JL-DCF: Joint Learning and Densely-Cooperative Fusion Framework for RGB-D Salient Object Detection

Keren Fu[1] Deng-Ping Fan[2,3,*] Ge-Peng Ji[4] Qijun Zhao[1]
[1] College of Computer Science, Sichuan University [2] College of CS, Nankai University
[3] Inception Institute of Artificial Intelligence [4] School of Computer Science, Wuhan University
http://dpfan.net/JLDCF/

Abstract

*This paper proposes a novel joint learning and densely-cooperative fusion (**JL-DCF**) architecture for RGB-D salient object detection. Existing models usually treat RGB and depth as independent information and design separate networks for feature extraction from each. Such schemes can easily be constrained by a limited amount of training data or over-reliance on an elaborately-designed training process. In contrast, our JL-DCF learns from both RGB and depth inputs through a Siamese network. To this end, we propose two effective components: joint learning (JL), and densely-cooperative fusion (DCF). The JL module provides robust saliency feature learning, while the latter is introduced for complementary feature discovery. Comprehensive experiments on four popular metrics show that the designed framework yields a robust RGB-D saliency detector with good generalization. As a result, JL-DCF significantly advances the top-1 D3Net model by an average of ∼1.9% (S-measure) across six challenging datasets, showing that the proposed framework offers a potential solution for real-world applications and could provide more insight into the cross-modality complementarity task. The code will be available at https://github.com/kerenfu/JLDCF/.*

1. Introduction

Salient object detection (SOD) aims at detecting the objects in a scene that humans would naturally focus on [2, 9, 78]. It has many useful applications, including object segmentation and recognition [27, 32, 39, 51, 70, 79], image/video compression [24], video detection/summarization [19, 41], content-based image editing [14, 23, 42, 57, 63], informative common object discovery [71, 72], and image retrieval [8, 22, 37]. Many SOD models have been developed under the assumption that the inputs are individual RGB/color images [21, 47, 66, 74–76] or sequences [56, 62, 67, 68]. As depth cameras such as Kinect and RealSense become more and more popular, SOD from RGB-D inputs ("D" refers to depth) is emerging as an attractive topic. Although a number of prior works have tried to explore the role of depth in saliency analysis, several issues remain:

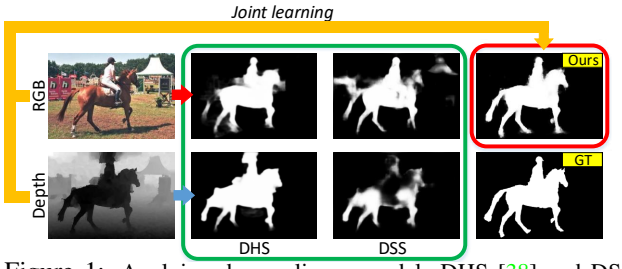

Figure 1: Applying deep saliency models DHS [38] and DSS [29], which are fed with an RGB image (1^{st} row) or a depth map (2^{nd} row). Both of the models are trained on a single RGB modality. By contrast, our *JL-DCF* considers both modalities and thus generates better results (last column).

(i) Deep-based RGB-D SOD methods are still under-explored: Despite more than one hundred papers on RGB SOD models being published since 2015 [15, 61, 64, 65, 69], there are only a few deep learning-based works focusing on RGB-D SOD. The first model utilizing convolutional neural networks (CNNs) for RGB-D SOD [49], which adopts a shallow CNN as the saliency map integration model, was described in 2017. Since then, only a dozen deep models have been proposed, as summarized in [18, 73], leaving large room for further improvement in performance.

(ii) Less effective feature extraction and fusion: Most learning-based models fuse features of different modalities either by early-fusion [18, 31, 40, 55] or late-fusion [26, 60]. Although these two simple strategies have achieved encouraging progress in this field in the past (as pointed out in [4]), they face challenges in either extracting representative multi-modal features or effectively fusing them. While other works have adopted a middle-fusion strategy [4, 5, 80], which conducts independent feature extraction and fusion using individual CNNs, their sophisticated net-

*Corresponding author: Deng-Ping Fan *(dengpfan@gmail.com)*

work architectures and large number of parameters require an elaborately-designed training process and large amount of training data. Unfortunately, high-quality depth maps are still sparse [77], which may lead to sub-optimal solutions of deep learning-based models.

Motivation. To tackle RGB-D SOD, we propose a novel joint learning and densely-cooperative fusion (*JL-DCF*) architecture that outperforms all existing deep learning-based techniques. Our method adopts the middle-fusion strategy mentioned above. However, different from previous works which conduct independent feature extraction from RGB and depth views, *JL-DCF* effectively extracts deep hierarchical features from RGB and depth inputs simultaneously, through a Siamese network (shared backbone). The underlying motivation is that, although depth and RGB images come from different modalities, they nevertheless share similar features/cues, such as strong figure-ground contrast [10,43,44], closure of object contours [20,53], and connectivity to image borders [36, 59]. This makes cross-modal transferring feasible, even for deep models. As evidenced in Fig. 1, a model trained on a single RGB modality, like DHS [38], can sometimes perform well in the depth view. Nevertheless, a similar model, like DSS [29], could also fail in the depth view without proper adaption or transferring.

To the best of our knowledge, the proposed *JL-DCF* scheme is *the first to leverage such transferability in deep models*, by treating a depth image as a special case of a color image and employing a shared CNN for both RGB and depth feature extraction. Additionally, we develop a densely-cooperative fusion strategy to reasonably combine the learned features of different modalities. This paper provides two main contributions:

- We introduce a general framework for RGB-D SOD, called *JL-DCF*, which consists of two sub-modules: joint learning and densely-cooperative fusion. The key features of these two components are their robustness and effectiveness, which will be beneficial for future modeling in related multi-modality tasks in computer vision. In particular, we advance the state-of-the-art (SOTA) by a significant average of $\sim$2% (F-measure score) across six challenging datasets.
- We present a thorough evaluation of 14 SOTA methods [4–6, 13, 18, 20, 25, 26, 34, 46, 49, 55, 60, 77], which is the largest-scale comparison in this field to date. Besides, we conduct a comprehensive ablation study, including using different input sources, learning schemes, and feature fusion strategies, to demonstrate the effectiveness of *JL-DCF*. Some interesting findings also encourage further research in this field.

2. Related Work

Traditional. The pioneering work for RGB-D SOD was produced by Niu *et al.* [43], who introduced disparity contrast and domain knowledge into stereoscopic photography to measure stereo saliency. After Niu's work, various hand-crafted features/hypotheses originally applied for RGB SOD were extended to RGB-D, such as center-surround difference [25, 34], contrast [10, 13, 44], background enclosure [20], center/boundary prior [10, 12, 36, 59], compactness [12, 13], or a combination of various saliency measures [55]. All the above models rely heavily on heuristic hand-crafted features, resulting in limited generalizability in complex scenarios.

Deep-based. Recent advances in this field have been obtained by using deep learning and CNNs. Qu *et al.* [49] first utilized a CNN to fuse different low-level saliency cues for judging the saliency confidence values of superpixels. Shigematsu *et al.* [53] extracted ten superpixel-based hand-crafted depth features capturing the background enclosure cue, depth contrast, and histogram distance. These features are fed to a CNN, whose output is shallowly fused with the RGB feature output to compute superpixel saliency.

A recent trend in this field is to exploit fully convolutional neural networks (FCNs) [52]. Chen *et al.* [4] proposed a bottom-up/top-down architecture [48], which progressively performs cross-modal complementarity-aware fusion in its top-down pathway. Han *et al.* [26] modified/extended the structure of the RGB-based deep neural network in order for it to be applicable for the depth view and then fused the deep representations of both views via a fully-connected layer. A three-stream attention-aware network was proposed in [5], which extracts hierarchical features from RGB and depth inputs through two separate streams. Features are then progressively combined and selected via attention-aware blocks in the third stream. A new multi-scale multi-path fusion network with cross-modal interactions was proposed in [6]. [40] and [31] formulated a four-channel input by concatenating RGB and depth. The input is later fed to a single-stream recurrent CNN and an FCN with short connections, respectively. [80] employed a subsidiary network to obtain depth features and used them to enhance the intermediate representation in an encoder-decoder architecture. Zhao *et al.* [77] proposed a model that generates a contrast-enhanced depth map, which is later used as a prior map for feature enhancement in subsequent fluid pyramid integration. Fan *et al.* [18] constructed a new RGB-D dataset called the Salient Person (SIP) dataset, and introduced a depth-depurator network to judge whether a depth map should be concatenated with the RGB image to formulate an input signal.

Generally, as summarized by previous literature [4, 77], the above approaches can be divided into three categories: (a) Early-fusion [18, 31, 40, 55], (b) late-fusion [26, 60] and (c) middle-fusion [4–6, 80]. Middle-fusion complements (a) and (b), since both feature-extraction and subsequent-fusion are handled by relatively deep CNNs. As a conse-

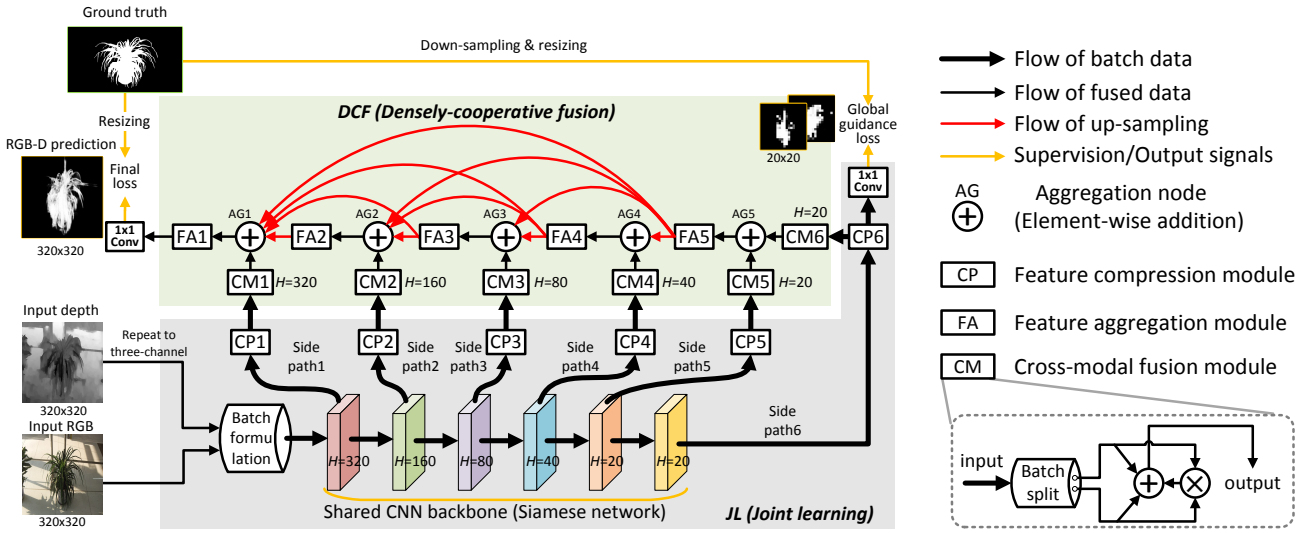

Figure 2: Block diagram of the proposed *JL-DCF* framework for RGB-D SOD. The JL (joint learning) component is shown in gray, while the DCF (densely-cooperative fusion) component is shown in light green. CP1∼CP6: Feature compression modules. FA1∼FA6: Feature aggregation modules. CM1∼CM6: Cross-modal fusion modules. "*H*" denotes the spatial size of output feature maps on a particular stage. See Section 3 for details.

quence, high-level concepts can be learnt from both modalities and complex integration rules can be mined. Besides, performing individual deep supervision for RGB and depth is straightforward. The proposed *JL-DCF* scheme falls into the middle-fusion strategy.

However, unlike the aforementioned methods [4–6, 80], where the two feature extraction streams are independent, we propose to utilize a Siamese architecture [11], where both the network architecture and weights are shared. This results in two major benefits: 1) Cross-modal knowledge-sharing becomes straightforward via joint learning; 2) The model parameters are largely reduced as only one shared network is needed, leading to facilitated learning.

3. Methodology

The overall architecture of the proposed *JL-DCF* is shown in Fig. 2. It follows the classic bottom-up/top-down strategy [48]. For illustrative purpose, Fig. 2 depicts an example backbone with six hierarchies that are common in the widely-used VGG [54] and ResNet [28]. The architecture consists of a JL component and a DCF component. The JL component conducts joint learning for the two modalities using a Siamese network. It aims to discover the commonality between these two views from a "model-based" perspective, since their information can be merged into model parameters via back-propagation. As seen in Fig. 2, the hierarchical features jointly learned by the backbone are then fed to the subsequent DCF component. DCF is dedicated to feature fusion and its layers are constructed in a densely-cooperative way. In this sense, the complementarity between RGB and depth modalities can be explored from a "feature-based" perspective. To perform cross-view feature fusion, in the DCF component, we elaborately design a cross-modal fusion module (CM module in Fig. 2). Details about *JL-DCF* will be given in the following sections.

3.1. Joint Learning (JL)

As shown in Fig. 2 (gray part), the inputs of the JL component are an RGB image together with its corresponding depth map. We first normalize the depth map into intervals [0, 255] and then convert it to a three-channel map through color mapping. In our implementation, we use the naive gray color mapping, which is equivalent to replicating the single channel map into three channels. Note that other color mapping [1] or transformations, like the mean used in [26], could also be considered for generating the three-channel representation. Next, the three-channel RGB image and transformed depth map are concatenated to formulate a *batch*, so that the subsequent CNN backbone can perform parallel processing. Note that, unlike previous early-fusion schemes aforementioned, which often concatenate the RGB and depth inputs in the 3^{rd} channel dimension, our scheme concatenates in the 4^{th} dimension, often called the batch dimension. For example, in our case a transformed $320 \times 320 \times 3$ depth and a $320 \times 320 \times 3$ RGB map will formulate a batch of size $320 \times 320 \times 3 \times 2$, rather than $320 \times 320 \times 6$.

The hierarchical features from the shared CNN backbone are then leveraged in a side-output way like [29]. Since the side-output features have varied resolutions and channel numbers (usually the deeper, the more channels), we first employ a set of CP modules (CP1∼CP6 in Fig. 2) to compress the side-output features to an identical, smaller number, denoted as k. We do this for the following two reasons:

(1) Using a large number of feature channels for subsequent decoding is memory and computationally expensive and (2) Unifying the number of feature channels facilitates various element-wise operations. Note that, here, the outputs from our CP modules are still batches, which are denoted as the thicker black arrows in Fig. 2.

Coarse localization can provide the basis for the following top-down refinement [48]. In addition, jointly learning the coarse localization guides the shared CNN to learn to extract independent hierarchical features from the RGB and depth views simultaneously. In order to enable the CNN backbone to coarsely locate the targets from both the RGB and depth views, we apply deep supervision to the JL component in the last hierarchy. To conduct this, as shown in Fig. 2, we add a $(1 \times 1, 1)$ convolutional layer on the CP6 module to achieve coarse prediction. The depth and RGB-associated outputs are supervised by the down-sampled ground truth map. The generated loss in this stage is called the global guidance loss $\mathcal{L}_g$.

3.2. Densely-cooperative Fusion (DCF)

As shown in Fig. 2 (light green part), the output batch features from the CP modules contain depth and RGB information. They are fed to the DCF component, which can be deemed a decoder that performs multi-scale cross-modal fusion. Firstly, we design a CM (cross-modal fusion) module to split and then merge the batch features (Fig. 2, bottom-right). This module first splits the batch data and then conducts "addition and multiplication" feature fusion, which we call *cooperative fusion*. Mathematically, let a batch feature be denoted by $\{X_{rgb}, X_d\}$, where X_{rgb}, X_d represent the RGB and depth parts, each with k channels, respectively. The CM module conducts the fusion as:

$$CM(\{X_{rgb}, X_d\}) = X_{rgb} \oplus X_d \oplus (X_{rgb} \otimes X_d), \quad (1)$$

where "$\oplus$" and "$\otimes$" denote element-wise addition and multiplication. The blended features output from the CM modules are still made up of k channels. Compared to element-wise addition "$\oplus$", which exploits *feature complementarity*, element-wise multiplication "$\otimes$" puts more emphasis on *commonality*. These two properties are generally important in cross-view fusion.

One may argue that such a CM module could be replaced by channel concatenation, which generates $2k$-channel concatenated features. However, we find such a choice tends to result in the learning process being trapped in a local optimum, where it becomes biased towards only RGB information. The reason seems to be that the channel concatenation does indeed involve feature selection rather than explicit feature fusion. This leads to degraded learning outcomes, where only RGB features dominate the final prediction. Note that, as will be shown in Section 4.4, solely using RGB input can also achieve fairly good performance

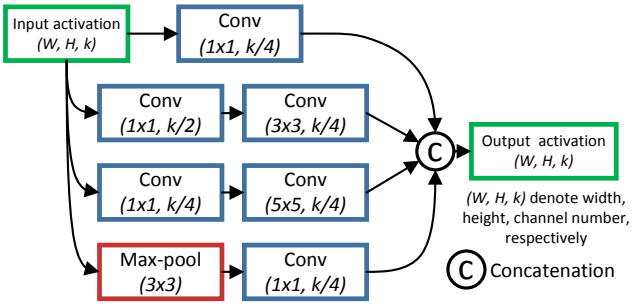

Figure 3: Inception structure used for the FA modules in Fig. 2. All convolutional layers and max-pooling layers have stride 1, therefore maintaining spatial feature sizes. Unlike the original Inception module [58], we adapt it to have the same input/output channel number k.

in the proposed framework. Comparisons between our CM modules and concatenation will be given in Section 4.4.

As shown in Fig. 2, the fused features from CM1∼CM6 are fed to a decoder augmented with a dense connection [30]. Using the dense connection promotes the blending of depth and RGB features at various scales. Therefore, unlike the traditional UNet-like decoder [50], an aggregation module FA takes inputs from all levels deeper than itself. Specifically, FA denotes a feature aggregation module performing non-linear aggregation. To this end, we use the Inception module [58] shown in Fig. 3, which performs multi-level convolutions with filter size $1 \times 1, 3 \times 3, 5 \times 5$, and max-pooling. Note that the FA module in our framework is flexible. Other modules may also be considered in the future to improve the performance.

Finally, the FA module with the finest features is denoted as FA1, whose output is then fed to a $(1 \times 1, 1)$ convolutional layer to generate the final activation and then ultimately the saliency map. This final prediction is supervised by the resized ground truth (GT) map during training. We denote the loss generated in this stage as $\mathcal{L}_f$.

3.3. Loss Function

The overall loss function of our scheme is composed of the global guidance loss $\mathcal{L}_g$ and final loss $\mathcal{L}_f$. Assume that G denotes supervision from the ground truth, S^c_{rgb} and S^c_d denote the coarse prediction maps contained in the batch after module CP6, and S^f is the final prediction after module FA1. The overall loss function is defined as:

$$\mathcal{L}_{total} = \mathcal{L}_f(S^f, G) + \lambda \sum_{x \in \{rgb, d\}} \mathcal{L}_g(S^c_x, G), \quad (2)$$

where λ balances the emphasis of global guidance, and we adopt the widely used cross-entropy loss for $\mathcal{L}_g$ and $\mathcal{L}_f$ as:

$$\mathcal{L}(S, G) = -\sum_i [G_i \log(S_i) + (1 - G_i) \log(1 - S_i)], \quad (3)$$

where i denotes pixel index, and $S \in \{S^c_{rgb}, S^c_d, S^f\}$.

Table 1: Quantitative measures: S-measure (S_α) [16], max F-measure ($F_\beta^{\max}$) [3], max E-measure ($E_\phi^{\max}$) [17] and MAE (M) [45] of SOTA methods and the proposed *JL-DCF* on six RGB-D datasets. The best performance is highlighted in **bold**.

	Metric	ACSD [34]	LBE [20]	DCMC [13]	MDSF [55]	SE [25]	DF [49]	AFNet [60]	CTMF [26]	MMCI [6]	PCF [4]	TANet [5]	CPFP [77]	DMRA [46]	D3Net [18]	*JL-DCF* Ours
NJU2K [34]	$S_\alpha \uparrow$	0.699	0.695	0.686	0.748	0.664	0.763	0.772	0.849	0.858	0.877	0.878	0.879	0.886	0.895	**0.903**
	$F_\beta^{\max} \uparrow$	0.711	0.748	0.715	0.775	0.748	0.804	0.775	0.845	0.852	0.872	0.874	0.877	0.886	0.889	**0.903**
	$E_\phi^{\max} \uparrow$	0.803	0.803	0.799	0.838	0.813	0.864	0.853	0.913	0.915	0.924	0.925	0.926	0.927	0.932	**0.944**
	$M \downarrow$	0.202	0.153	0.172	0.157	0.169	0.141	0.100	0.085	0.079	0.059	0.060	0.053	0.051	0.051	**0.043**
NLPR [44]	$S_\alpha \uparrow$	0.673	0.762	0.724	0.805	0.756	0.802	0.799	0.860	0.856	0.874	0.886	0.888	0.899	0.906	**0.925**
	$F_\beta^{\max} \uparrow$	0.607	0.745	0.648	0.793	0.713	0.778	0.771	0.825	0.815	0.841	0.863	0.867	0.879	0.885	**0.916**
	$E_\phi^{\max} \uparrow$	0.780	0.855	0.793	0.885	0.847	0.880	0.879	0.929	0.913	0.925	0.941	0.932	0.947	0.946	**0.962**
	$M \downarrow$	0.179	0.081	0.117	0.095	0.091	0.085	0.058	0.056	0.059	0.044	0.041	0.036	0.031	0.034	**0.022**
STERE [43]	$S_\alpha \uparrow$	0.692	0.660	0.731	0.728	0.708	0.757	0.825	0.848	0.873	0.875	0.871	0.879	0.886	0.891	**0.905**
	$F_\beta^{\max} \uparrow$	0.669	0.633	0.740	0.719	0.755	0.757	0.823	0.831	0.863	0.860	0.861	0.874	0.886	0.881	**0.901**
	$E_\phi^{\max} \uparrow$	0.806	0.787	0.819	0.809	0.846	0.847	0.887	0.912	0.927	0.925	0.923	0.925	0.938	0.930	**0.946**
	$M \downarrow$	0.200	0.250	0.148	0.176	0.143	0.141	0.075	0.086	0.068	0.064	0.060	0.051	0.047	0.054	**0.042**
RGBD135 [10]	$S_\alpha \uparrow$	0.728	0.703	0.707	0.741	0.741	0.752	0.770	0.863	0.848	0.842	0.858	0.872	0.900	0.904	**0.929**
	$F_\beta^{\max} \uparrow$	0.756	0.788	0.666	0.746	0.741	0.766	0.728	0.844	0.822	0.804	0.827	0.846	0.888	0.885	**0.919**
	$E_\phi^{\max} \uparrow$	0.850	0.890	0.773	0.851	0.856	0.870	0.881	0.932	0.928	0.893	0.910	0.923	0.943	0.946	**0.968**
	$M \downarrow$	0.169	0.208	0.111	0.122	0.090	0.093	0.068	0.055	0.065	0.049	0.046	0.038	0.030	0.030	**0.022**
LFSD [35]	$S_\alpha \uparrow$	0.727	0.729	0.746	0.694	0.692	0.783	0.730	0.788	0.779	0.786	0.794	0.820	0.839	0.824	**0.854**
	$F_\beta^{\max} \uparrow$	0.763	0.722	0.813	0.779	0.786	0.813	0.740	0.787	0.767	0.775	0.792	0.821	0.852	0.815	**0.862**
	$E_\phi^{\max} \uparrow$	0.829	0.797	0.849	0.819	0.832	0.857	0.807	0.857	0.831	0.827	0.840	0.864	0.893	0.856	**0.893**
	$M \downarrow$	0.195	0.214	0.162	0.197	0.174	0.146	0.141	0.127	0.139	0.119	0.118	0.095	0.083	0.106	**0.078**
SIP [18]	$S_\alpha \uparrow$	0.732	0.727	0.683	0.717	0.628	0.653	0.720	0.716	0.833	0.842	0.835	0.850	0.806	0.864	**0.879**
	$F_\beta^{\max} \uparrow$	0.763	0.751	0.618	0.698	0.661	0.657	0.712	0.694	0.818	0.838	0.830	0.851	0.821	0.862	**0.885**
	$E_\phi^{\max} \uparrow$	0.838	0.853	0.743	0.798	0.771	0.759	0.819	0.829	0.897	0.901	0.895	0.903	0.875	0.910	**0.923**
	$M \downarrow$	0.172	0.200	0.186	0.167	0.164	0.185	0.118	0.139	0.086	0.071	0.075	0.064	0.085	0.063	**0.051**

4. Experiments

4.1. Datasets and Metrics

Experiments are conducted on six public RGB-D benchmark datasets: NJU2K [34] (2000 samples), NLPR [44] (1000 samples), STERE [43] (1000 samples), RGBD135 [10] (135 samples), LFSD [35] (100 samples), and SIP [18] (929 samples). Following [77], we choose the same 700 samples from NLPR and 1500 samples from NJU2K to train our algorithms. The remaining samples are used for testing. For fair comparisons, we apply the model trained on this training set to other datasets. For evaluation, we adopt four widely used metrics, namely S-measure (S_α) [16, 77], maximum F-measure ($F_\beta^{\max}$) [3, 29], maximum E-measure ($E_\phi^{\max}$) [17, 18], and MAE (M) [3, 45]. The definitions for these metrics are omitted here and readers are referred to the related papers. Note that, since the E-measure metric was originally proposed in [17] for evaluating binary maps, to extend it for comparing a non-binary saliency map against a binary ground truth map, we follow a similar strategy to $F_\beta^{\max}$. Specifically, we first binarize a saliency map into a series of foreground maps using all possible threshold values in [0, 255], and then report the maximum E-measure among them.

4.2. Implementation Details

The proposed *JL-DCF* scheme is generally independent from the network backbone. In this work, we implement two versions of *JL-DCF* based on VGG-16 [54] and ResNet-101 [28], respectively. We fix the input size of the network as $320 \times 320 \times 3$. Simple gray color mapping is adopted to convert a depth map into a three-channel map.

VGG-16 configuration: For the VGG-16 with the fully-connected layers removed and having 13 convolutional layers, the *side path1*$\sim$*path6* are successively connected to *conv1_2*, *conv2_2*, *conv3_3*, *conv4_3*, *conv5_3*, and *pool5*. Inspired by [29], we add two extra convolutional layers into *side path1*$\sim$*path6*. To augment the resolution of the coarsest feature maps from *side path6*, while at the same time preserving the receptive field, we let *pool5* have a stride of 1 and instead use dilated convolution [7] with a rate of 2 for the two extra side convolutional layers. In general, the coarsest features produced by our final modified VGG-16 backbone have a spatial size of 20×20, as shown in Fig. 2.

ResNet-101 configuration: Similar to the VGG-16 case above, the spatial size of the coarsest features produced by our modified ResNet-101 backbone is also 20×20. As the first convolutional layer of ResNet already has a stride of 2, the features from the shallowest level have a spatial size of 160×160. To obtain the full size (320×320) features without trivial up-sampling, we borrow the *conv1_1* and *conv1_2* layers from VGG-16 for feature extraction. *Side path1*$\sim$*path6* are connected to *conv1_2*, and *conv1*, *res2c*, *res3b3*, *res4b22*, *res5c* of the ResNet-101, respectively. We also change the stride of the *res5a* block from 2 to 1, but subsequently use dilated convolution with rate 2.

Decoder configuration: All CP modules in Fig. 2 are 3×3 convolutions with $k = 64$ filters, and all FA modules

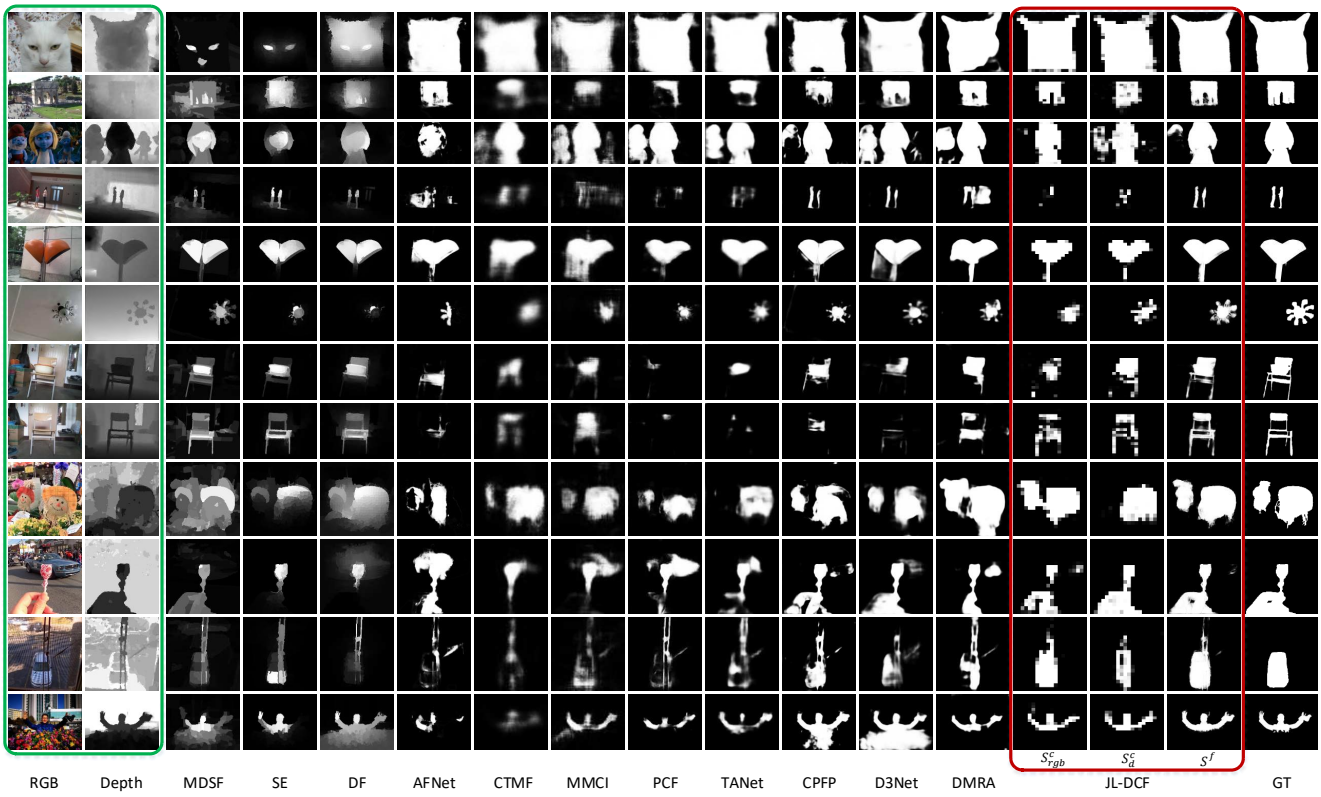

Figure 4: Visual comparisons of *JL-DCF* with SOTA RGB-D saliency models. The jointly learned coarse prediction maps (S^c_{rgb} and S^c_d) from RGB and depth are also shown together with the final maps (S^f) of *JL-DCF*.

are Inception modules. Up-sampling is achieved by simple bilinear interpolation. As depicted in Fig. 2, to align the feature sizes in the decoder, the output from an FA module is up-sampled by various factors. In an extreme case, the output from FA5 is up-sampled by a factor of 2, 4, 8, and 16. The final output from FA1 has a spatial size of 320×320, which is identical to the initial input.

Training setup: We implement *JL-DCF* on Caffe [33]. During training, the backbone [28, 54] is initialized by the pre-trained parameters of DSS [29], and other layers are randomly initialized. We fine-tune the entire network through end-to-end joint learning. Training data is augmented by mirror reflection to generate double the amount of data. The momentum parameter is set as 0.99, the learning rate is set to $lr = 10^{-9}$, and the weight decay is 0.0005. The weight λ in Eq. (2) is set as 256 (=16^2) to balance the loss between the low- and high-resolution predictions. Stochastic Gradient Descent learning is adopted and accelerated by an NVIDIA 1080Ti GPU. The training time is about 20 hours/18 hours for 40 epochs under the ResNet-101/VGG-16 configuration.

4.3. Comparisons to SOTAs

We compare *JL-DCF* (ResNet configuration) with 14 SOTA methods. Among the competitors, DF [49], AFNet [60], CTMF [26], MMCI [6], PCF [4], TANet [5], CPFP [77], D3Net [18], DMRA [46] are recent deep learning-based methods, while ACSD [34], LBE [20], DCMC [13], MDSF [55], SE [25] are traditional techniques using various hand-crafted features/hypotheses. Quantitative results are shown in Table 1. Notable performance gains of *JL-DCF* over existing and recently proposed techniques, like CPFP [77], D3Net [18] and DMRA [46], can be seen in all four metrics. This validates the consistent effectiveness of *JL-DCF* and its generalizability. Some visual examples are shown in Fig. 4. *JL-DCF* appears to be more effective at utilizing depth information for cross-modal compensation, making it better for detecting target objects in the RGB-D mode. Additionally, the deeply-supervised coarse predictions are listed in Fig. 4. One can see that they provide basic object localization support for the subsequent cross-modal refinement, and our densely-cooperative fusion architecture learns an adaptive and "image-dependent" way of fusing such support with the hierarchical multi-view features. This proves that the fusion process does not degrade in either of the two views (RGB or depth), leading to boosted performance after fusion.

4.4. Ablation Studies

We conduct thorough ablation studies by removing or replacing components from the full implementation of *JL-DCF*. We set the ResNet version of *JL-DCF* as

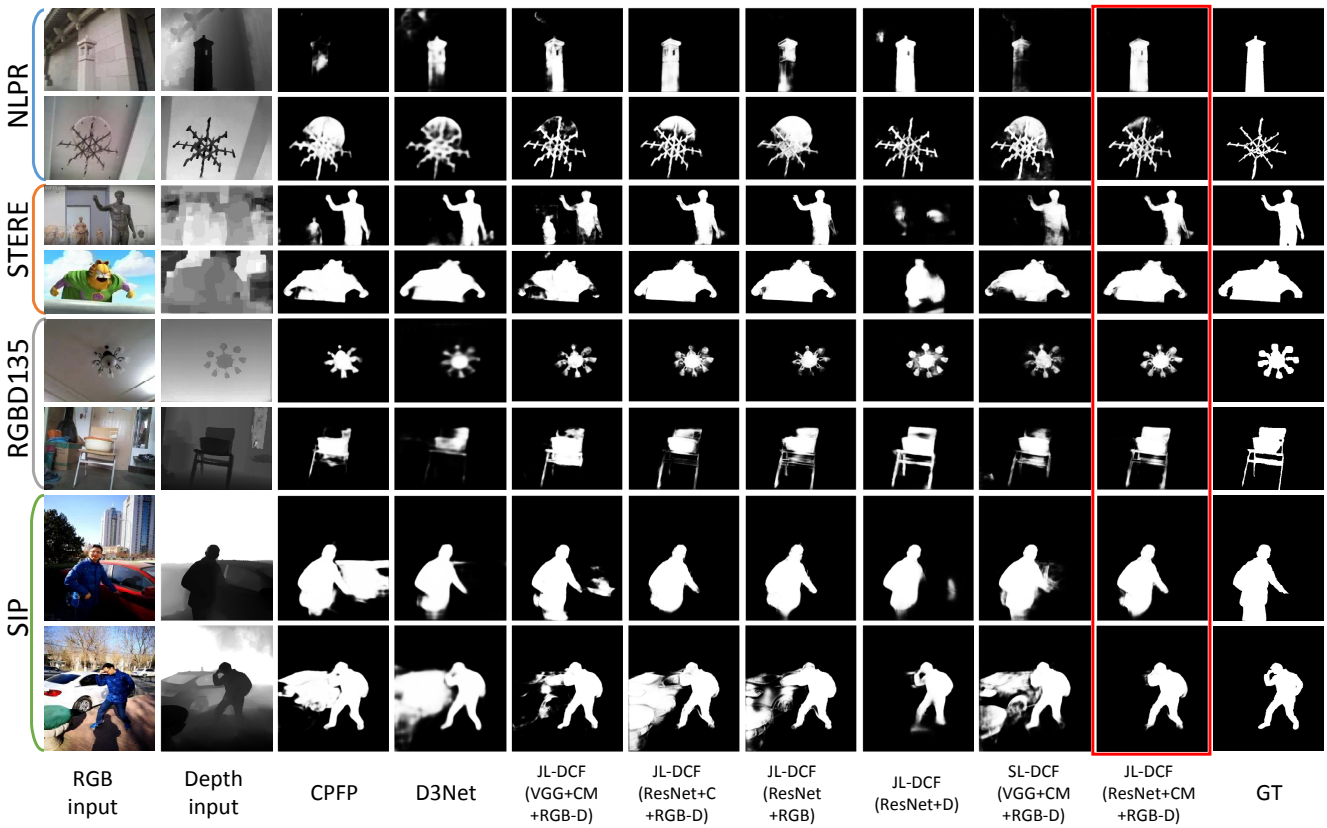

Figure 5: Visual examples from NLPR, STERE, RGB135, SIP datasets for ablation studies. Generally, the full implementation of *JL-DCF* (ResNet+CM+RGB-D, highlighted in the red box) achieves the closest results to the ground truth.

reference, and then compare various ablation experiments to it. We denote this reference version as "*JL-DCF* (ResNet+CM+RGB-D)", where "CM" refers to the usage of CM modules and "RGB-D" refers to both RGB and depth inputs.

Firstly, to compare different backbones, a version "*JL-DCF* (VGG+CM+RGB-D)" is trained by replacing the ResNet backbone with VGG, while keeping other settings unchanged. To validate the effectiveness of the adopted cooperative fusion modules, we train another version "*JL-DCF* (ResNet+C+RGB-D)", by replacing the CM modules with a concatenation operation. To demonstrate the effectiveness of combining RGB and depth, we train two versions "*JL-DCF* (ResNet+RGB)" and "*JL-DCF* (ResNet+D)" respectively, where all the batch-related operations (such as CM modules) in Fig. 2 are replaced with identity mappings, while all the other settings, including the dense decoder and deep supervision, are kept unchanged. Note that this validation is important to show that our network has learned complementary information by fusing RGB and depth. Lastly, to illustrate the benefit of joint learning, we train a scheme "SL-DCF (VGG+CM+RGB-D)" using two separate backbones for RGB and depth. "SL" stands for "Separate Learning", in contrast to the proposed "Joint Learning". In this test, we adopt VGG-16, which is smaller, since using two separate backbones leads to almost twice the overall model size.

Quantitative comparisons for various metrics are shown in Table 2. Two SOTA methods CPFP [77] and D3Net [18] are listed for reference. Fig. 5 shows visual ablation comparisons. Five different observations can be made:

ResNet-101 *vs.* **VGG-16:** From the comparison between columns "A" and "B" in Table 2, the superiority of the ResNet backbone over VGG-16 is evident, which is consistent with previous works. Note that the VGG version of our scheme still outperforms the leading methods CPFP (VGG-16 backbone) and D3Net (ResNet backbone).

Effectiveness of CM modules: Comparing columns "A" and "C" demonstrates that changing the CM modules into concatenation operations leads to a certain amount of degeneration. The underlying reason is that the whole network tends to bias its learning towards only RGB information, while ignoring depth, since it is able to achieve fairly good results (column "D") by doing so on the most datasets. Although concatenation is a popular way to fuse features, the learning may become easily trapped without appropriate guidance. In contrast, our CM modules perform the "explicit fusion operation" across RGB and depth modalities.

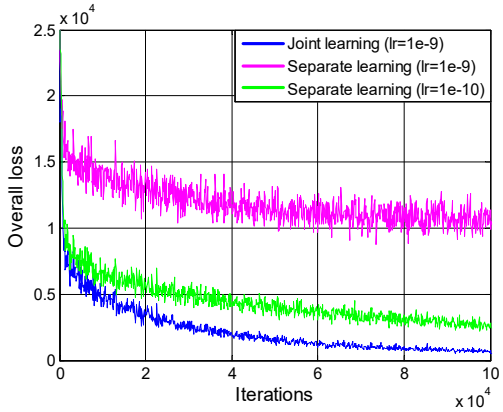

Figure 6: Learning curve comparison between joint learning (*JL-DCF*) and separate learning (SL-DCF).

Combining RGB and depth: The effectiveness of combining RGB and depth for boosting the performance is clearly validated by the consistent improvement over most datasets (compare column "A" with columns "D" and "E"). The only exception is on STERE [43], with the reason being that the quality of depth maps in this dataset is much worse compared to other datasets. Visual examples are shown in Fig. 5, in the 3^{rd} and 4^{th} rows. We find that many depth maps from STERE are too coarse and have very inaccurate object boundaries, misaligning with the true objects. Absorbing such unreliable depth information may, in turn, degrade the performance. Quantitative evidence can be seen in Table 2, column "E" (STERE dataset), where solely using depth cues achieves much worse performance (about 16%/20% lower on $S_\alpha/F_\beta^{\max}$ comparing to RGB) than on other datasets.

RGB only $vs.$ **depth only:** The comparison between columns "D" and "E" in Table 2 proves that using RGB data for saliency estimation is superior to using depth in most cases, indicating that the RGB view is generally more informative. However, using depth information achieves better results than RGB on SIP [18] and RGBD135 [10], as visualized in Fig. 5. This implies that the depth maps from the two datasets are of relatively good quality.

Efficiency of JL component: Existing models usually use separate learning approaches to extract features from RGB and depth data, respectively. In contrast, our *JL-DCF* adopts a joint learning strategy to obtain the features from an RGB and depth map simultaneously. We compare the two learning strategies and find that using separate learning (two separate backbones) is likely to increase the training difficulties. Fig. 6 shows typical learning curves for such a case. In the separate learning setting, where the initial learning rate is $lr = 10^{-9}$, the network is easily trapped in a local optimum with high loss, while the joint learning setting (shared network) can converge nicely. Further, for separate learning, if the learning rate is set to $lr = 10^{-10}$, the learning process is rescued from local oscillation but converges slowly compared to our joint learning strategy. As shown in columns "B" and "F" in Table 2, the resulting converged model after 40 epochs achieves worse performance than *JL-DCF*, namely 1.1%/1.76% overall drop on $S_\alpha/F_\beta^{\max}$. We attribute the better performance of *JL-DCF* to its joint learning from both RGB and depth data.

Table 2: Quantitative evaluation for ablation studies described in Section 4.4. For different configurations, "A": JL-DCF (ResNet+CM+RGB-D), "B": JL-DCF (VGG+CM+RGB-D), "C": JL-DCF (ResNet+C+RGB-D), "D": JL-DCF (ResNet+RGB), "E": JL-DCF (ResNet+D), "F": SL-DCF (VGG+CM+RGB-D).

	Metric	CPFP	D3Net	A	B	C	D	E	F
NJU2K [34]	$S_\alpha \uparrow$	.878	.895	**.903**	.897	.900	.895	.865	.886
	$F_\beta^{\max} \uparrow$	.877	.889	**.903**	.899	.898	.892	.863	.883
	$E_\phi^{\max} \uparrow$	.926	.932	**.944**	.939	.937	.937	.916	.929
	$M \downarrow$	.053	.051	**.043**	.044	.045	.046	.063	.053
NLPR [44]	$S_\alpha \uparrow$	.888	.906	**.925**	.920	.924	.922	.873	.901
	$F_\beta^{\max} \uparrow$	.868	.885	**.916**	.907	.914	.909	.843	.881
	$E_\phi^{\max} \uparrow$	.932	.946	**.962**	.959	.961	.957	.930	.946
	$M \downarrow$	.036	.034	**.022**	.026	.023	.025	.041	.033
STERE [43]	$S_\alpha \uparrow$	.879	.891	.905	.894	.906	**.909**	.744	.886
	$F_\beta^{\max} \uparrow$	.874	.881	**.901**	.889	.899	.901	.708	.876
	$E_\phi^{\max} \uparrow$	.925	.930	**.946**	.938	.945	.946	.834	.931
	$M \downarrow$	.051	.054	.042	.046	.041	**.038**	.110	.053
RGBD135 [10]	$S_\alpha \uparrow$	.872	.904	**.929**	.913	.916	.903	.918	.893
	$F_\beta^{\max} \uparrow$	.846	.885	**.919**	.905	.906	.894	.906	.876
	$E_\phi^{\max} \uparrow$	.923	.946	**.968**	.955	.957	.947	.967	.950
	$M \downarrow$	.038	.030	**.022**	.026	.025	.027	.027	.033
LFSD [35]	$S_\alpha \uparrow$	.820	.832	**.854**	.833	.852	.845	.752	.826
	$F_\beta^{\max} \uparrow$	.821	.819	**.862**	.840	.854	.846	.764	.828
	$E_\phi^{\max} \uparrow$	.864	.864	**.893**	.877	.893	.889	.816	.864
	$M \downarrow$	.095	.099	**.078**	.091	.078	.083	.126	.101
SIP [18]	$S_\alpha \uparrow$	.850	.864	**.879**	.866	.870	.855	.872	.865
	$F_\beta^{\max} \uparrow$	.851	.862	**.885**	.873	.873	.857	.877	.863
	$E_\phi^{\max} \uparrow$	.903	.910	**.923**	.916	.916	.908	.920	.913
	$M \downarrow$	.064	.063	**.051**	.056	.055	.061	.056	.061

5. Conclusion

We present a novel framework for RGB-D based SOD, named *JL-DCF*, which is based on joint learning and densely-cooperative fusion. Experimental results show the feasibility of learning a shared network for salient object localization in RGB and depth views, simultaneously, to achieve accurate prediction. Moreover, the densely-cooperative fusion strategy employed is effective for exploiting cross-modal complementarity. *JL-DCF* shows superior performance against SOTAs on six benchmark datasets and is supported by comprehensive ablation studies. Our framework is quite flexible and general, and its inner modules could be replaced by their counterparts for further improvement.

Acknowledgments. This work was supported by the NSFC, under No. 61703077, 61773270, 61971005, the Fundamental Research Funds for the Central Universities No. YJ201755, and the Sichuan Science and Technology Major Projects (2018GZDZX0029).

References

[1] Jamil Al Azzeh, Hussein Alhatamleh, Ziad A Alqadi, and Mohammad Khalil Abuzalata. Creating a color map to be used to convert a gray image to color image. *International Journal of Computer Applications*, 153(2):31–34, 2016.

[2] Ali Borji, Ming-Ming Cheng, Qibin Hou, Huaizu Jiang, and Jia Li. Salient object detection: A survey. *CVM*, pages 1–34, 2019.

[3] Ali Borji, Ming-Ming Cheng, Huaizu Jiang, and Jia Li. Salient object detection: A benchmark. *IEEE TIP*, 24(12):5706–5722, 2015.

[4] Hao Chen and Youfu Li. Progressively complementarity-aware fusion network for rgb-d salient object detection. In *CVPR*, pages 3051–3060, 2018.

[5] Hao Chen and Youfu Li. Three-stream attention-aware network for rgb-d salient object detection. *IEEE TIP*, 28(6):2825–2835, 2019.

[6] Hao Chen, Youfu Li, and Dan Su. Multi-modal fusion network with multi-scale multi-path and cross-modal interactions for rgb-d salient object detection. *Pattern Recognition*, 86:376–385, 2019.

[7] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE TPAMI*, 40(4):834–848, 2017.

[8] Tao Chen, Ming-Ming Cheng, Ping Tan, Ariel Shamir, and Shi-Min Hu. Sketch2photo: Internet image montage. *ACM TOG*, 28(5):1–10, 2006.

[9] Ming-Ming Cheng, Niloy J Mitra, Xiaolei Huang, Philip HS Torr, and Shi-Min Hu. Global contrast based salient region detection. *IEEE TPAMI*, 37(3):569–582, 2015.

[10] Yupeng Cheng, Huazhu Fu, Xingxing Wei, Jiangjian Xiao, and Xiaochun Cao. Depth enhanced saliency detection method. In *Int'l Conference on Internet Multimedia Computing and Service*. ACM, 2014.

[11] Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *CVPR*, pages 539–546, 2005.

[12] Runmin Cong, Jianjun Lei, Huazhu Fu, Junhui Hou, Qingming Huang, and Sam Kwong. Going from rgb to rgbd saliency: A depth-guided transformation model. *IEEE TCYB*, 2019.

[13] Runmin Cong, Jianjun Lei, Changqing Zhang, Qingming Huang, Xiaochun Cao, and Chunping Hou. Saliency detection for stereoscopic images based on depth confidence analysis and multiple cues fusion. *IEEE SPL*, 23(6):819–823, 2016.

[14] Yuanyuan Ding, Jing Xiao, and Jingyi Yu. Importance filtering for image retargeting. In *CVPR*, pages 89–96, 2011.

[15] Deng-Ping Fan, Ming-Ming Cheng, Jiang-Jiang Liu, Shang-Hua Gao, Qibin Hou, and Ali Borji. Salient objects in clutter: Bringing salient object detection to the foreground. In *ECCV*, pages 196–212, 2018.

[16] Deng-Ping Fan, Ming-Ming Cheng, Yun Liu, Tao Li, and Ali Borji. Structure-measure: A New Way to Evaluate Foreground Maps. In *ICCV*, pages 4548–4557, 2017.

[17] Deng-Ping Fan, Cheng Gong, Yang Cao, Bo Ren, Ming-Ming Cheng, and Ali Borji. Enhanced-alignment measure for binary foreground map evaluation. In *IJCAI*, pages 698–704, 2018.

[18] Deng-Ping Fan, Zheng Lin, Zhao Zhang, Menglong Zhu, and Ming-Ming Cheng. Rethinking RGB-D salient object detection: Models, datasets, and large-scale benchmarks. *IEEE TNNLS*, 2020.

[19] Deng-Ping Fan, Wenguan Wang, Ming-Ming Cheng, and Jianbing Shen. Shifting more attention to video salient object detection. In *CVPR*, pages 8554–8564, 2019.

[20] David Feng, Nick Barnes, Shaodi You, and Chris McCarthy. Local background enclosure for rgb-d salient object detection. In *CVPR*, pages 2343–2350, 2016.

[21] Mengyang Feng, Huchuan Lu, and Errui Ding. Attentive feedback network for boundary-aware salient object detection. In *CVPR*, pages 1623–1632, 2019.

[22] Yuan Gao, Miaojing Shi, Dacheng Tao, and Chao Xu. Database saliency for fast image retrieval. *IEEE TMM*, 17(3):359–369, 2015.

[23] Stas Goferman, Lihi Zelnik-Manor, and Ayellet Tal. Context-aware saliency detection. In *CVPR*, pages 2376–2383, 2010.

[24] Chenlei Guo and Liming Zhang. A novel multiresolution spatiotemporal saliency detection model and its applications in image and video compression. *IEEE TIP*, 19(1):185–198, 2010.

[25] Jingfan Guo, Tongwei Ren, and Jia Bei. Salient object detection for rgb-d image via saliency evolution. In *ICME*, pages 1–6, 2016.

[26] Junwei Han, Hao Chen, Nian Liu, Chenggang Yan, and Xuelong Li. Cnns-based rgb-d saliency detection via cross-view transfer and multiview fusion. *IEEE TCYB*, 48(11):3171–3183, 2017.

[27] Junwei Han, King Ngi Ngan, Mingjing Li, and Hong-Jiang Zhang. Unsupervised extraction of visual attention objects in color images. *IEEE TCSVT*, 16(1):141–145, 2006.

[28] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[29] Qibin Hou, Ming-Ming Cheng, Xiaowei Hu, Ali Borji, Zhuowen Tu, and Philip HS Torr. Deeply supervised salient object detection with short connections. *IEEE TPAMI*, 41(4):815–828, 2019.

[30] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *CVPR*, pages 4700–4708, 2017.

[31] Posheng Huang, Chin-Han Shen, and Hsu-Feng Hsiao. Rgbd salient object detection using spatially coherent deep learning framework. In *International Conference on Digital Signal Processing*, pages 1–5, 2018.

[32] Koteswar Rao Jerripothula, Jianfei Cai, and Junsong Yuan. Image co-segmentation via saliency co-fusion. *IEEE TMM*, 18(9):1896–1909, 2016.

[33] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In *ACM MM*, pages 675–678, 2014.

[34] Ran Ju, Ling Ge, Wenjing Geng, Tongwei Ren, and Gangshan Wu. Depth saliency based on anisotropic center-surround difference. In *ICIP*, pages 1115–1119, 2014.

[35] Nianyi Li, Jinwei Ye, Yu Ji, Haibin Ling, and Jingyi Yu. Saliency detection on light field. In *CVPR*, pages 2806–2813, 2014.

[36] Fangfang Liang, Lijuan Duan, Wei Ma, Yuanhua Qiao, Zhi Cai, and Laiyun Qing. Stereoscopic saliency model using contrast and depth-guided-background prior. *Neurocomputing*, 275:2227–2238, 2018.

[37] Guanghai Liu and Dengping Fan. A model of visual attention for natural image retrieval. In *ISCC-C*, pages 728–733, 2013.

[38] Nian Liu and Junwei Han. Dhsnet: Deep hierarchical saliency network for salient object detection. In *CVPR*, pages 678–686, 2016.

[39] Zhi Liu, Ran Shi, Liquan Shen, Yinzhu Xue, King Ngi Ngan, and Zhaoyang Zhang. Unsupervised salient object segmentation based on kernel density estimation and two-phase graph cut. *IEEE TMM*, 14(4):1275–1289, 2012.

[40] Zhengyi Liu, Song Shi, Quntao Duan, Wei Zhang, and Peng Zhao. Salient object detection for rgb-d image by single stream recurrent convolution neural network. *Neurocomputing*, 363:46–57, 2019.

[41] Yu-Fei Ma, Xian-Sheng Hua, Lie Lu, and Hong-Jiang Zhang. A generic framework of user attention model and its application in video summarization. *IEEE TMM*, 7(5):907–919, 2005.

[42] Luca Marchesotti, Claudio Cifarelli, and Gabriela Csurka. A framework for visual saliency detection with applications to image thumbnailing. In *ICCV*, pages 2232–2239, 2009.

[43] Yuzhen Niu, Yujie Geng, Xueqing Li, and Feng Liu. Leveraging stereopsis for saliency analysis. In *CVPR*, pages 454–461, 2012.

[44] Houwen Peng, Bing Li, Weihua Xiong, Weiming Hu, and Rongrong Ji. Rgbd salient object detection: A benchmark and algorithms. In *ECCV*, pages 92–109, 2014.

[45] Federico Perazzi, Philipp Krähenbühl, Yael Pritch, and Alexander Hornung. Saliency filters: Contrast based filtering for salient region detection. In *CVPR*, pages 733–740, 2012.

[46] Yongri Piao, Wei Ji, Jingjing Li, Miao Zhang, and Huchuan Lu. Depth-induced multi-scale recurrent attention network for saliency detection. In *ICCV*, pages 7254–7263, 2019.

[47] Yongri Piao, Zhengkun Rong, Miao Zhang, Xiao Li, and Huchuan Lu. Deep light-field-driven saliency detection from a single view. In *IJCAI*, pages 904–911, 2019.

[48] Pedro O Pinheiro, Tsung-Yi Lin, Ronan Collobert, and Piotr Dollár. Learning to refine object segments. In *ECCV*, pages 75–91, 2016.

[49] Liangqiong Qu, Shengfeng He, Jiawei Zhang, Jiandong Tian, Yandong Tang, and Qingxiong Yang. Rgbd salient object detection via deep fusion. *IEEE TIP*, 26(5):2274–2285, 2017.

[50] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, pages 234–241, 2015.

[51] Ueli Rutishauser, Dirk Walther, Christof Koch, and Pietro Perona. Is bottom-up attention useful for object recognition. In *CVPR*, pages II–II, 2004.

[52] Evan Shelhamer, Jonathan Long, and Trevor Darrell. Fully convolutional networks for semantic segmentation. *IEEE TPAMI*, 39(4):640–651, 2017.

[53] Riku Shigematsu, David Feng, Shaodi You, and Nick Barnes. Learning rgb-d salient object detection using background enclosure, depth contrast, and top-down features. In *ICCVW*, pages 2749–2757, 2017.

[54] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

[55] Hangke Song, Zhi Liu, Huan Du, Guangling Sun, Olivier Le Meur, and Tongwei Ren. Depth-aware salient object detection and segmentation via multiscale discriminative saliency fusion and bootstrap learning. *IEEE TIP*, 26(9):4204–4216, 2017.

[56] Hongmei Song, Wenguan Wang, Sanyuan Zhao, Jianbing Shen, and Kin-Man Lam. Pyramid dilated deeper convlstm for video salient object detection. In *ECCV*, pages 715–731, 2018.

[57] Fred Stentiford. Attention based auto image cropping. In *ICVS*, 2007.

[58] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, pages 1–9, 2015.

[59] Anzhi Wang and Minghui Wang. Rgb-d salient object detection via minimum barrier distance transform and saliency fusion. *IEEE SPL*, 24(5):663–667, 2017.

[60] Ningning Wang and Xiaojin Gong. Adaptive fusion for rgb-d salient object detection. *IEEE Access*, 7:55277–55284, 2019.

[61] Wenguan Wang, Qiuxia Lai, Huazhu Fu, Jianbing Shen, and Haibin Ling. Salient object detection in the deep learning era: An in-depth survey. *arXiv preprint arXiv:1904.09146*, 2019.

[62] Wenguan Wang, Xiankai Lu, Jianbing Shen, David J Crandall, and Ling Shao. Zero-shot video object segmentation via attentive graph neural networks. In *ICCV*, pages 9236–9245, 2019.

[63] Wenguan Wang and Jianbing Shen. Deep cropping via attention box prediction and aesthetics assessment. In *ICCV*, pages 2186–2194, 2017.

[64] Wenguan Wang, Jianbing Shen, Ming-Ming Cheng, and Ling Shao. An iterative and cooperative top-down and bottom-up inference network for salient object detection. In *CVPR*, pages 5968–5977, 2019.

[65] Wenguan Wang, Jianbing Shen, Xingping Dong, and Ali Borji. Salient object detection driven by fixation prediction. In *CVPR*, pages 1711–1720, 2018.

[66] Wenguan Wang, Jianbing Shen, Ling Shao, and Fatih Porikli. Correspondence driven saliency transfer. *IEEE TIP*, 25(11):5025–5034, 2016.

[67] Wenguan Wang, Jianbing Shen, Jianwen Xie, Ming-Ming Cheng, Haibin Ling, and Ali Borji. Revisiting video saliency prediction in the deep learning era. *IEEE TPAMI*, 2019.

[68] Wenguan Wang, Hongmei Song, Shuyang Zhao, Jianbing Shen, Sanyuan Zhao, Steven CH Hoi, and Haibin Ling. Learning unsupervised video object segmentation through visual attention. In *CVPR*, pages 3064–3074, 2019.

[69] Wenguan Wang, Shuyang Zhao, Jianbing Shen, Steven CH Hoi, and Ali Borji. Salient object detection with pyramid attention and salient edges. In *CVPR*, pages 1448–1457, 2019.

[70] Linwei Ye, Zhi Liu, Lina Li, Liquan Shen, Cong Bai, and Yang Wang. Salient object segmentation via effective integration of saliency and objectness. *IEEE TMM*, 19(8):1742–1756, 2017.

[71] Dingwen Zhang, Junwei Han, Chao Li, Jingdong Wang, and Xuelong Li. Detection of co-salient objects by looking deep and wide. *IJCV*, 120(2):215–232, 2016.

[72] Dingwen Zhang, Deyu Meng, and Junwei Han. Co-saliency detection via a self-paced multiple-instance learning framework. *IEEE TPAMI*, 39(5):865–878, 2017.

[73] Jing Zhang, Deng-Ping Fan, Yuchao Dai, Saeed Anwar, Fatemeh Sadat Saleh, Tong Zhang, and Nick Barnes. UC-Net: uncertainty inspired rgb-d saliency detection via conditional variational autoencoders. In *CVPR*, 2020.

[74] Pingping Zhang, Dong Wang, Huchuan Lu, Hongyu Wang, and Xiang Ruan. Amulet: Aggregating multi-level convolutional features for salient object detection. In *ICCV*, pages 202–211, 2017.

[75] Pingping Zhang, Dong Wang, Huchuan Lu, Hongyu Wang, and Baocai Yin. Learning uncertain convolutional features for accurate saliency detection. In *ICCV*, pages 212–221, 2017.

[76] Xiaoning Zhang, Tiantian Wang, Jinqing Qi, Huchuan Lu, and Gang Wang. Progressive attention guided recurrent network for salient object detection. In *CVPR*, pages 714–722, 2018.

[77] Jia-Xing Zhao, Yang Cao, Deng-Ping Fan, Ming-Ming Cheng, Xuan-Yi Li, and Le Zhang. Contrast prior and fluid pyramid integration for rgbd salient object detection. In *CVPR*, pages 3927–3936, 2019.

[78] Jia-Xing Zhao, Jiang-Jiang Liu, Deng-Ping Fan, Yang Cao, Jufeng Yang, and Ming-Ming Cheng. EGNet: Edge guidance network for salient object detection. In *ICCV*, pages 8779–8788, 2019.

[79] Tao Zhou, Huazhu Fu, Chen Gong, Jianbing Shen, Ling Shao, and Fatih Porikli. Multi-mutual consistency induced transfer subspace learning for human motion segmentation. In *CVPR*, 2020.

[80] Chunbiao Zhu, Xing Cai, Kan Huang, Thomas H Li, and Ge Li. PDNet: prior-model guided depth-enhanced network for salient object detection. In *ICME*, pages 199–204, 2019.