMM.2021.3129730.

Digital Object Identifier 10.1109/TMM.2021.3129730

Specifically, we first devise a self-guided convolutional neural network (SG-CNN) to integrate the self-supervised learning mechanism into a multi-task learning framework for detecting salient objects from RGB-D data. In SG-CNN, we develop a set of three-layer cross-model feature fusion (TCF) modules to fuse CNN features from depth and RGB modalities, and predict both saliency map and an image rotation angle. After that, we take the developed SG-CNN as both student network and teacher network, present a self-supervised two-task supervised loss for labeled data, which is computed as the summation of the losses on saliency prediction and image rotation angle prediction. Further, we enforce the two tasks' results of the student network and teacher network to be consistent, respectively, on all the unlabeled data. By adding the supervised loss from the developed SG-CNN and the consistency loss from two tasks to train the model, our network can more accurately detect salient objects than the state-of-the-art methods. We summarize our contributions as follows:

- First, we develop a self-supervised self-ensembling network (S ^{3} Net) for RGB-D saliency detection by simultaneously exploring the unlabeled data and the self-supervision mechanism. As a self-ensembling model, our framework has the potential to be used for developing semi-supervised frameworks on other vision tasks, including shadow detection, boundary detection, and semantic segmentation.
- Second, we design a self-guided convolutional neural network (SG-CNN) with a series of three-level cross-modal fusion (TCF) modules for a multi-model fusion, as well as a joint prediction of a saliency map and an additional image rotation angle. Our SG-CNN with these two tasks achieves better results than only predicting the saliency maps even using only the labeled data.
- Lastly, we show that the developed S ^{3} Net outperforms state-of-the-art RGB-D saliency detection methods on seven widely-used benchmark datasets.

II. Related Work

In this section, we present a detail review on RGB-D saliency detection methods, self-ensembling methods, and self-supervised learning methods.

Traditional methods: Early attempts [19], [21], [28]–[30] inferred salient objects by exploring hand-crafted features or priors, including center-surround difference [29], [31], contrast [32], background enclosure [19], center/boundary prior [33]–[35], or a combination of various saliency measures [20]. Unfortunately, most of them work well only on high-quality and well-constrained images.

Deep-learning-based methods: Motivated by outstanding performance of convolutional neural networks (CNNs) in diverse vision tasks, many CNN-based RGB-D saliency detectors have been proposed by leveraging the complementary information between them. We can further divide these CNN-based detectors into two main categories: one-stream CNN and two-stream CNN. *One-stream CNNs* combined the input RGB and depth images as a four-channel input, and fed the four-channel input into a CNN for saliency detection. Liu *et al.* [38] and Huang *et al.* [39] passed the four-channel input from RGB-D images into a single-stream recurrent convolutional neural network and a FCN with short connections, respectively.

Two-stream CNNs: usually designed a two-stream network architecture to learn features from RGB and depth inputs independently and fused these features from two modalities for saliency inference. Chen *et al.* [22] predicted a saliency map by extracting CNN features from the RGB and depth images respectively and develop a complementarity-aware fusion (CA-Fuse) module to fuse CNN features from the input two modalities for predicting a saliency map. Han *et al.* [40] transferred the structure of the RGB-based CNN to be applicable for depth view and fused the deep representations of both views automatically to obtain the final saliency map. Chen *et al.* [41] presented a three-stream multi-model fusion network for RGB-D saliency detection. Apart from two separate streams to learn features from the RGB and depth views, a cross-modal distillation stream is introduced to extract new RGB-D features in each CNN level, and a channel-wise attention mechanism is presented to adaptively select complementary feature maps from each modality in each CNN level for a cross-level feature fusion. Later, Chen *et al.* [42] developed a new multi-scale multi-path fusion network with cross-modal interactions. Zhao *et al.* [23] enhanced the depth information by integrating contrast priors into a CNN-based architecture and then fused the enhanced depth cues with RGB features for saliency detection. Piao *et al.* [24] proposed a depth distiller to utilize the network prediction and attention as two bridges to transfer the depth knowledge from the depth stream to the RGB stream for detecting saliency information.

In [17], Fan *et al.* collected a RGB-D dataset and built a depth-depurator network to judge whether a depth map should be concatenated with the RGB image to formulate an input signal. Instead of conducting independent feature extraction from RGB and depth views, Fu *et al.* [25] achieved RGB-D saliency detection by developing a network with two modules: joint learning (JL) and densely-cooperative fusion (DCF). The JL module simultaneously learned saliency features from RGB and depth inputs via a shared Siamese network while DCF module was introduced for complementary feature discovery. Zhang *et al.* [26] selected useful feature representation from the RGB and depth data, and effectively integrated cross-modal features for accurately locating salient objects with fine edge details. Liu *et al.* [27] fused the self-attention and the other modality's attention in the non-local model for fusing multi-modal information for RGB-D saliency detection. Zhang *et al.* [43] designed an asymmetric CNN to fuse RGB and depth information by a depth attention mechanism to locate salient objects. Pang *et al.* [44] combined RGB and depth features to generate multi-scale convolution kernels to guide the decoding process of RGB stream. Zhao *et al.* [45] utilized the depth map to guide feature fusion of RGB and depth modalities.

Apart from concentrating on cross-module fusion between depth and RGB modalities, other CNNs have been proposed to detecting salient objects from RGB-D data from new perspectives. Chen *et al.* [46] fed RGB and depth features to guided residual blocks to progressively refine the saliency prediction.

Ji *et al.* [47] adopted a collaborative learning framework to leverage edge, depth and saliency results for RGB-D saliency detection.

Luo *et al.* [48] modeled the mutual relations of RGB and depth modalities over a set of cascade graphs.

Although higher accuracy were achieved comparing with the traditional methods, the existing CNN-based RGB-D saliency detectors almost suffered from a common limitation that training their networks requires a large amount of data with pixel-level annotations. In this paper, we leverage unlabelled data and embed a self-supervised learning into a self-ensembling framework to boost RGB-D saliency detection. Experimental results on seven benchmark datasets show that our method outperforms state-of-the-art RGB-D saliency detectors, as shown in the experiment section.

Self-ensembling methods: As a semi-supervised method, self-ensembling techniques usually devise a consistency loss on the unlabeled data, thereby guaranteeing invariant predictions for perturbations of unlabeled data. For example, Π-model [49] proposed to utilize consistency constraints between the output of the current network and the temporal average of network outputs during the network training process. The mean teacher (MT) framework [36] averaged the network parameters to replace the network prediction output average in Π-model [49] and has achieved improved performance in the semi-supervised learning. Motivated by the success of self-ensembling methods in many vision tasks [37], [50], this work presents the first semi-supervised RGB-D saliency detection method by taking the self-ensembling as the basic network to include the unlabeled data into the training set. More importantly, we seamlessly integrate the self-supervised learning strategy into a two-task self-ensembling framework, where an auxiliary image rotation angle is predicted for enhancing RGB-D saliency detection, but any other supervision is not required.

Semi-supervised saliency detectors: Apart from labeled data, several SOD methods leveraged additional unlabeled data to gain more understanding of salient information in general. Zhao *et al.* [51] presented a semi-supervised low-rank optimization to use label information to construct a graph and then propagate the label information to unlabeled data for image classification and saliency inference. Zhou *et al.* [52] designed a semi-supervised saliency classifiers to utilize a linear feedback control system for establishing a relationship between control states and salient object detection. Zhang *et al.* [53] detected saliency regions without any human annotation by a synthesized mechanism to generate supervisory signals. Zhang *et al.* [54] learned an effective salient object detection model based on the manual annotation on a few training images only via an adversarial-paced learning (APL)-based framework. Apparently, these semi-supervised learning methods addressed the task of detecting and segmenting salient objects from a single input image. However, our semi-supervised learning network is designed for salient object detection by integrating paired labeled RGB-D data and unlabeled RGB-D data.

Self-supervised learning: Self-supervised learning aims to learn representations that are useful for solving real-world downstream tasks by automatically creating self-supervised pretext tasks [55], [56]. Gidaris *et al.* [57] randomly rotated an image by one of four possible angles and then formulated the pretext task to predict such rotation angle. Caron *et al.* [58] utilized the clustering of the training images to create image labels for predictions. Another group of the pretext task aims to generate dense pixel-wise outputs, including image inpainting [59], image colorization [60], motion segmentation prediction [61], and so on. Motivated by the prediction accuracy gain of these networks, we also first integrate the self-supervised mechanism into a semi-supervised learning framework to boost RGB-D saliency detection, which has been proved our superior saliency detection performance on benchmark datasets. Moreover, to reduce the time consuming of the network training, we take simultaneously minimize the loss functions of both the pretext task and the underlying saliency detection to train our network for RGB-D saliency detection.

III. Methodology

To explore the knowledge from both labeled and unlabeled data, we formulate a self-supervised self-ensembling network (S ^{3} Net), as shown in Fig. 1. Specifically, we first develop a self-guided convolutional neural network (SG-CNN) for RGB-D saliency detection by taking the RGB-D images as the inputs and predicting the saliency maps as well as the rotation angle of the inputs. Note that the supervision of the rotation angle can be directly obtained from the input images without any manual labels, which is used as the self-supervised learning. After that, we assign this SG-CNN as the student and teacher networks of the overall self-ensembling learning framework. During the training, the labeled data is fed into the student network, which is trained by fusing a saliency detection loss and a self-supervised (rotation angle) loss. Then, for unlabeled data, we feed it into the student network and teacher network, respectively. A self-supervised consistency loss is computed on both saliency prediction and rotation angle prediction. In the testing stage, we only adopt the student network to predict the saliency detection maps for the input RGB-D images.

A. Self-Guided Convolutional Neural Network (SG-CNN)

Fig. 2 overviews the schematic illustration of the developed self-guided convolutional neural network (SG-CNN), which explores the knowledge from RGB image as well as the depth image and adopts the multi-task learning strategy to train the network, i.e., jointly predicting the saliency map and rotation angle. Given the RGB-D images, our SG-CNN first utilizes a convolutional neural network (CNN) to extract five feature maps (i.e., r_1 to r_5) with different spatial resolutions from the RGB image and another CNN (i.e., d_1 to d_5) to learn features at five CNN layers from the depth image. To combine the complementary information from the depth map, we then develop a three-layer cross-model feature fusion (TCF) module at each CNN layer to leverage both RGB and depth views by fusing features at two adjacent CNN layers, resulting in five integrated features (i.e., f_1 to f_5). As shown in Fig. 3, TCF at the i-th CNN layer takes three pairs of RGB and depth features (i.e., r_{i-1} and d_{i-1} at $(i-1)$-th CNN layer, r_i and d_i at i-th CNN layer, and r_{i+1} and

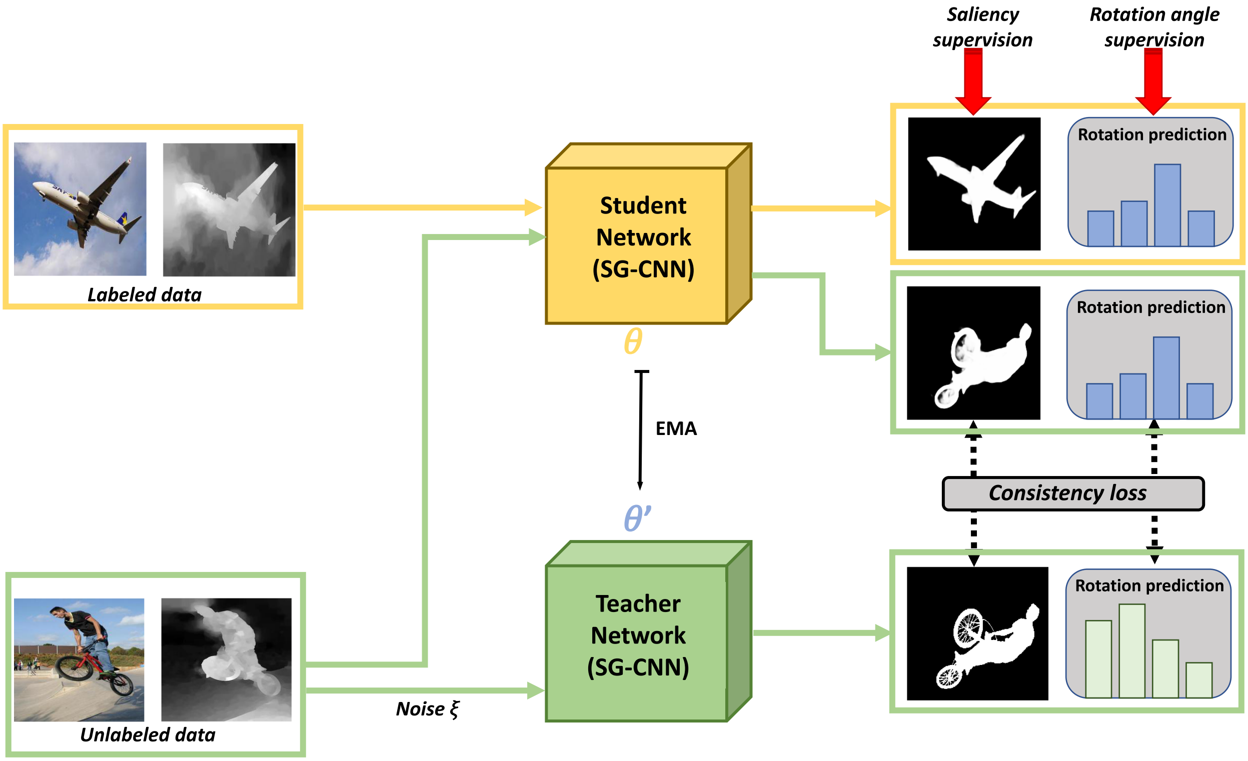


Fig. 1. The schematic illustration of our self-supervised self-ensembling framework (S ^{3} Net). We first develop a self-guided CNN (SG-CNN; see Fig. 2) to learn an additional image rotation angle for saliency detection. After that, we compute a supervised loss for labeled data and a consistency loss for unlabeled data. Finally, we fuse the supervised loss and consistency loss to train our S ^{3} Net for RGB-D saliency detection. EMA: exponential moving average; see [36], [37]. For unlabeled data, we produce an auxiliary image by employing a color jitter operation (ξ) (four parameters: brightness=0.1, contrast=0.1, saturation=0.1, and hue=0.0) on the input unlabeled data to change its brightness, contrast, and saturation.

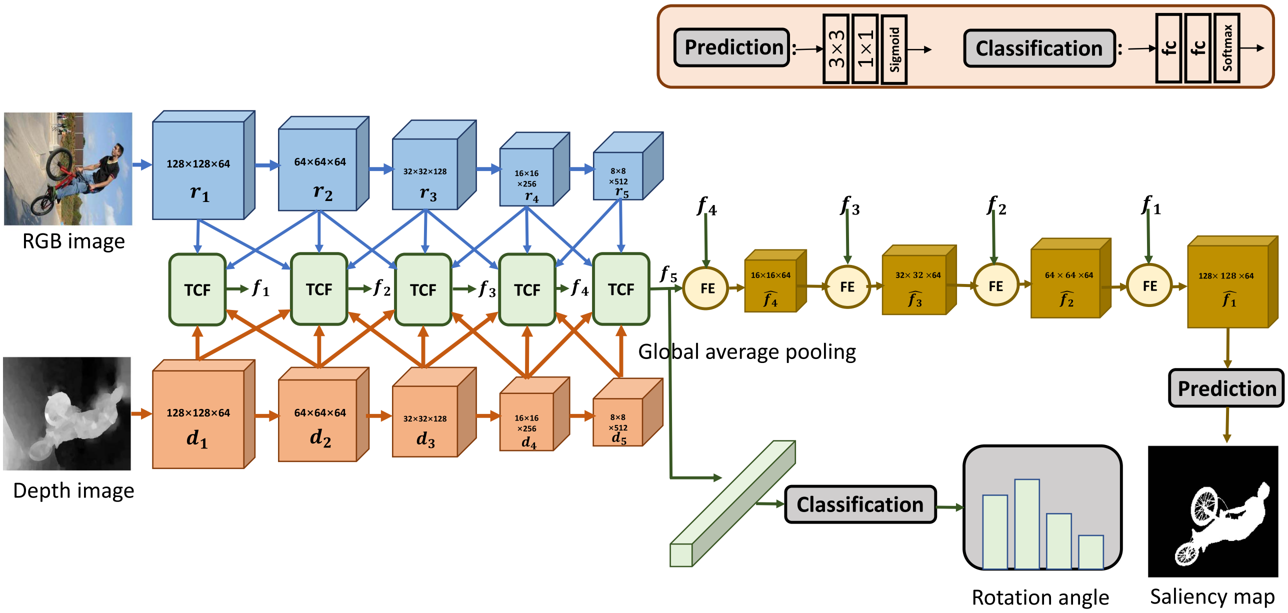


Fig. 2. The schematic illustration of the proposed SG-CNN in Fig. 1. SG-CNN takes a pair of RGB and depth images as the input and predicts a saliency map and an image rotation angle. We first obtain five CNN features (d_1 to d_5; r_1 to r_5) from the depth image and the RGB image, respectively. Then, we develop a series of three-layer cross-modal feature integration (TCF) modules to fuse RGB and depth features, and design two branches to predict an image rotation angle and a saliency map from the output features (f_1 to f_5) of these five TCF modules.

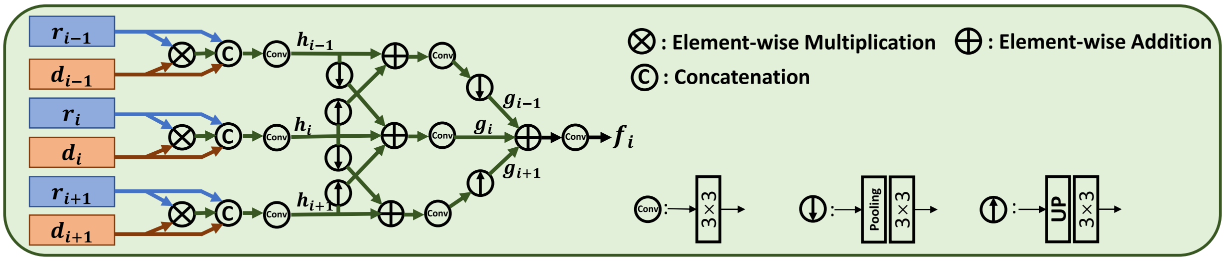


Fig. 3. The schematic illustration of the proposed three-layer cross-model feature fusion (TCF) module (see Fig. 2).

d_{i+1} at $(i+1)$-th CNN layer) at three adjacent CNN layers as the input and outputs an aggregated feature map (f_i). we multiple each pair of RGB and depth features, concatenate the result with the original RGB and depth features, and apply a 3 × 3 convolutional layer on the concatenated feature map. After that, we can obtain three new features; see h_{i-1}, h_i, and h_{i+1} in Fig. 3. Mathematically, h_{i-1}, h_i, and h_{i+1} are computed as:

$$h_{i-1} = W_1 * Cat(r_{i-1}, r_{i-1} \bigotimes d_{i-1}, d_{i-1}) + b_1 \tag{1}$$

$$h_i = W_2 * Cat(r_i, r_i \bigotimes d_i, d_i) + b_2 \tag{2}$$

$$h_{i+1} = W_3 * Cat(r_{i+1}, r_{i+1} \bigotimes d_{i+1}, d_{i+1}) + b_3 \tag{3}$$

where $Cat()$ is used to concatenate feature maps. (W_1, b_1), (W_2, b_2), and (W_3, b_3) are the weights and bias of the three 3 × 3 convolutional layers on the concatenated features. Once obtaining h_{i-1}, h_i, and h_{i+1}, we upsample h_i to the same spatial resolution of h_{i-1}, element-wisely add it with h_{i-1}, apply a 3 × 3 convolutional layer on the addition result, and then downsample the obtained features into the spatial resolution of h_i to produce new features g_{i-1}:

$$g_{i-1} = down(Conv(up(h_i) + h_{i-1})) \tag{4}$$

where up and $down$ denote the feature upsampling and downsampling operations, respectively. $Conv$ represents a 3 × 3 convolutional operation. Second, we downsample h_{i-1} and upsample h_{i+1} to the same resolution of h_i, add these two resized feature maps with h_i, and apply a 3 × 3 convolutional layer on the addition result to produce a new feature g_i:

$$g_i = Conv(up(h_{i+1}) + h_i + down(h_{i-1})) \tag{5}$$

where $up(h_{i+1})$ and $down(h_i)$ denote the upsampled features of h_{i+1} and the downsampled features of h_i. and both $up(h_{i+1})$ and $down(h_i)$ have the same resolution of h_i. Third, we downsample h_i to the resolution of h_{i+1}, element-wisely add it with h_{i+1}, apply a 3 × 3 convolutional layer on the addition result, and then upsample the obtained features into the spatial resolution of h_i to produce new features g_{i+1}:

$$g_{i+1} = up(Conv(down(h_i) + h_{i+1})) \tag{6}$$

where $down(h_i)$ denotes the downsampled features of h_i, and it has the same resolution of h_{i+1}.

As shown in Fig. 2, we can obtain five features (f_1 to f_5; see Fig. 2) by applying TCF modules to integrate RGB and depth features at each CNN layer, and use them for predicting the rotation angle and the saliency map. In the first branch, we predict a rotation angle of the input RGB-D images from f_5. To be specific, we first apply an average pooling operation on f_5 to obtain a new feature map, which is then passed into two fully connected layers and a softmax layer for obtaining a vector with four elements, *i.e.*, $\Omega = \{0^o, 90^o, 180^o$, and $270^o\}$, to represent the rotation angle.

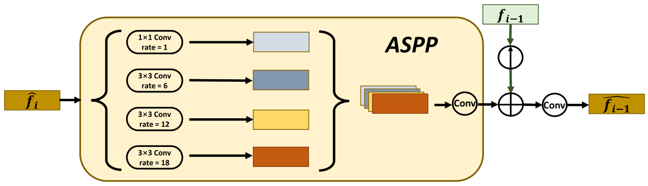


Fig. 4. The schematic illustration of our feature enhancement (FE) module (see Fig. 2).

Our second branch fuses these integrated features (i.e., f_1 to f_5) to predict an output saliency map. To achieve this, we formulate a feature enhancement (FE) module to combine features at two adjacent convolutional layers and iteratively perform the feature combination from deep to shallow convolutional layers. Specifically, we pass f_5 and f_4 to a FE module and obtain a new feature map (denoted as $\widehat{f_4}$), which is then fused with f_3 by using the second FE module, resulting in a feature map $\widehat{f_3}$. Then, the third FE module is applied to combine $\widehat{f_3}$ and f_2 for obtaining $\widehat{f_2}$, and we further pass $\widehat{f_2}$ and f_1 to the fourth FE module. From the the output features $\widehat{f_1}$ of the fourth FE module, we predict a saliency map by applying a 3 × 3 convolutional layer, a 1 × 1 convolutional layer, and a sigmoid activation function, and take this saliency map as the final output of our SG-CNN.

Fig. 4 shows the overflow of the FE module at the i-th CNN layer. Given the integrated features f_{i-1} at $(i-1)$-th CNN layer and the output features $\widehat{f_i}$ of at the $(i-1)$-th FE module, the FE module first upsamples f_{i-1} to the resolution of $\widehat{f_i}$, and passes $\widehat{f_i}$ to an atrous spatial pyramid pooling (ASPP) block [62] (four dilated rates: 1, 6, 12, and 18). After that, we element-wisely add the upsampled f_{i-1} and the resultant features of the ASPP block and feed the addition result to a 3 × 3 convolutional layer to obtain the output features (denoted as $\widehat{f_{i-1}}$) of the FE module.

B. Supervised Loss on Labeled Data

For labeled data, we have a pair of input RGB and depth images and the corresponding annotated saliency mask. It is natural that we take the annotated saliency mask as the ground truth (G_s) for RGB-D saliency detection. On the other hand, we randomly select an angle from the set $\{0^o, 90^o, 180^o$, and $270^o,\}$ for each image of the training set (including labeled data and unlabeled data), rotate it accordingly, and feed it into our SG-CNN (see Fig. 2). Then, we take the selected rotation angle as the ground truth (G_a) of the rotation angle classification task.

With these two ground truths ((G_s) & (G_a)), the supervised loss (denoted as $\mathcal{L}^s$) for a labeled image (x) is computed as the summation of the saliency detection loss and rotation angle prediction loss, i.e.,

$$\mathcal{L}^s(x) = \Phi_{BCE}\left(P_s, G_s\right) + \alpha\ \Phi_{CE}\left(P_a, G_a\right) \tag{7}$$

where P_s and P_a denote the predicted saliency map and rotation angle, respectively. Φ_{BCE} and Φ_{CE} are the binary cross-entropy loss and cross-entropy loss functions, respectively. We empirically set the weight $\alpha = 0.1$ during the network training.

C. Unsupervised Loss on Unlabeled Data

For the unlabeled data, we pass it into the student network and teacher network to obtain two groups of prediction results, where each prediction consists of a saliency prediction map and a rotation angle. We then enforce the predictions of the student network and teacher network to be consistent, resulting in an unsupervised loss ($\mathcal{L}^u$) on unlabeled data. Mathematically, $\mathcal{L}^u$ for an unlabeled image (denoted as y) is defined as

$$\mathcal{L}^u(y) = \Phi_{MSE}\left(S_s, T_s\right) + \varphi\ \Phi_{KL}\left(S_a, T_a\right) \tag{8}$$

where S_s and T_s denote the saliency predictions from student network and teacher network, respectively; S_a and T_a are the rotation angle predictions from the student network and the teacher network. Φ_{MSE} and Φ_{KL} are the MSE loss and KL divergence loss, respectively. We empirically set the balancing weight $\varphi = 1$.

D. Training Strategies

We apply the self-guided multi-task learning with the self-ensembling model for RGB-D saliency detection. The total loss of our network is computed as:

$$\mathcal{L}_{total} = \sum_{i=1}^{N_1} \mathcal{L}^s(x_i) + \lambda \sum_{j=1}^{N_2} \mathcal{L}^u(y_j) \tag{9}$$

where N_1 and N_2 are the numbers of labeled images and unlabeled images in our training set. $\mathcal{L}^s(x_i)$ denotes the supervised loss (Eq. (7)) for the i-th labeled image while $\mathcal{L}^u(y_j)$ is the unsupervised loss (Eq. (8)) for the j-th unlabeled image. The weight λ is to balance $\mathcal{L}^s(x_i)$ and $\mathcal{L}^u(y_j)$. Following [36], [37], we use a time dependent Gaussian warming up function to update λ: $\lambda(t) = \lambda_{\max} e^{(-5(1-t/t_{\max})^2)}$, where t denotes the current epoch number and $t_{\max}$ is the maximum epoch number in the training process. In our experiments, we empirically set $\lambda_{\max} = 1$.

Exponential moving average (EMA): Following existing self-ensembling frameworks [36], [37], we minimize the total loss $\mathcal{L}_{total}$ of Eq. (9) to train the student network, and the parameters of the teacher network is computed as the exponential moving average (EMA) of the parameters of the student network to ensemble the information in different training steps. The parameters of the teacher network at the t training iteration are defined as:

$$\theta'_t = \mu\theta'_{t-1} + (1-\mu)\theta_t \tag{10}$$

where θ_t denotes the student network parameter at the t-th training iteration while θ'_{t-1} denotes the teacher network parameter at the $(t-1)$-th training iteration. The EMA decay $\mu = 0.99$ as indicated in [36].

Our unlabeled data: Note that Song *et al.* [63] collected a SUN-RGBD benchmark dataset for RGB-D scene understanding. The training set in SUN-RGBD contains 5,285 pairs of RGB and depth images. In this regard, we empirically use all these 5,285 pairs of SUN-RGBD as the unlabeled data in our work. Apparently, all these unlabeled data do not contain any annotations of saliency maps.

E. Difference Between Our Method and MTMT [37]

MTMT [37] and our work are different in three aspects. (1) We first admit that both two works exploited the multi-task learning and semi-supervised learning with unlabeled data, but our work is designed for saliency detection from RGB-D paired data while MTMT [37] is to detect shadows from single image. (2) The multi-task learning is different in two works. MTMT [37] jointly detected shadow regions, shadow edges, and the number of shadow regions, but our work simultaneously identified saliency regions and predicted an image rotation angle. (3) The auxiliary tasks in MTMT [37] are with a supervised learning mechanism while the additional image rotation angle prediction is learning in a self-supervised learning manner.

IV. Experimental Results

In this section, we will introduce benchmark datasets and evaluation metrics, as well as present experiments to verify our S^3-Net. Our code, the trained models, and the predicted saliency maps on all seven benchmark datasets are released at: https://github.com/Robert-xiaoqiang/S3Net.

Training Patameters: We adopt ResNet50 [68] (pre-trained on ImageNet [69]) as the feature extraction backbone of our network. Training data is resized to 256 × 256 and augmented by a random rotation and horizontal flipping for training. Color jittering is employed as the perturbation noise of unlabeled data (see Noise ξ of Fig. 2). We use a stochastic gradient descent (SGD) optimizer with the batch size 8 (*i.e.*, 4 labeled data and 4 unlabeled data). The epoch number, momentum, and decay rate are empirically set as 50, 0.9, and 0.0005, respectively. The learning rate is adjusted by a poly strategy with an initial learning rate of 0.001 and the power of 0.9.

A. Datasets and Evaluation Metrics

Benchmark Datasets: We conducted comparisons on seven widely-used benchmark datasets in our experiments. They are (i) NJU2K [31] (2,000 images), (ii) NLPR [32] (1,000 images), (iii) STERE [28] (1,000 images), (iv) RGBD135 [33] (135 images), (v) LFSD [70] (100 images), (vi) SIP [17] (929 images), and (vii) DUT-RGBD [24] (1,200 images captured by Lytro camera in real life scenes). We followed the same settings of existing works [24] to use 800 images for the network training and remaining 400 images were utilized for testing different methods to obtain their results of DUT-RGBD. Moreover, following recent works [17], [23], [25], we utilized a same training set consisting of 700 images from NLPR and 1,500 images from NJU2K to train our network and competitors for obtaining results of other six benchmark datasets for fair comparisons.

Evaluation metrics: We adopt four widely-used metrics to quantitatively compare RGB-D saliency detection performance of different approaches. They are S-measure (denoted as S_m), F-measure (denoted as $F_\beta^{\max}$) [71], [72], and E-measure (denoted as $E_\phi^{\max}$), and mean absolute error (denoted as MAE). Overall, a better RGB-D saliency detector shall have a larger S_m, a larger $F_\beta^{\max}$, a larger $E_\phi^{\max}$, and a smaller MAE.

S-measure [73] (S_m) computes the similarity of $\mathcal{D}$ and $\mathcal{G}$ by considering its object-aware and region-aware structural similarities:

$$S_m = \rho\, S_o(\mathcal{D}, \mathcal{G}) + (1-\rho)\, S_r(\mathcal{D}, \mathcal{G}) \tag{11}$$

where $S_o(\mathcal{D}, \mathcal{G})$ and $S_r(\mathcal{D}, \mathcal{G})$ denote the object-aware and region-aware structural similarities; respectively. Please refer to [73] for their definitions. $\rho = 0.5$, as suggested in [73].

F-measure ($F_\beta^{\max}$) [71], [72] is to balance the average precision and average recall over saliency maps of all images in the dataset for evaluation. Its definition is given by:

$$F_\beta^{\max} = \frac{(1+\beta^2) \times Precision \times Recall}{\beta^2 \times Precision + Recall}, \tag{12}$$

where $\beta^2 = 0.3$; see [74], [75]. Instead of plotting the whole F-measure, we follow existing RGB-D saliency detectors [72], [75], [76] to directly use the maximal $F_\beta^{\max}$ for comparisons.

E-measure ($E_\phi^{\max}$) [77] quantitatively compares $\mathcal{D}$ and $\mathcal{G}$ by simultaneously considering the global means of the image and the local pixel matching:

$$E_\phi^{\max} = \frac{1}{W_{\mathcal{D}} \times H_{\mathcal{D}}} \sum_{p=1}^{W_{\mathcal{D}}} \sum_{q=1}^{H_{\mathcal{D}}} \mathbf{A}(p,q) \tag{13}$$

where $W_{\mathcal{D}}$ and $H_{\mathcal{D}}$ are the width and height of $\mathcal{D}$. $\mathbf{A}$ denotes the enhanced alignment matrix, which represents the correlation between $\mathcal{D}$ and $\mathcal{G}$; please see [77] for the details of computing $\mathbf{A}$.

MAE [78] averages the pixel-wise absolute difference between $\mathcal{D}$ and $\mathcal{G}$:

$$MAE = \frac{1}{W_{\mathcal{D}} \times H_{\mathcal{D}}} \sum_{p=1}^{W_{\mathcal{D}}} \sum_{q=1}^{H_{\mathcal{D}}} \|\mathcal{D}(p,q) - \mathcal{G}(p,q)\|, \tag{14}$$

where $\mathcal{D}(p,q)$ and $\mathcal{G}(p,q)$ denote the value at the pixel (p, q) of the predicted saliency map $\mathcal{D}$ and the ground truth $\mathcal{G}$; respectively.

B. Comparison With the State-of-The-Arts

We evaluate the effectiveness of our network by comparing it against 20 state-of-the-art RGB-D salient object detectors. They are LBE [19], DF [64], CTMF [40], PCF [22], TANet [41], CPFP [23], DMRA [24], D ^{3} Net [17], SSF [26], UCNet [18], JLDCF [25], HDF-Net [44], ATSA [43], PGA-Net [46], DANet [45], cmMS [65], Cas-Gnn [48], CMWNet [66], CoNet [47], and BBS-Net [67]. Among them, LBE [19] focused on hand-crafted features, while other 19 methods relied on convolutional neural networks (CNNs) to learn deep features for RGB-D saliency detection. To make the comparisons fair, we obtained the saliency maps of all 20 competitors either from the authors or by using their implementations with the released training models and parameters.

Quantitative comparisons: Table I reports the S_m, $F_\beta^{\max}$, $E_\phi^{\max}$, and MAE scores of our method and all competitors on seven benchmark datasets. From these quantitative results, we can find that our S^3-Net produces a superior metric performance over other saliency detectors on almost all seven datasets. It indicates that our network can more accurately detect salient objects from RGB-D data than compared methods. Specifically, our method has largest S_m, $F_\beta^{\max}$, and $E_\phi^{\max}$ scores and smallest MAE scores on STERE, LFSD, and DUT-RGBD. Moreover, we has the best $F_\beta^{\max}$ and MAE results and the the second best results of S_m and $E_\phi^{\max}$ for NJU2K. For NLPR, our method takes the first places of $F_\beta^{\max}$, $E_\phi^{\max}$, and MAE scores, as well as second place of S_m scores. Regarding the remaining two datasets (RGBD135 and SIP), our $E_\phi^{\max}$ and MAE results rank first, while S_m, $F_\beta^{\max}$ results of our method are top five; see Table I.

Visual comparisons: Fig. 5 visually compares saliency maps produced by our network and state-of-the-art RGB-D saliency detectors. Apparently, other methods in Fig. 5(e)-(m) tend to include non-salient backgrounds or lose salient details in their predicted saliency maps, while our S^3-Net produces more accurate saliency maps (d), which are more consistent with the ground truths (c). It indicates that exploring unlabeled data and self-supervised multi-task learning in our network is capable to suppress non-salient objects and detect more salient pixels than the compared RGB-D saliency detectors, which are mainly trained in a supervised learning manner.

C. Ablation Analysis

Baseline network design: We perform ablation study experiments to evaluate the effectiveness of different components in our S^3Net, s.t., TCF module, image rotation angle prediction, and self-supervised learning on unlabeled data. Here, we consider six baseline networks and evaluate them on seven benchmark datasets.

The first three baseline networks are constructed by removing the teacher model and the unlabeled data. It means that only supervised loss on labeled data is used to train SG-CNN, and we directly use the SG-CNN with labeled data use to predict the RGB-D saliency map. Specifically, we first construct a baseline network (denoted as "basic") that removes the rotation angle classification branch from SG-CNN. The second (denoted as "basic-TCF") network replaces the TCF module of "basic" with a simple element-wise addition on all six input features of the TCF module for integrating CNN features from the RGB image

TABLE I
QUANTITATIVE COMPARISONS BETWEEN OUR NETWORK AND STATE-OF-THE-ART DETECTORS ON SEVEN BENCHMARK DATASETS. THE BEST RESULTS ARE HIGHLIGHTED IN **BOLD**

Dataset	Metric	LBE [19]	DF [64]	CTMF [40]	PCF [22]	TANet [41]	CPFP [23]	DMRA [24]	D^{3}Net [17]	SSF [26]	UCNet [18]	JLDCF [25]	HDF-Net [44]	PGA-Net [46]	DANet [45]	cmMS [65]	Cas-Gnn [48]	CMWNet [66]	ATSA [43]	BBS-Net [67]	CoNet [47]	Our method
NJU2K	$S_m \uparrow$	0.695	0.763	0.849	0.877	0.878	0.879	0.886	0.895	0.899	0.897	0.903	0.908	0.906	0.901	0.904	0.911	0.903	0.899	**0.921**	0.894	0.913
	$F_{\beta}^{max} \uparrow$	0.748	0.804	0.845	0.872	0.874	0.877	0.886	0.889	0.886	0.886	0.903	0.922	0.883	0.893	0.914	0.903	0.902	0.910	0.920	0.872	**0.928**
	$E_{\phi}^{max} \uparrow$	0.803	0.864	0.913	0.924	0.925	0.926	0.927	0.932	-	0.930	0.944	0.932	0.914	0.921	-	0.936	0.933	0.922	**0.949**	0.912	0.944
	$MAE \downarrow$	0.153	0.141	0.085	0.059	0.060	0.053	0.051	0.051	0.043	0.043	0.043	0.038	0.045	0.040	0.044	0.035	0.046	0.045	0.035	0.047	**0.034**
NLPR	$S_m \uparrow$	0.762	0.802	0.860	0.874	0.886	0.888	0.899	0.905	0.914	0.920	0.925	0.923	0.918	0.907	0.900	0.919	0.917	0.915	**0.930**	0.907	0.927
	$F_{\beta}^{max} \uparrow$	0.745	0.778	0.825	0.841	0.863	0.867	0.879	0.885	0.875	0.891	0.916	0.927	0.871	0.876	0.914	0.904	0.903	0.916	0.918	0.848	**0.923**
	$E_{\phi}^{max} \uparrow$	0.855	0.880	0.929	0.925	0.941	0.932	0.947	0.946	-	0.951	0.962	0.957	0.948	0.945	-	0.952	0.951	0.949	0.961	0.936	**0.962**
	$MAE \downarrow$	0.081	0.085	0.056	0.044	0.041	0.036	0.031	0.034	0.026	0.025	0.023	0.023	0.028	0.028	0.273	0.025	0.029	0.028	0.023	0.031	**0.021**
STERE	$S_m \uparrow$	0.660	0.757	0.848	0.875	0.871	0.879	0.886	0.891	0.893	0.903	0.905	0.900	0.897	-	0.889	0.899	0.905	0.903	0.908	0.908	**0.913**
	$F_{\beta}^{max} \uparrow$	0.633	0.757	0.831	0.860	0.861	0.874	0.886	0.881	0.880	0.884	0.901	0.910	0.884	-	0.908	0.901	0.901	0.872	0.903	0.885	**0.918**
	$E_{\phi}^{max} \uparrow$	0.787	0.847	0.912	0.925	0.923	0.925	0.937	0.930	-	0.935	0.936	0.931	0.921	-	-	0.930	0.934	0.914	0.932	0.923	**0.945**
	$MAE \downarrow$	0.250	0.141	0.086	0.064	0.060	0.051	0.047	0.054	0.044	0.039	0.042	0.041	0.039	-	0.042	0.039	0.043	0.044	0.041	0.041	**0.038**
RGBD135	$S_m \uparrow$	0.703	0.752	0.863	0.842	0.858	0.872	0.900	0.904	0.905	**0.934**	0.929	0.926	0.894	0.907	-	0.905	**0.934**	0.924	0.933	0.910	0.932
	$F_{\beta}^{max} \uparrow$	0.788	0.766	0.844	0.804	0.827	0.846	0.888	0.885	0.876	0.919	0.919	**0.932**	0.870	0.885	-	0.906	0.930	0.928	0.927	0.861	0.925
	$E_{\phi}^{max} \uparrow$	0.890	0.870	0.932	0.893	0.910	0.923	0.943	0.946	-	0.967	0.968	**0.971**	0.935	0.952	-	0.947	0.969	0.968	0.966	0.945	**0.971**
	$MAE \downarrow$	0.208	0.093	0.055	0.049	0.041	0.038	0.030	0.030	0.025	0.019	0.022	0.021	0.032	0.024	-	0.028	0.022	0.023	0.021	0.027	**0.018**
LFSD	$S_m \uparrow$	0.729	0.783	0.788	0.786	0.794	0.820	0.839	0.824	0.859	0.854	0.854	0.854	0.855	-	0.860	0.849	0.856	0.833	0.854	0.862	**0.874**
	$F_{\beta}^{max} \uparrow$	0.722	0.813	0.787	0.775	0.792	0.821	0.852	0.815	0.867	0.855	0.862	0.883	0.862	-	0.883	0.864	0.883	0.830	0.858	0.848	**0.892**
	$E_{\phi}^{max} \uparrow$	0.797	0.857	0.857	0.827	0.840	0.864	0.893	0.856	-	0.901	0.893	0.891	0.900	-	-	0.877	**0.902**	0.869	0.901	0.897	**0.902**
	$MAE \downarrow$	0.214	0.146	0.127	0.119	0.118	0.095	0.083	0.106	0.086	0.086	0.078	0.076	0.086	-	0.082	0.083	0.086	0.093	0.072	0.071	**0.066**
SIP	$S_m \uparrow$	0.727	0.653	0.716	0.842	0.835	0.850	0.806	0.864	-	0.875	0.879	**0.886**	0.875	0.875	-	-	0.867	-	0.879	0.858	0.875
	$F_{\beta}^{max} \uparrow$	0.751	0.657	0.694	0.838	0.830	0.851	0.821	0.862	-	0.867	0.885	**0.901**	0.892	0.848	-	-	0.874	-	0.883	0.842	0.891
	$E_{\phi}^{max} \uparrow$	0.853	0.759	0.829	0.901	0.895	0.903	0.875	0.910	-	0.914	0.923	0.922	0.915	0.908	-	-	0.913	-	0.922	0.909	**0.933**
	$MAE \downarrow$	0.200	0.185	0.139	0.071	0.075	0.064	0.085	0.063	-	**0.051**	**0.051**	0.057	0.054	0.059	-	-	0.062	-	0.055	0.063	**0.051**
DUT-RGBD	$S_m \uparrow$	-	0.695	0.499	-	0.526	0.736	0.702	0.831	0.791	0.801	0.808	0.818	-	0.899	0.903	-	-	0.889	-	0.898	**0.912**
	$F_{\beta}^{max} \uparrow$	-	0.692	0.411	-	0.458	0.740	0.659	0.823	0.767	0.771	0.790	0.898	-	0.918	0.901	-	-	0.795	-	0.903	**0.922**
	$E_{\phi}^{max} \uparrow$	-	0.800	0.654	-	0.709	0.823	0.796	0.899	0.859	0.856	0.861	0.859	-	0.937	0.937	-	-	0.933	-	0.931	**0.939**
	$MAE \downarrow$	-	0.220	0.243	-	0.201	0.144	0.122	0.097	0.113	0.100	0.093	0.076	-	0.043	0.043	-	-	0.048	-	0.045	**0.035**

TABLE II
QUANTITATIVE RESULTS OF OUR METHOD AND BASELINE NETWORKS OF THE ABLATION STUDY EXPERIMENTS ON NJU2K [31], NLPR [32], AND STERE [28]. THE BEST RESULTS ARE HIGHLIGHTED IN **BOLD**

Name	Networks	TCF	Sod	Rot	Sod-MT	Rot-MT	NJU2K [31]				NLPR [32]				STERE [28]			
							$S_m \uparrow$	$F_{\beta}^{max} \uparrow$	$E_{\phi}^{max} \uparrow$	$MAE \downarrow$	$S_m \uparrow$	$F_{\beta}^{max} \uparrow$	$E_{\phi}^{max} \uparrow$	$MAE \downarrow$	$S_m \uparrow$	$F_{\beta}^{max} \uparrow$	$E_{\phi}^{max} \uparrow$	$MAE \downarrow$
M_1	basic-TCF		√				0.835	0.862	0.857	0.094	0.786	0.807	0.815	0.070	0.795	0.829	0.814	0.111
M_2	basic	√	√				0.896	0.905	0.904	0.042	0.894	0.889	0.944	0.037	0.889	0.895	0.910	0.060
M_3	basic + Rot	√	√	√			0.903	0.904	0.915	0.040	0.901	0.895	0.948	0.031	0.892	0.907	0.932	0.052
M_5	basic + Rot + Sod-MT	√	√	√	√		0.910	0.922	0.938	0.035	0.926	0.915	0.958	0.023	0.911	0.915	0.942	0.044
M_6	basic + Rot + Rot-MT	√	√	√		√	0.906	0.913	0.928	0.038	0.904	0.905	0.953	0.027	0.901	0.910	0.937	0.048
	Our method	√	√	√	√	√	**0.913**	**0.928**	**0.943**	**0.034**	**0.927**	**0.923**	**0.961**	**0.021**	**0.913**	**0.918**	**0.945**	**0.038**
M_4	Sod-MT	√	√		√		0.909	0.915	0.933	0.036	0.925	0.912	0.951	0.025	0.903	0.912	0.942	0.048

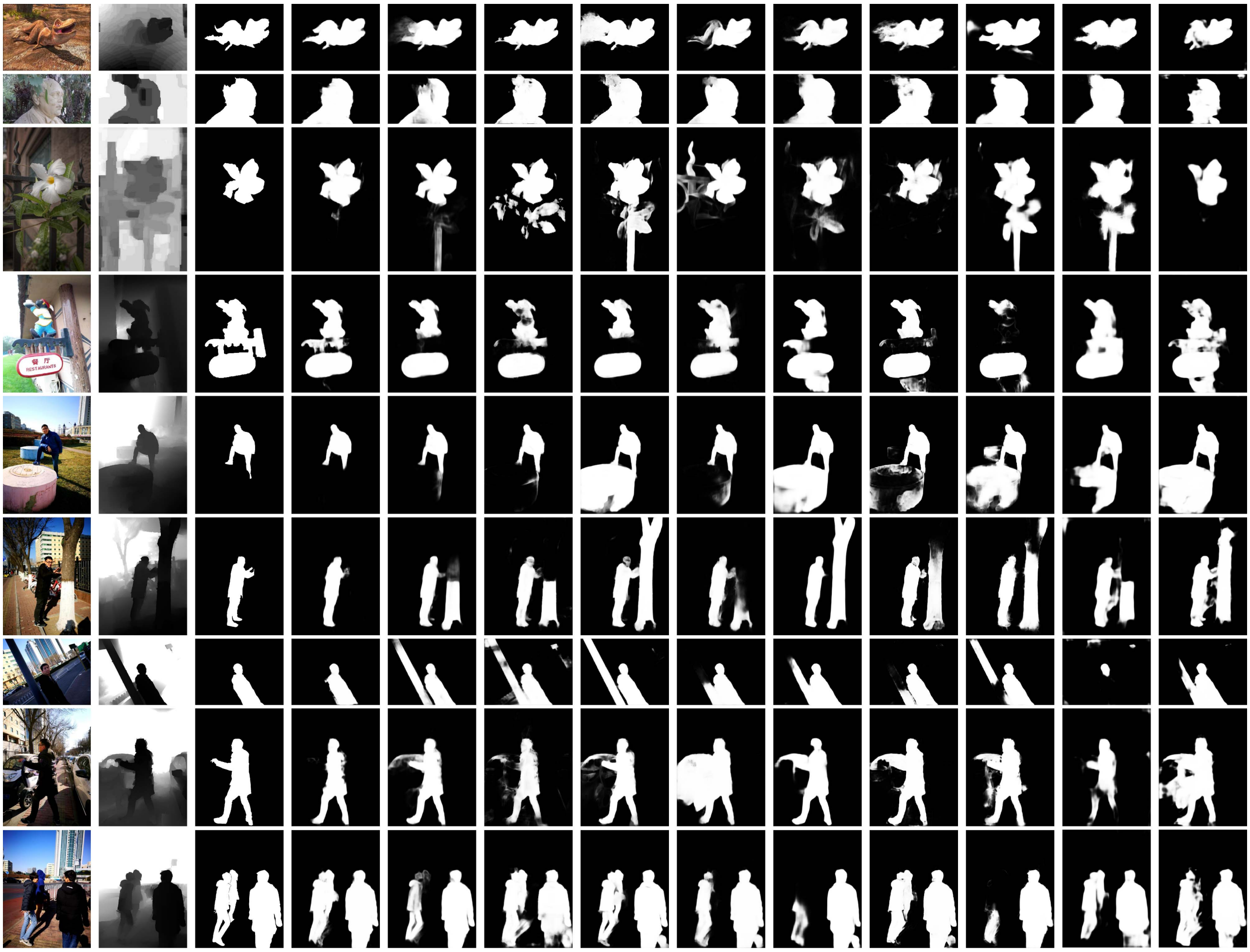
TABLE III
QUANTITATIVE RESULTS OF OUR METHOD AND BASELINE NETWORKS OF THE ABLATION STUDY EXPERIMENTS ON RGBD135 [33], LFSD [70], SIP [17], AND DUT-RGBD [24]. THE BEST RESULTS ARE HIGHLIGHTED IN **BOLD**

Name	Networks	TCF	Sod	Rot	Sod-MT	Rot-MT	RGBD135 [33]				LFSD [70]				SIP [17]				DUT-RGBD [24]			
							$S_m \uparrow$	$F_{\beta}^{max} \uparrow$	$E_{\phi}^{max} \uparrow$	$MAE \downarrow$	$S_m \uparrow$	$F_{\beta}^{max} \uparrow$	$E_{\phi}^{max} \uparrow$	$MAE \downarrow$	$S_m \uparrow$	$F_{\beta}^{max} \uparrow$	$E_{\phi}^{max} \uparrow$	$MAE \downarrow$	$S_m \uparrow$	$F_{\beta}^{max} \uparrow$	$E_{\phi}^{max} \uparrow$	$MAE \downarrow$
M_1	basic-TCF		√				0.749	0.833	0.741	0.074	0.696	0.751	0.706	0.180	0.711	0.726	0.720	0.146	0.744	0.777	0.753	0.093
M_2	basic	√	√				0.906	0.898	0.939	0.039	0.830	0.865	0.880	0.087	0.853	0.863	0.906	0.071	0.778	0.798	0.805	0.082
M_3	basic+Rot	√	√	√			0.912	0.903	0.958	0.037	0.845	0.872	0.893	0.079	0.855	0.881	0.912	0.068	0.786	0.800	0.821	0.071
M_5	basic + Rot + Sod-MT	√	√	√	√		0.927	0.905	0.969	0.020	0.872	0.889	0.900	0.071	0.871	0.888	0.928	0.059	0.881	0.889	0.904	0.057
M_6	basic + Rot + Rot-MT	√	√	√		√	0.917	0.903	0.959	0.033	0.851	0.882	0.896	0.078	0.860	0.871	0.925	0.065	0.871	0.878	0.893	0.063
	Our method	√	√	√	√	√	**0.932**	**0.915**	**0.971**	**0.018**	**0.874**	**0.892**	**0.903**	**0.066**	**0.875**	**0.891**	**0.933**	**0.051**	**0.912**	**0.922**	**0.939**	**0.035**
M_4	Sod-MT	√	√		√		0.921	0.912	0.962	0.029	0.854	0.868	0.899	0.077	0.862	0.867	0.918	0.063	0.875	0.885	0.901	0.059

and the depth image, while the third (denoted as "basic+Rot") is to add the rotation angle branch to "basic," which means that we directly apply SG-CNN with only labeled data to predict RGB-D saliency maps.

Apart from the supervised loss on labeled data in the SG-CNN, another two baseline networks are built to train the network by fusing additional consistency loss from unlabeled data. The first one (denoted as "basic+Rot+Sod-MT") is to add the consistency loss from the only saliency detection while another (denoted as "basic+Rot+Rot-MT") is to add the consistency loss from the only angle rotation prediction. Lastly, we build a baseline network (denoted as "Sod-MT") by removing the angle rotation prediction from our network. It means that "Sod-MT" follows the mechanism of the original mean teacher framework [36] to use the supervised loss on saliency detection and the consistency loss on saliency detection to train a network.

Quantitative comparisons: Table II summaries metric values of our network and baseline networks on NJUD [31], NLPR [32], and STERE [28], while Table III compares metric values on other four benchmarks, i.e., RGBD135 [33], LFSD [70], SIP [17], and DUT-RGBD [24]. From the results, we have the following observations: (i) "basic" has superior metric performance over "basic-TCF," which means that our TCF module can produce a more accurate feature map than a simple element-wisely addition when combing six



(a) Input RGB (b) Input depth (c) Ground truth (d) Our method (e) BBS-Net [67] (f) CMW Net [66] (g) HDF-Net [44] (h) PGA-Net [46] (i) DANet [45] (j) UCNet [18] (k) JLD CF [25] (l) DMRA [24] (m) CPFP [23]

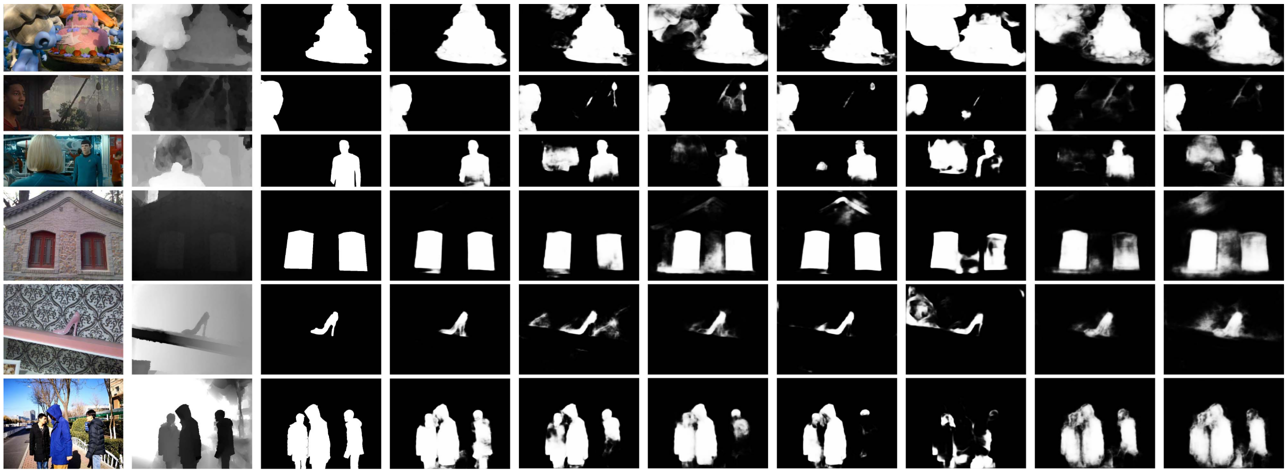
Fig. 5. Visual comparison of saliency map results produced by different methods. (a) Input RGB image with different complex scenarios; (b) Input depth image; (c) Ground truth (denoted as ‘GT’); (d)-(h) Saliency maps predicted by our method and compared RGB-D saliency detectors. Apparently, our network produces more accurate saliency maps than compared methods.

feature maps, which are from three adjacent CNN layers from the RGB images and the depth maps. (ii) “basic+Rot” has higher scores in terms of four metrics than “basic,” which indicates that the self-supervised prediction on an angle rotation helps our method to produce more accurately identify saliency maps from RGB-D images. (iii) “basic+Rot+Sod-MT” and “basic+Rot+Rot-MT” produce smaller MAE results and larger S_m, $F_\beta^{\max}$, and $E_\phi^{\max}$ than “basic+Rot,” demonstrating that additional consistency loss from unlabeled data is capable to enhance the RGB-D saliency detection performance with only labeled data. (iv) Moreover, “basic+Rot+Sod-MT” has better results than “basic+Rot+Rot-MT” on seven benchmark datasets. It shows that the saliency detection has a more contribution than the image rotation angle prediction to the success of our method when exploring the consistency loss from unlabeled data. (v) By combining two consistency losses from the saliency detection and the rotation angle prediction, our method has the best metric performance on all seven benchmark datasets. (vi) Our method can more accurately detect saliency regions than “Sod-MT”. It indicates that the rotation angle prediction benefits the mean teacher model for RGB-D saliency detection.

Visual comparisons: We further visually compare the saliency maps produced by our method and six baseline networks; see Fig. (6). Although there are one or more salient objects in different input images, our method consistently produce more accurate saliency map than all six baseline methods (i.e., M_1 to M_6). It further proves the effectiveness of our RGB-D saliency detection network and its major components.

Model size and inference time: Given an input RGB image and an input depth image, we pass them into the student network (SG-Net) to predict a saliency detection map, which is then taken as the final result of our semi-supervised RGB-D saliency detection network. The model size of our network is 239.1 MB, and our method takes about 0.024 seconds (42 FPS) to process a pair of 256 × 256 RGB-D images on a single NVIDIA 2080Ti GPU card.

Table VI reports the model size and inference time of our network and state-of-the-art methods. Apparently, as a lightweight CNN model, PGA-Net has the best performance of the model



(a) RGB (b) depth (c) GT (d) ours (e) M_6 (f) M_5 (g) M_4 (h) M_3 (i) M_2 (j) M_1

Fig. 6. Visual comparison of saliency map results produced by different methods. (a) Input RGB image from benchmark datasets; (b) Input depth image; (c) Ground truths (denoted as 'GT'); (d)-(j) Saliency maps predicted by our method and six constructed baseline networks (i.e., M_1 to M_6); please refer to Table II and Table III for the explanation of M_1 to M_6. Apparently, our network produces more accurate saliency maps than six baseline networks.

TABLE IV
QUANTITATIVE RESULTS OF OUR METHOD WITH DIFFERENT PRETEXT TASKS ON NJU2K [31], NLPR [32], AND STERE [28]. THE BEST RESULTS ARE HIGHLIGHTED IN **BOLD**

Name	**NJU2K [31]**				**NLPR [32]**				**STERE [28]**			
	$S_m\uparrow$	$F_\beta^{max}\uparrow$	$E_\phi^{max}\uparrow$	$MAE\downarrow$	$S_m\uparrow$	$F_\beta^{max}\uparrow$	$E_\phi^{max}\uparrow$	$MAE\downarrow$	$S_m\uparrow$	$F_\beta^{max}\uparrow$	$E_\phi^{max}\uparrow$	$MAE\downarrow$
our-cmMS	0.896	0.913	0.929	0.040	0.903	0.906	0.941	0.044	0.895	0.898	0.916	0.060
out-HDF	0.899	0.913	0.933	0.037	0.907	0.910	0.943	0.043	0.898	0.900	0.919	0.060
our-jigsaw	0.903	0.909	0.922	0.042	0.902	0.902	0.944	0.039	0.884	0.895	0.927	0.058
our-inpainting	0.904	0.912	0.926	0.039	0.904	0.903	0.949	0.037	0.898	0.897	0.933	0.054
Our method	**0.913**	**0.928**	**0.943**	**0.034**	**0.927**	**0.923**	**0.961**	**0.021**	**0.913**	**0.918**	**0.945**	**0.038**
our-decoupling	0.907	0.915	0.936	0.042	0.914	0.911	0.957	0.035	0.903	0.909	0.938	0.049
our-supervised-unlabeled	0.906	0.916	0.932	0.038	0.916	0.912	0.950	0.035	0.899	0.902	0.934	0.053
teacher-network	0.909	0.924	0.940	0.035	0.922	0.922	0.961	0.027	0.911	0.914	0.943	0.040

TABLE V
QUANTITATIVE RESULTS OF OUR METHOD WITH DIFFERENT PRETEXT TASKS ON RGBD135 [33], LFSD [70], SIP [17], AND DUT-RGBD [24]. THE BEST RESULTS ARE HIGHLIGHTED IN **BOLD**

Name	**RGBD135 [33]**				**LFSD [70]**				**SIP [17]**				**DUT-RGBD [24]**			
	$S_m\uparrow$	$F_\beta^{max}\uparrow$	$E_\phi^{max}\uparrow$	$MAE\downarrow$	$S_m\uparrow$	$F_\beta^{max}\uparrow$	$E_\phi^{max}\uparrow$	$MAE\downarrow$	$S_m\uparrow$	$F_\beta^{max}\uparrow$	$E_\phi^{max}\uparrow$	$MAE\downarrow$	$S_m\uparrow$	$F_\beta^{max}\uparrow$	$E_\phi^{max}\uparrow$	$MAE\downarrow$
our-cmMS	0.897	0.883	0.948	0.032	0.848	0.842	0.883	0.078	0.859	0.862	0.904	0.076	0.872	0.877	0.892	0.065
ours-HDF	0.897	0.886	0.951	0.030	0.852	0.844	0.884	0.075	0.864	0.862	0.906	0.072	0.875	0.877	0.894	0.060
our-jigsaw	0.902	0.879	0.951	0.035	0.846	0.851	0.885	0.083	0.854	0.866	0.911	0.067	0.866	0.875	0.888	0.067
our-inpainting	0.904	0.890	0.956	0.033	0.851	0.852	0.889	0.079	0.860	0.869	0.913	0.074	0.871	0.875	0.891	0.064
Our method	**0.932**	**0.915**	**0.971**	**0.018**	**0.874**	**0.892**	**0.903**	**0.066**	**0.875**	**0.891**	**0.933**	**0.051**	**0.912**	**0.922**	**0.939**	**0.035**
our-decoupling	0.923	0.902	0.950	0.023	0.861	0.881	0.890	0.073	0.861	0.878	0.925	0.066	0.901	0.916	0.931	0.046
our-supervised-unlabeled	0.905	0.892	0.958	0.020	0.864	0.863	0.890	0.074	0.863	0.873	0.915	0.065	0.880	0.882	0.901	0.061
teacher-network	0.931	0.909	0.966	0.018	0.871	0.889	0.897	0.069	0.872	0.889	0.930	0.054	0.908	0.918	0.934	0.036

size (63.9 M) and the inference time (72 FPS), but their quantitative results on seven benchmark results is not comparable to more recent RGB-D saliency detectors (i.e., HDF-Net, Cas-Gnn, CMWNet, BBS-Net); see Table I. Apart from BBS-Net, our network achieves comparable results or even smaller results than model or even smaller than these recent RGB-D saliency detectors. However, our method (42 FPS) has a faster inference time than BBS-Net (26 FPS). Although there are still some works (e.g., 45 FPS of cmMS, 46 FPS of ATSA, 50 FPS of CTMF, 52 FPS of HDF-Net) with a faster inference time, our method is capable to infer saliency maps form RGB-D data in a real-time manner, since it takes about 0.024 (42 FPS) to process a pair of RGB-D images (256 × 256). In summary, although our method does not perform best in the model size and inference time, our method better identifies salient objects than state-of-the-art RGB-D saliency detectors in almost all seven benchmark datasets, as shown in Table I of the revised manuscript. Moreover, we take a task of reducing the model size and speeding up our inference process as a future direction of our work.

TABLE VI
MODEL SIZE AND INFERENCE TIME OF DIFFERENT RGB-D SALIENCY DETECTION METHODS. FPS: FRAMES PER SECOND

Method	Model size (MB)↓	Inference time (FPS) ↑
PCA [22]	533.6	15
TANet [41]	951.9	14
MMCI [42]	929.7	19
PDNet [79]	192	19
CPFP [23]	278	6
CTMF [40]	826	50
DMRA [24]	238.8	22
D^{3}Net [17]	439.7	18
SSF [26]	329	13
UCNet [18]	308	20
JLDCF [25]	520	9
HDF-Net [44]	170	52
DANet [45]	106.7	32
Cas-GNN [48]	219.1	40
CMWNet [66]	156	30
ATSA [43]	128.9	46
CoNet [47]	167.6	34
Ours	239.1	42

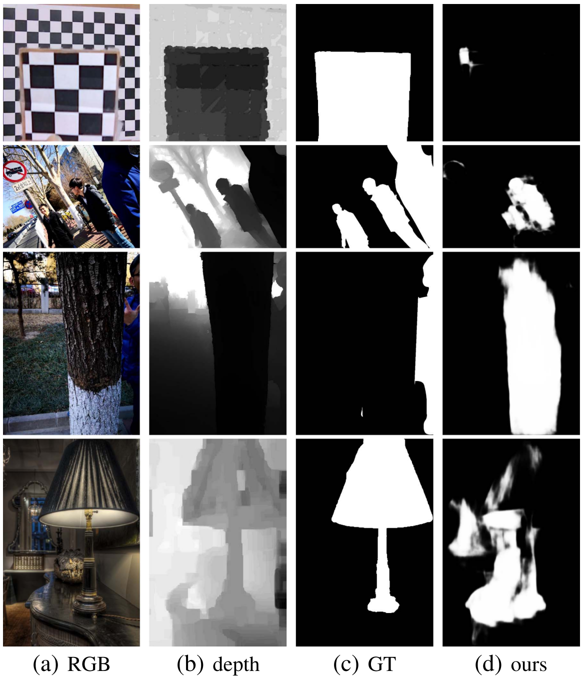


Fig. 7. Failure cases of our RGB-D saliency detection method. (a) Input RGB image from benchmark datasets; (b) Input depth image; (c) Ground truths (denoted as 'GT'); and (d) Our results.

D. Discussion

Self-supervised rotation angle prediction loss on unlabeled data: Note that the rotation angle prediction is conducted in a self-supervised learning manner. Hence, it is natural to explore the corresponding result when we remove the consistency loss on the rotation angle prediction and add the self-supervised rotation angle prediction on unlabeled data. In this regard, we have conducted an experiment to construct a baseline network (denoted as "our-supervised-unlabeled") by employing a supervised loss on unlabeled data to replace the consistency loss on rotation predictions of unlabeled data. Table IV and Table V report the quantitative results on seven RGB-D saliency detection benchmark datasets of our method and "our-supervised-unlabeled". Apparently, we can find that our method consistently outperforms "our-supervised-unlabeled" in terms of all four metrics on seven benchmark datasets. It indicates that the consistency loss on rotation predictions enables us to better identify saliency objects when compared to a supervised loss on unlabeled data.

Rotation feature decoupling: Note that [56] decoupled the image rotation features to rotation related and unrelated parts. Then, we conduct an experiment to construct a network (denoted as "our-decoupling") to modify our network by using the rotation feature disentangling in [56] for the image rotation angle prediction in our work. Table IV and Table V compares the quantitative results of our network and "our-decoupling" on seven RGB-D saliency detection benchmark datasets. It shows that our network has a superior metric performance over "our-decoupling" in terms of all seven benchmark datasets. The reason behind is that the rotation angle prediction mechanism in [56] increases the number of network parameters and the network training difficulties, thereby degrading the RGB-D saliency detection performance.

Results of the teacher network: Table IV and Table V summarize the results of the student network and the teacher network in our semi-supervised RGB-D saliency detection method, showing that the final results of the student network and the teacher network are close for all seven benchmark datasets. Following all research works based on the mean-teacher framework, we also utilized the student network to do the saliency inference from the input paired RGB-D images.

The choice of pretext tasks: Note that the salient objects can be with arbitrary angles. Hence, our work takes the rotation prediction as the auxiliary task to enable our method to better understand the angle information of the target salient objects, thereby making the RGB-D salient object detection more accurate. Moreover, we have conducted an experiment to construct two baseline networks (denoted as "our-jigsaw" and "our-inpainting") by replacing the rotation angle prediction with the jigsaw puzzle and the image inpainting as the auxiliary task of our RGB-D saliency detection. Table IV and Table V summaries the quantitative results on seven RGB-D saliency detection benchmark datasets, showing that our method outperforms "our- jigsaw" and "our-inpainting" in terms of four metrics on all seven benchmark datasets. It indicates that taking the self-supervised rotation angle prediction as the auxiliary task can better identify salient objects from paired RGB-D data than that with the jigsaw puzzle and image inpainting.

Failure cases: Although our method has obtained superior RGB-D saliency detection performance on the seven benchmark datasets, it also has the failure cases, which are also challenging for existing state-of-the-art RGB-D saliency detectors. For example, our method may fail for (i) salient objects with complex salient object boundaries (see the first row of Fig. 7); (ii) salient regions with only partial human objects (see the second and third rows of Fig. 7); and (iii) salient objects with a close intensity distribution with non-salient backgrounds (see the last row of Fig. 7). We take the task of addressing those failure cases as a future direction of our work.

V. Conclusion

This paper presents a self-supervised self-ensembling network for RGB-D saliency detection by learning from both labeled and unlabeled data. We first develop a self-guided multi-task convolutional neural network for simultaneously predicting a saliency map and classifying a rotation angle of the image without any additional supervision signal. Then we employ the self-ensembling framework to leverage additional unlabeled data to further improve the performance of RGB-D saliency detection. Experimental results on seven benchmark datasets show that our network consistently outperforms the state-of-the-art methods both quantitatively and visually. Considering diverse pretext tasks and more unlabeled data into our network is taken as one of the future directions of our work.

References

[1] K. Fu, Q. Zhao, and I. Y.-H. Gu, "Refinet: A deep segmentation assisted refinement network for salient object detection," *IEEE Trans. Multimedia*, vol. 21, pp. 457–469, 2019.

[2] H. Xiao, J. Feng, Y. Wei, M. Zhang, and S. Yan, "Deep salient object detection with dense connections and distraction diagnosis," *IEEE Trans. Multimedia*, vol. 20, pp. 3239–3251, 2018.

[3] X. Ding and Z. Chen, "Improving saliency detection based on modeling photographer's intention," *IEEE Trans. Multimedia*, vol. 21, pp. 124–134, 2019.

[4] L. Ye, Z. Liu, L. Li, L. Shen, C. Bai, and Y. Wang, "Salient object segmentation via effective integration of saliency and objectness," *IEEE Trans. Multimedia*, vol. 19, pp. 1742–1756, 2017.

[5] C.-C. Tsai, K.-J. Hsu, Y.-Y. Lin, X. Qian, and Y.-Y. Chuang, "Deep co-saliency detection via stacked autoencoder-enabled fusion and self-trained cnns," *IEEE Trans. Multimedia*, vol. 22, pp. 1016–1031, 2020.

[6] G. Ma, C. Chen, S. Li, C. Peng, A. Hao, and H. Qin, "Salient object detection via multiple instance joint re-learning," *IEEE Trans. Multimedia*, vol. 22, pp. 324–336, 2020.

[7] L. Zhu *et al.*, "Aggregating attentional dilated features for salient object detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 10, pp. 3358–3371, Oct. 2020.

[8] L. Wang, R. Chen, L. Zhu, H. Xie, and X. Li, "Deep sub-region network for salient object detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 2, pp. 728–741, Feb. 2021.

[9] W. Wang, J. Shen, and F. Porikli, "Saliency-aware geodesic video object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3395–3402.

[10] S. Hong, T. You, S. Kwak, and B. Han, "Online tracking by learning discriminative saliency map with convolutional neural network," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 597–606.

[11] H. Hadizadeh and I. V. Bajic, "Saliency-aware video compression," *IEEE Trans. Image Process.*, vol. 23, no. 1, pp. 19–33, Jan. 2014.

[12] H. Zhao, X. Mao, X. Jin, J. Shen, F. Wei, and J. Feng, "Real-time saliency-aware video abstraction," *Vis. Comput.*, vol. 25, no. 11, pp. 973–984, 2009.

[13] M.-M. Cheng, F.-L. Zhang, N. J. Mitra, X. Huang, and S.-M. Hu, "RepFinder: Finding approximately repeated scene elements for image editing," *ACM Trans. Graph.*, ACM, vol. 29, no. 4, pp. 1–8, 2010.

[14] L. Zhu, X. Hu, C.-W. Fu, J. Qin, and P.-A. Heng, "Saliency-aware texture smoothing," *IEEE Trans. Vis. Comput. Graphics*, vol. 26, no. 7, pp. 2471–2484, Jul. 2020.

[15] D. Liu, K. Zhang, and Z. Chen, "Attentive cross-modal fusion network for RGB-D saliency detection," *IEEE Trans. Multimedia*, vol. 23, pp. 967–981, 2021.

[16] B. Jiang, Z. Zhou, X. Wang, J. Tang, and B. Luo, "cmSalGAN: RGB-D salient object detection with cross-view generative adversarial networks," *IEEE Trans. Multimedia*, vol. 23, pp. 1343–1353, 2021.

[17] D.-P. Fan, Z. Lin, Z. Zhang, M. Zhu, and M.-M. Cheng, "Rethinking RGB-D salient object detection: Models, data sets, and large-scale benchmarks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 5, pp. 2075–2089, May 2021.

[18] J. Zhang *et al.*, "UC-Net: Uncertainty inspired RGB-D saliency detection via conditional variational autoencoders," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 8582–8591.

[19] D. Feng, N. Barnes, S. You, and C. McCarthy, "Local background enclosure for RGB-D salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2343–2350.

[20] H. Song, Z. Liu, H. Du, G. Sun, O. L. Meur, and T. Ren, "Depth-aware salient object detection and segmentation via multiscale discriminative saliency fusion and bootstrap learning," *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4204–4216, Sep. 2017.

[21] R. Cong, J. Lei, H. Fu, J. Hou, Q. Huang, and S. Kwong, "Going from RGB to RGBD saliency: A depth-guided transformation model," *IEEE Trans. Cybern.*, vol. 50, no. 8, pp. 3627–3639, Aug. 2020.

[22] H. Chen and Y. Li, "Progressively complementarity-aware fusion network for RGB-D salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3051–3060.

[23] J.-X. Zhao, Y. Cao, D.-P. Fan, M.-M. Cheng, X.-Y. Li, and L. Zhang, "Contrast prior and fluid pyramid integration for RGBD salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3927–3936.

[24] Y. Piao, W. Ji, J. Li, M. Zhang, and H. Lu, "Depth-induced multi-scale recurrent attention network for saliency detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 7254–7263.

[25] K. Fu, D.-P. Fan, G.-P. Ji, and Q. Zhao, "JL-DCF: Joint learning and densely-cooperative fusion framework for RGB-D salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 3052–3062.

[26] M. Zhang, W. Ren, Y. Piao, Z. Rong, and H. Lu, "Select, supplement and focus for RGB-D saliency detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 3472–3481.

[27] N. Liu, N. Zhang, and J. Han, "Learning selective self-mutual attention for RGB-D saliency detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 13 756–13 765.

[28] Y. Niu, Y. Geng, X. Li, and F. Liu, "Leveraging stereopsis for saliency analysis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 454–461.

[29] J. Guo, T. Ren, and J. Bei, "Salient object detection for RGB-D image via saliency evolution," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2016, pp. 1–6.

[30] R. Cong, J. Lei, C. Zhang, Q. Huang, X. Cao, and C. Hou, "Saliency detection for stereoscopic images based on depth confidence analysis and multiple cues fusion," *IEEE Signal Process. Lett.*, vol. 23, no. 6, pp. 819–823, Jun. 2016.

[31] R. Ju, L. Ge, W. Geng, T. Ren, and G. Wu, "Depth saliency based on anisotropic center-surround difference," in *Proc. IEEE Int. Conf. Image Process.*, 2014, pp. 1115–1119.

[32] H. Peng, B. Li, W. Xiong, W. Hu, and R. Ji, "RGBD salient object detection: A benchmark and algorithms," in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2014, pp. 92–109.

[33] Y. Cheng, H. Fu, X. Wei, J. Xiao, and X. Cao, "Depth enhanced saliency detection method," in *Proc. Int. Conf. Internet Multimedia Comput. Serv.*, 2014, pp. 23–27.

[34] F. Liang, L. Duan, W. Ma, Y. Qiao, Z. Cai, and L. Qing, "Stereoscopic saliency model using contrast and depth-guided-background prior," *Neurocomputing*, vol. 275, pp. 2227–2238, 2018.

[35] A. Wang and M. Wang, "RGB-D salient object detection via minimum barrier distance transform and saliency fusion," *IEEE Signal Process. Lett.*, vol. 24, no. 5, pp. 663–667, May 2017.

[36] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 1195–1204.

[37] Z. Chen, L. Zhu, L. Wan, S. Wang, W. Feng, and P.-A. Heng, "A multi-task mean teacher for semi-supervised shadow detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 5611–5620.

[38] Z. Liu, S. Shi, Q. Duan, W. Zhang, and P. Zhao, "Salient object detection for RGB-D image by single stream recurrent convolution neural network," *Neurocomputing*, vol. 363, pp. 46–57, 2019.

[39] P. Huang, C.-H. Shen, and H.-F. Hsiao, "RGBD salient object detection using spatially coherent deep learning framework," in *Proc. IEEE 23rd Int. Conf. Digit. Signal Process.*, 2018, pp. 1–5.

[40] J. Han, H. Chen, N. Liu, C. Yan, and X. Li, "CNNs-based RGB-D saliency detection via cross-view transfer and multiview fusion," *IEEE Trans. Cybern.*, vol. 48, no. 11, pp. 3171–3183, Nov. 2018.

[41] H. Chen and Y. Li, "Three-stream attention-aware network for RGB-D salient object detection," *IEEE Trans. Image Process.*, vol. 28, no. 6, pp. 2825–2835, Jun. 2019.

[42] H. Chen, Y. Li, and D. Su, "Multi-modal fusion network with multi-scale multi-path and cross-modal interactions for RGB-D salient object detection," *Pattern Recognit.*, vol. 86, pp. 376–385, 2019.

[43] M. Zhang, S. X. Fei, J. Liu, S. Xu, Y. Piao, and H. Lu, "Asymmetric two-stream architecture for accurate RGB-D saliency detection," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 374–390.

[44] Y. Pang, L. Zhang, X. Zhao, and H. Lu, "Hierarchical dynamic filtering network for RGB-D salient object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 235–252.

[45] X. Zhao, L. Zhang, Y. Pang, H. Lu, and L. Zhang, "A single stream network for robust and real-time RGB-D salient object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 646–662.

[46] S. Chen and Y. Fu, "Progressively guided alternate refinement network for RGB-D salient object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 520–538.

[47] W. Ji, J. Li, M. Zhang, Y. Piao, and H. Lu, "Accurate RGB-D salient object detection via collaborative learning," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 52–69.

[48] A. Luo, X. Li, F. Yang, Z. Jiao, H. Cheng, and S. Lyu, "Cascade graph neural networks for RGB-D salient object detection," in *Proc. Eur. Conf. Comput. Vis. ICLR*, Springer, 2020, pp. 346–364.

[49] S. Laine and T. Aila, "Temporal ensembling for semi-supervised learning," in *Proc. Int. Conf. Learn. Representations*, 2017. [Online]. Available: https://openreview.net/forum?id=BJ6oOfqge

[50] L. Yu, S. Wang, X. Li, C.-W. Fu, and P.-A. Heng, "Uncertainty-aware self-ensembling model for semi-supervised 3D left atrium segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Interv.*, Springer, 2019, pp. 605–613.

[51] M. Zhao, L. Jiao, W. Ma, H. Liu, and S. Yang, "Classification and saliency detection by semi-supervised low-rank representation," *Pattern Recognit.*, vol. 51, pp. 281–294, 2016.

[52] Y. Zhou, S. Huo, W. Xiang, C. Hou, and S.-Y. Kung, "Semi-supervised salient object detection using a linear feedback control system model," *IEEE Trans. Cybern.*, vol. 49, no. 4, pp. 1173–1185, Apr. 2019.

[53] D. Zhang, J. Han, Y. Zhang, and D. Xu, "Synthesizing supervision for learning deep saliency network without human annotation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 7, pp. 1755–1769, Jul. 2020.

[54] D. Zhang, H. Tian, and J. Han, "Few-cost salient object detection with adversarial-paced learning," in *Annual Conf. Neural Informat. Process. Syst. (NeurIPS)*, 2020.

[55] X. Zhai, A. Oliver, A. Kolesnikov, and L. Beyer, "S4L: Self-supervised semi-supervised learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1476–1485.

[56] Z. Feng, C. Xu, and D. Tao, "Self-supervised representation learning by rotation feature decoupling," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 10 364–10374.

[57] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised representation learning by predicting image rotations," in *Proc. Int. Conf. Learn. Representations*, 2018.

[58] M. Caron, P. Bojanowski, A. Joulin, and M. Douze, "Deep clustering for unsupervised learning of visual features," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 132–149.

[59] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2536–2544.

[60] R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2016, pp. 649–666.

[61] D. Pathak, R. Girshick, P. Dollár, T. Darrell, and B. Hariharan, "Learning features by watching objects move," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2701–2710.

[62] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018.

[63] S. Song, S. P. Lichtenberg, and J. Xiao, "Sun RGB-D: A RGB-D scene understanding benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 567–576.

[64] L. Qu, S. He, J. Zhang, J. Tian, Y. Tang, and Q. Yang, "RGBD salient object detection via deep fusion," *IEEE Trans. Image Process.*, vol. 26, no. 5, pp. 2274–2285, May 2017.

[65] C. Li, R. Cong, Y. Piao, Q. Xu, and C. C. Loy, "RGB-D salient object detection with cross-modality modulation and selection," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 225–241.

[66] G. Li, Z. Liu, L. Ye, Y. Wang, and H. Ling, "Cross-modal weighting network for RGB-D salient object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 665–681.

[67] D.-P. Fan, Y. Zhai, A. Borji, J. Yang, and L. Shao, "BBS-Net: RGB-D salient object detection with a bifurcated backbone strategy network," in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2020, pp. 275–292.

[68] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[69] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 248–255.

[70] N. Li, J. Ye, Y. Ji, H. Ling, and J. Yu, "Saliency detection on light field," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 2806–2813.

[71] C. Lang, J. Feng, S. Feng, J. Wang, and S. Yan, "Dual low-rank pursuit: Learning salient features for saliency detection," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 6, pp. 1190–1200, Jun. 2016.

[72] N. Liu, J. Han, and M.-H. Yang, "PiCANet: Learning pixel-wise contextual attention for saliency detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3089–3098.

[73] D.-P. Fan, M.-M. Cheng, Y. Liu, T. Li, and A. Borji, "Structure-measure: A new way to evaluate foreground maps," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 4548–4557.

[74] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 1597–1604.

[75] Q. Hou, M. Cheng, X. Hu, A. Borji, Z. Tu, and P. H. S. Torr, "Deeply supervised salient object detection with short connections," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 5300–5309.

[76] Z. Deng *et al.*, "R^3 Net: Recurrent residual refinement network for saliency detection," in *Proc. Int. Joint Conf. Artif. Intell.*, 2018, pp. 684–690.

[77] D.-P. Fan, C. Gong, Y. Cao, B. Ren, M.-M. Cheng, and A. Borji, "Enhanced-alignment measure for binary foreground map evaluation," in *Proc. Int. Joint Conf. Artif. Intell.*, 2018, pp. 698–704.

[78] F. Perazzi, P. Krähenbühl, Y. Pritch, and A. Hornung, "Saliency filters: Contrast based filtering for salient region detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 733–740.

[79] C. Zhu, X. Cai, K. Huang, T. H. Li, and G. Li, "PDNet: Prior-model guided depth-enhanced network for salient object detection," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2019, pp. 199–204.

Lei Zhu received the Ph.D. degree from the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong, in 2017. He is currently an assistant professor at The Hong Kong University of Science and Technology. Before that, He was a Postdoctoral Research Associate, the University of Cambridge, U.K. His research interests include computer graphics, computer vision, medical AI, and deep learning.

Xiaoqiang Wang received the B.Eng. degree from Zhejiang University, Hangzhou, China, where he is currently working toward the master's degree with the College of Computer Science and Technology. His research interests include computer vision and machine learning, with a particular focus on multimodal data analysis.

Ping Li (Member, IEEE) received the Ph.D. degree in computer science and engineering from The Chinese University of Hong Kong, Hong Kong. He is currently a Research Assistant Professor with The Hong Kong Polytechnic University, Hong Kong. He has one image/video processing national invention patent, and has excellent research project reported worldwide by *ACM TechNews*. His current research interests include image/video stylization, artistic rendering and synthesis, and creative media.

Xin Yang received the B.S. degree in computer Science from Jilin University, Changchun, China, in 2007 and the joint Ph.D. degree from Zhejiang University, Hangzhou, China, and UC Davis for Graphics, in July 2012. He is currently a Professor with the Department of Computer Science, Dalian University of Technology, Dalian, China. From 2007 to June 2012, he was a joint Ph.D. Student with Zhejiang University and UC Davis for Graphics. His research interests include computer graphics and robotic vision.

Qing Zhang received the Ph.D. degree from the School of Computer Science, Wuhan University, Wuhan, China, in 2017. He is currently a Research Associate Professor with the School of Data and Computer Science, Sun Yat-Sen University, Guangzhou, China. His research interests include computer graphics, computer vision, and computational photography.

Weiming Wang received the bachelor's degree from the Huazhong University of Science and Technology, Wuhan, China, in 2007 and the Ph.D. degree from The Chinese University of Hong Kong, Hong Kong, in 2013. He is currently a Research Assistant Professor with Hong Kong Metropolitan University (HKMU), Hong Kong. From 2014 to 2015, he was an Assistant Researcher with the Shenzhen Institutes of Advanced Technology. After that, he worked in high-tech companies for a few years. In 2020, he joined HKMU. His research interests include image processing, deep learning, and computer graphics.

Carola-Bibiane Schönlieb received the Degree in mathematics from the Institute for Mathematics, University of Salzburg, Salzburg, Austria, in 2004 and the Ph.D. degree from the University of Cambridge, Cambridge, U.K., in 2009. From 2004 to 2005, she held a Teaching position with Salzburg. After one year of post-doctoral activity with the University of Gottingen, Germany, she became a Lecturer with the Department of Applied Mathematics and Theoretical Physics (DAMTP) in 2010, where she was promoted to a Reader in 2015 and a Professor in 2018. She is currently a Professor of applied mathematics with the DAMTP, University of Cambridge, where she is also the Head of Cambridge Image Analysis Group, the Director of the Cantab Capital Institute for Mathematics of Information, the Director of the EPSRC Centre for Mathematical and Statistical Analysis of Multimodal Clinical Imaging, and has been a Fellow of the Jesus College Cambridge since 2011. Her research interests include variational methods, partial differential equations and machine learning for image analysis, image processing, and inverse imaging problems

C. L. Philip Chen (Fellow, IEEE) received the Ph.D. degree in electrical engineering from Purdue University, West Lafayette, IN, USA, in 1988.

He is currently a Chair Professor and the Dean of the School of Computer Science and Engineering, South China University of Technology, Guangzhou, China. Being a Program Evaluator of the Accreditation Board of Engineering and Technology Education (ABET) in the U.S., for computer engineering, electrical engineering, and software engineering programs, he successfully architects the University of Macau's Engineering and Computer Science Programs receiving accreditations from Washington/Seoul Accord through Hong Kong Institute of Engineers (HKIE), of which is considered as his utmost contribution in engineering/computer science education for Macau as the former Dean of the Faculty of Science and Technology. He is a Fellow of AAAS, IAPR, CAA, and HKIE, a Member of the Academia Europaea (AE), European Academy of Sciences and Arts (EASA), and International Academy of Systems and Cybernetics Science (IASCYS). He was the recipient of the IEEE Norbert Wiener Award in 2018 for his contribution in systems and cybernetics, and machine learnings. He is also a highly cited Researcher by Clarivate Analytics in 2018 and 2019.

His current research interests include systems, cybernetics, and computational intelligence. Dr. Chen was the recipient of the 2016 Outstanding Electrical and Computer Engineers Award from his alma mater, Purdue University (in 1988), after he graduated from the University of Michigan at Ann Arbor, Ann Arbor, MI, USA in 1985. He was the IEEE Systems, Man, and Cybernetics Society President from 2012 to 2013, the Editor-in-Chief of the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS during 2014–2019, and he is currently the Editor-in-Chief of the IEEE TRANSACTIONS ON CYBERNETICS, and an Associate Editor for the IEEE TRANSACTIONS ON FUZZY SYSTEMS. From 2015 to 2017, he was the Chair of the TC 9.1 Economic and Business Systems of International Federation of Automatic Control and is currently the Vice President of the Chinese Association of Automation (CAA).