Salient Object Detection in the Deep Learning Era: An In-Depth Survey

Wenguan Wang, *Member, IEEE*, Qiuxia Lai, Huazhu Fu, *Senior Member, IEEE*, Jianbing Shen, *Senior Member, IEEE*, Haibin Ling, and Ruigang Yang, *Senior Member, IEEE*

Abstract—As an essential problem in computer vision, salient object detection (SOD) from images has been attracting an increasing amount of research effort over the years. Recent advances in SOD, not surprisingly, are dominantly led by deep learning-based solutions (named deep SOD) and reflected by hundreds of published papers. To facilitate the in-depth understanding of deep SODs, in this paper we provide a comprehensive survey covering various aspects ranging from algorithm taxonomy to unsolved open issues. In particular, we first review deep SOD algorithms from different perspectives including network architecture, level of supervision, learning paradigm and object/instance level detection. Following that, we summarize existing SOD evaluation datasets and metrics. Then, we carefully compile a thorough benchmark results of SOD methods based on previous work, and provide detailed analysis of the comparison results. Moreover, we study the performance of SOD algorithms under different attributes, which have been barely explored previously, by constructing a novel SOD dataset with rich attribute annotations. We further analyze, for the first time in the field, the robustness and transferability of deep SOD models w.r.t. adversarial attacks. We also look into the influence of input perturbations, and the generalization and hardness of existing SOD datasets. Finally, we discuss several open issues and challenges of SOD, and point out possible research directions in future. All the saliency prediction maps, our constructed dataset with annotations, and codes for evaluation are made publicly available at https://github.com/wenguanwang/SODsurvey.

Index Terms—Salient Object Detection, Deep Learning, Image Saliency.

1 Introduction

Salient object detection (SOD) aims at highlighting visually salient object regions in images, which is driven by and applied to a wide spectrum of object-level applications in various areas. In computer vision, representative applications include image understanding [1], [2], image captioning [3]–[5], object detection [6], [7], un-supervised video object segmentation [8], [9], semantic segmentation [10]–[12], person re-identification [13], [14], video summarization [15], [16], *etc*. In computer graphics, SOD plays an essential role in various tasks like non-photo-realist rendering [17], [18], automatic image cropping [19], image retargeting [20], [21], *etc*. Examplary applications in robotics, like human-robot interaction [22], [23], and object discovery [24], [25] also benefit from SOD for better scene/object understanding.

Different from fixation prediction (FP), which is originated from cognitive and psychology research communities, Significant improvement for SOD has been witnessed in recent years with the renaissance of deep learning techniques, thanks to the powerful representation learning methods. Since the first introduction in 2015 [26]–[28], deep learning-based SOD (or *deep SOD*) algorithms have soon shown superior performance over traditional solutions, and kept residing the top of various benchmarking leaderboards. On the other hand, hundreds of research papers have been produced on deep SOD, making it non-trivial for effectively understanding the state-of-the-arts.

This paper provides a comprehensive and in-depth survey on SOD in the deep learning era. It aims to cover thoroughly various aspects of deep SOD and related issues, ranging from algorithm taxonomy to unsolved open issues. Aside from taxonomically reviewing existing deep SOD methods and datasets, it investigates crucial but largely under-explored issues such as the effect of attributes in SOD, and the robustness and transferability of deep SOD models w.r.t. adversarial attacks. For these novel studies, we construct a new dataset and annotations, and derive baselines on top of previous studies. All the saliency prediction maps, our constructed dataset with annotations, and codes for evaluation are made publicly available at https://github.com/wenguanwang/SODsurvey.

1.1 History and Scope

Human beings are able to quickly allocate attentions on important regions in visual scenes. Undertanding and modeling such an astonishing ability, *i.e.*, visual attention or visual saliency, is a fundamental research problem in psychology, neurobiology, cognitive science and computer vision. There are two categories of computational models for visual saliency, namely *Fixation Prediction* (FP) and *Salient Object*

- *W. Wang is with ETH Zurich, Switzerland. (Email: wenguanwang.ai@gmail.com)*
- *Q. Lai is with the Department of Computer Science and Engineering, the Chinese University of Hong Kong, Hong Kong, China. (Email: qxlai@cse.cuhk.edu.hk)*
- *H. Fu is with Inception Institute of Artificial Intelligence, UAE. (Email: hzfu@ieee.org)*
- *J. Shen is with Beijing Laboratory of Intelligent Information Technology, School of Computer Science, Beijing Institute of Technology, China. (Email: shenjianbing@bit.edu.cn)*
- *H. Ling is with the Department of Computer and Information Sciences, Temple University, Philadelphia, PA, USA. (Email: hbling@temple.edu)*
- *R. Yang is with the University of Kentucky, Lexington, KY 40507. (Email: ryang@cs.uky.edu)*
- *First two authors contribute equally.*
- *Corresponding author: Jianbing Shen*

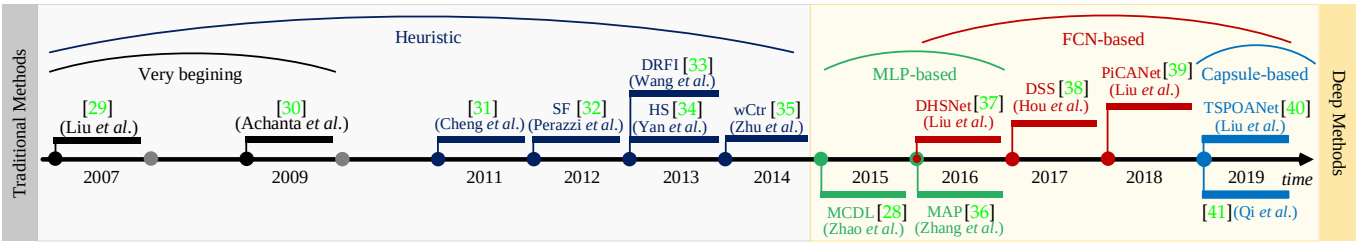

Fig. 1. A brief chronology of salient object detection (SOD). The very first SOD models date back to the work of Liu *et al*. [29] and Achanta *et al*. [30]. The first incorporation of deep learning techniques in SOD models is from 2015. See §1.1 for more details.

TABLE 1
Summary of previous reviews. See §1.2 for more detailed descriptions.

#	Title	Year	Venue	Description
1	State-of-the-Art in Visual Attention Modeling [42]	2013	TPAMI	This paper reviews visual attention (*i.e.* fixation prediction) models before 2013.
2	Salient Object Detection: A Benchmark [43]	2015	TIP	This paper benchmarks 29 heuristic SOD models and 10 FP methods over 7 datasets.
3	Attentive Systems: A Survey [44]	2017	IJCV	This paper reviews applications that utilize visual saliency cues.
4	A Review of Co-Saliency Detection Algorithms: Fundamentals, Applications, and Challenges [45]	2018	TIST	This paper reviews the fundamentals, challenges, and applications of co-saliency detection.
5	Review of Visual Saliency Detection with Comprehensive Information [46]	2018	TCSVT	This paper reviews RGB-D saliency detection, co-saliency detection and video saliency detection.
6	Advanced Deep-Learning Techniques for Salient and Category-Specific Object Detection: A survey [47]	2018	SPM	This paper reviews several sub-directions of object detection, namely objectness detection, salient object detection and category-specific object detection.
7	Saliency prediction in the deep learning era: Successes and limitations [48]	2019	TPAMI	This paper reviews image and video fixation prediction models and analyzes specific questions.
8	Salient Object Detection: A Survey [49]	2019	CVM	This paper reviews 65 heuristic and 21 deep SOD models till 2017 and discusses closely related areas like object detection, fixation prediction, segmentation, *etc*.

Detection (SOD). FP originated from cognitive and psychology research communities [50]–[52], which is to predict the human fixation points when observing a scene.

The history of SOD is relatively short and can be traced back to the pioneer works in [29] and [30]. SOD is mainly driven by the wide range of object-level computer vision applications. Most of early, **non-deep learning SOD models** [35], [53]–[55] are based on low-level features and rely on certain heuristics (*e.g.*, *color contrast* [31], *background prior* [56]). For obtaining uniformly highlighted salient objects and clear object boundaries, an over-segmentation process that generates regions [57], super-pixels [58], [59], or object proposals [60] is often integrated into above models. Please see [43] for a comprehensive overview.

With the compelling success of deep learning technologies in computer vision, more and more **deep learning-based SOD methods** have been springing up since 2015. Earlier deep SOD models typically utilize multi-layer perceptron (MLP) classifiers to predict the saliency score of deep features extracted from each image processing unit [26]–[28]. More recently, a more effective and efficient form, *i.e.*, fully convolutional network (FCN)-based network, becomes the mainstream of SOD architecture. Later, Capsule [61] is introduced into SOD for better modeling the object properties [40], [62]. Different deep models have different levels of supervision, and may use different learning paradigms. Specially, some SOD methods distinguish individual instances among all the detected salient objects [36], [63]. A brief chronology is shown in Fig. 1.

Scope of the survey. Despite having a short history, research in deep SOD has produced hundreds of papers, making it impractical (and fortunately unnecessary) to review all of them. Instead, we carefully and thoroughly select influential papers published in prestigious journals and conferences. This survey mainly focuses on the major progress in the last five years; but for completeness and better readability, some early related works are also included. It is worth noting that we restrict this survey to *single image object-level SOD* methods, and leave instance-level SOD, RGB-D saliency detection, co-saliency detection, video SOD, FP, social gaze prediction, *etc*., as separate topics.

This paper clusters the existing approaches based on various aspects including network architectures, level of supervision, influence of learning paradigm, *etc*. Such comprehensive and multi-angular classifications are expected to facilitate the understanding of past efforts in deep SOD. More in-depth analysis are summarized in §1.3.

1.2 Related Previous Reviews and Surveys

Table 1 lists existing surveys that are closely related to our paper. Among these works, Borji *et al*. [43] comprehensively review SOD methods preceding 2015, thus does not refer to recent deep learning-based solutions. Zhang *et al*. [45] review methods for co-segmentation, a branch of visual saliency that detects and segments common and salient foregrounds from more than one relevant images. Cong *et al*. [46] review several extended SOD tasks including RGB-D SOD, co-saliency detection and video SOD. Han *et al*. [47] look into the sub-directions of object detection, and conclude the recent progress in objectness detection, SOD, and category-specific object detection. Borji *et al*. summarize both heuristic [42] and deep models [48] for FP, another important branch of visual saliency, and analyze several special issues. Nguyen *et al*. [44] mainly focuses on categorizing the applications of visual saliency (including both SOD and FP) in different areas. A recent published survey [49] covers both traditional non-deep methods and deep ones till 2017, and discusses the relation w.r.t. several other closely-related research areas such as special-purpose object detection, fixation prediction and segmentation.

Different from previous SOD surveys, in this paper we systematically and comprehensively review *deep learning-*

based SOD methods. Our survey is featured by in-depth analysis and discussion in various aspects, many of which, to the best of our knowledge, are the first time in this field. In particular, we summarize existing deep SOD methods based on several proposed taxonomies, gain deeper understanding of SOD models through attribute-based evaluation, discuss on the influence of input perturbation, analyze the robustness of deep SOD models w.r.t. adversarial attacks, study the generalization and hardness of existing SOD datasets, and offer insights for essential open issues, challenges, and future directions. We expect our survey to provide novel insight and inspiration for facilitating the understanding of deep SOD, and to inspire research on the raised open issues such as the adversarial attacks to SOD.

1.3 Our Contributions

Our contributions in this paper are summarized as follows:

1) **Systematic review of deep SOD models from various perspectives.** We categorize and summarize existing deep SOD models according to network architecture, level of supervision, learning paradigm, *etc*. The proposed taxonomies aim to help researchers with deeper understanding of the key features of deep SOD models.
2) **A novel attribute-based performance evaluation of deep SOD models.** We compile a hybrid benchmark and provide annotated attributes considering object categories, scene categories and challenge factors. We evaluate six popular SOD models on it, and discuss how these attributes affect different algorithms and the improvements brought by deep learning techniques.
3) **Discussion regarding the influence of input perturbations.** We investigate the effects of various types of perturbations on six representative SOD models, which compliments the study on intentionally designed perturbations such as adversarial disturbances.
4) **The first known adversarial attack analysis on SOD models.** We provide the first study on this issue with carefully designed baseline attacks and evaluations, which could serve as baselines for future study of the robustness and transferability of deep SOD models.
5) **Cross-dataset generalization study.** To study the bias exists in SOD datasets [43], we conduct a cross-dataset generalization study of existing SOD datasets with a representative baseline model.
6) **Overview of open issues and future directions.** We thoroughly look over several essential issues for model design, dataset collection, and the relation of SOD with other topics, which shed light on potential directions for future research.

These contributions altogether bring an exhaustive, up-to-date, and in-depth survey, and differentiate it from previous review papers significantly.

The rest of the paper is organized as follows. §2 explains the proposed taxonomies and conducts a comprehensive literature review accordingly. §3 examines the most notable SOD datasets, whereas §4 describes several widely used SOD metrics. §5 benchmarks several deep SOD models and provides in-depth analyses. §6 provides a discussion and presents open issues and research challenges of the field. Finally, §7 concludes the paper.

2 Deep Learning based SOD Models

Before reviewing in details recent deep SOD models, we first give a common formulation of the image-based SOD problem. Given an input image $\mathbf{I} \in \mathbb{R}^{W \times H \times 3}$ of size $W \times H$, an SOD model f maps the input image $\mathbf{I}$ to a continuous saliency map $\mathbf{S} = f(\mathbf{I}) \in [0,1]^{W \times H}$.

For learning-based SOD, the model f is learned through a set of training samples. Given a set of N static images $\mathcal{I} = \{\mathbf{I}_n \in \mathbb{R}^{W \times H \times 3}\}_{n=1}^{N}$ and the corresponding binary ground-truth annotations $\mathcal{G} = \{\mathbf{G}_n \in \{0,1\}^{W \times H}\}_{n=1}^{N}$, the goal of learning is to find $f \in \mathcal{F}$ that minimizes the prediction error, *i.e.*, $\sum_{n=1}^{N} \ell(\mathbf{S}_n, \mathbf{G}_n)$, where ℓ is a certain distance measure (*e.g.*, defined in §4), $\mathbf{S}_n = f(\mathbf{I}_n)$, and $\mathcal{F}$ is the set of potential mapping functions. Deep SOD methods typically model f through modern deep learning techniques, as will be reviewed in this section. The ground-truths $\mathcal{G}$ can be collected by different methodologies, *i.e.*, direct human-annotation or eye-fixation-guided labeling, and may have different formats, *i.e.*, pixel-wise or bounding-box level, which will be discussed in §3.

In the rest of this section, we review deep SOD methods in four taxonomies. We first characterize typical *network architectures* for SOD (§2.1). Next, we categorize the SOD methods based on the *level of supervision* (§2.2). Then, in §2.3, we look into the SOD methods from the perspective of *learning paradigm*. Finally, based on whether or not to distinguish among different objects, we classify the deep SOD methods into *object-level and instance-level* ones (§2.4). We group important models by type and describe them in rough chronological order. A comprehensive summary of the reviewed models is provided in Table 2.

2.1 Representative Network Architectures for SOD

Based on the primary network architectures adopted, we classify deep SOD models into four categories, namely *Multi-layer Perceptron* (MLP)-based (§2.1.1), *Fully Convolutional Network* (FCN)-based (§2.1.2), *Hybrid Network*-based (§2.1.3) and *Capsule*-based (§2.1.4).

2.1.1 Multi-Layer Perceptron (MLP)-based Methods

MLP-based methods typically extract deep features for each processing unit of an image to train an MLP-classifier for saliency score prediction, as shown in Fig. 2 (a). Commonly adopted processing units include *super-pixels/patches* [28], [64], [67], and generic *object proposals* [26], [27], [36], [75].

1) Super-pixel/patch-based methods use regular (patch) or nearly-regular (super-pixel) image decomposition.

• **MCDL [28]** uses two pathways for extracting local and global context from two super-pixel-centered windows of different sizes, which are fed into an MLP for foreground/background classification.

• **ELD [67]** concatenates deep convolution features and an *encoded low level distance* map (ELD-map) to construct a feature vector for each super-pixel. The ELD-map is generated from the initial hand-crafted feature distance maps of the queried super-pixel using CNN.

• **MDF [26]** extracts multi-scale feature vectors for each image segments using a pre-trained image classification DNN. An MLP is trained to regress segment-level saliency. The final map is the combination of three maps of each scale.

TABLE 2
Summary of popular SOD methods. See §2 for more detailed descriptions.

	#	Methods	Publ.	Architecture	Backbone	Level of Supervision	Learning Paradigm	Obj.-/Inst.-Level SOD	Training Dataset	#Training	CRF
2015	1	SuperCNN [64]	IJCV	MLP+super-pixel	-	Fully-Sup.	STL	Object	ECSSD [57]	800	
	2	MCDL [28]	CVPR	MLP+super-pixel	GoogleNet	Fully-Sup.	STL	Object	MSRA10K [65]	8,000	
	3	LEGS [27]	CVPR	MLP+segment	-	Fully-Sup.	STL	Object	MSRA-B [29]+PASCAL-S [66]	3,000+340	
	4	MDF [26]	CVPR	MLP+segment	-	Fully-Sup.	STL	Object	MSRA-B [29]	2,500	
2016	1	ELD [67]	CVPR	MLP+super-pixel	VGGNet	Fully-Sup.	STL	Object	MSRA10K [65]	∼9,000	
	2	DHSNet [37]	CVPR	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA10K [65]+DUT-OMRON [58]	6,000+3,500	
	3	DCL [68]	CVPR	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA-B [29]	2,500	✓
	4	RACDNN [69]	CVPR	FCN	VGGNet	Fully-Sup.	STL	Object	DUT-OMRON [58]+NJU2000 [70]+RGBD [71]	10,565	
	5	SU [72]	CVPR	FCN	VGGNet	Fully-Sup.	MTL	Object	MSRA10K [65]+SALICON [73]	10,000+15,000	✓
	6	MAP [36]	CVPR	MLP+obj. prop.	VGGNet	Fully-Sup.	MTL	Instance	SOS [74]	∼5,500	
	7	SSD [75]	ECCV	MLP+obj. prop.	AlexNet	Fully-Sup.	STL	Object	MSRA-B [29]	2,500	
	8	CRPSD [76]	ECCV	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA10K [65]	10,000	
	9	RFCN [77]	ECCV	FCN	VGGNet	Fully-Sup.	MTL	Object	PASCAL VOC 2010 [78]+MSRA10K [65]	10,103+10,000	
2017	1	MSRNet [63]	CVPR	FCN	VGGNet	Fully-Sup.	STL	Instance	MSRA-B [29]+HKU-IS [26] (+ILSO [63])	2,500+2,500 (+500)	✓
	2	DSS [38]	CVPR	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA-B [29]+HKU-IS [26]	2,500	✓
	3	WSS [79]	CVPR	FCN	VGGNet	Weakly-Sup.	MTL	Object	ImageNet [80]	456k	✓
	4	DLS [81]	CVPR	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA10K [65]	10,000	
	5	NLDF [82]	CVPR	FCN	VGGNet	Fully-Sup.	MTL	Object	MSRA-B [29]	2,500	✓
	6	DSOS [83]	ICCV	FCN	VGGNet	Fully-Sup.	MTL	Object	SOS [74]	6,900	
	7	Amulet [84]	ICCV	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA10K [65]	10,000	
	8	FSN [85]	ICCV	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA10K [65]	10,000	
	9	SBF [86]	ICCV	FCN	VGGNet	Un-Sup.	STL	Object	MSRA10K [65]	10,000	
	10	SRM [87]	ICCV	FCN	ResNet	Fully-Sup.	STL	Object	DUTS [79]	10,553	
	11	UCF [88]	ICCV	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA10K [65]	10,000	
2018	1	RADF [89]	AAAI	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA10K [65]	10,000	✓
	2	ASMO [90]	AAAI	FCN	ResNet101	Weakly-Sup.	MTL	Object	MS COCO [91]+MSRA-B [29]+HKU-IS [26]	82,783+2,500+2,500	✓
	3	LICNN [92]	AAAI	FCN	VGGNet	Weakly-Sup.	STL	Object	ImageNet [80]	456k	
	4	BDMP [93]	CVPR	FCN	VGGNet	Fully-Sup.	STL	Object	DUTS [79]	10,553	
	5	DUS [94]	CVPR	FCN	ResNet101	Un-Sup.	MTL	Object	MSRA-B [29]	2,500	
	6	DGRL [95]	CVPR	FCN	ResNet50	Fully-Sup.	STL	Object	DUTS [79]	10,553	
	7	PAGR [96]	CVPR	FCN	VGGNet19	Fully-Sup.	STL	Object	DUTS [79]	10,553	
	8	RSDNet [97]	CVPR	FCN	ResNet101	Fully-Sup.	MTL	Object	PASCAL-S [66]	425	
	9	ASNet [98]	CVPR	FCN	VGGNet	Fully-Sup.	MTL	Object	SALICON [73]+MSRA10K [65]+DUT-OMRON [58]	15,000+10,000+5,168	
	10	PiCANet [39]	CVPR	FCN	VGGNet/ResNet50	Fully-Sup.	STL	Object	DUTS [79]	10,553	✓
	11	C2S-Net [99]	ECCV	FCN	VGGNet	Weakly-Sup.	MTL	Object	MSRA10K [65]+Web	10,000+20,000	
	12	RAS [100]	ECCV	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA-B [29]	2,500	
2019	1	SuperVAE [101]	AAAI	FCN	N/A	Un-Sup.	STL	Object	N/A	N/A	
	2	DEF [102]	AAAI	FCN	ResNet101	Fully-Sup.	STL	Object	DUTS [79]	10,553	
	3	AFNet [103]	CVPR	FCN	VGGNet16	Fully-Sup.	MTL	Object	DUTS [79]	10,553	
	4	BASNet [62]	CVPR	FCN	ResNet-34	Fully-Sup.	STL	Object	DUTS [79]	10,553	
	5	CapSal [104]	CVPR	FCN	ResNet101	Fully-Sup.	MTL	Object	COCO-CapSal [104]/DUTS [79]	5,265/10,553	
	6	CPD-R [105]	CVPR	FCN	ResNet50	Fully-Sup.	STL	Object	DUTS [79]	10,553	
	7	MLSLNet [106]	CVPR	FCN	VGG16	Fully-Sup.	MTL	Object	DUTS [79]	10,553	
	8	†MWS [107]	CVPR	FCN	N/A	Weakly-Sup.	STL	Object	ImageNet DET [80]+MS COCO [91] +ImageNet [108]+DUTS [79]	456k+82,783 +300,000+10,553	
	9	PAGE-Net [109]	CVPR	FCN	VGGNet16	Fully-Sup.	MTL	Object	MSRA10K [65]	10,000	✓
	10	PS [110]	CVPR	FCN	ResNet50	Fully-Sup.	STL	Object	MSRA10K [65]	10,000	✓
	11	PoolNet [111]	CVPR	FCN	ResNet50	Fully-Sup.	STL/MTL	Object	DUTS [79]	10,553	
	12	BANet [112]	ICCV	FCN	ResNet50	Fully-Sup.	MTL	Object	DUTS [79]	10,553	
	13	EGNet [113]	ICCV	FCN	VGGNet/ResNet	Fully-Sup.	MTL	Object	DUTS [79]	10,553	
	14	HRSOD [114]	ICCV	FCN	VGGNet	Fully-Sup.	STL	Object	DUTS [79]/HRSOD [114]+DUTS [79]	10,553/12,163	
	15	JDFPR [115]	ICCV	FCN	VGG	Fully-Sup.	STL	Object	MSRA-B [29]	2,500	✓
	16	SCRN [116]	ICCV	FCN	ResNet50	Fully-Sup.	MTL	Object	DUTS [79]	10,553	
	17	SSNet [117]	ICCV	FCN	Desenet169	Fully-Sup.	MTL	Object	PASCAL VOC 2012 [78]+DUTS [79]	1,464+10,553	✓
	18	TSPOANet [40]	ICCV	Capsule	FLNet	Fully-Sup.	STL	Object	DUTS [79]	10,553	

• **SuperCNN [64]** constructs two hand-crafted input feature sequences for each super-pixel, which are further processed by two CNN columns separately to produce saliency scores using 1D convolution instead of fully connected layers.

2) Object proposal-based methods leverage object proposals [26], [27] or bounding-boxes [36], [75] as basic processing units that naturally encode object information.

• **LEGS** [27] constructs segment-level feature vectors out of pixel-level deep features, then uses an MLP to predict saliency scores from the segment-level features.

• **MAP [36]** uses a CNN model to generate a set of scored bounding boxes, then selects an optimized compact subset of bounding boxes as the salient objects.

• **SSD [75]** first generates region proposals and then uses a CNN to classify each proposal into a pre-defined shape class with standard binary map. The final saliency map is averaged over the binary maps of all the proposals.

2.1.2 Fully Convolutional Network (FCN)-based Methods

Though having outperformed previous non-deep learning SOD models and heuristic ones with deeply learned features, the MLP-based SOD models cannot capture well critical spatial information and are time-consuming as they need to process all visual sub-units one by one. Inspired by the great success of Fully Convolutional Network (FCN) [118] in semantic segmentation, recent deep SOD solutions adapt popular classification models, *e.g.*, VGGNet [119] and ResNet [120] to directly output whole saliency maps. This way, these deep SOD solutions benefit from end-to-end spatial saliency representation learning and efficiently predict saliency maps in a single feed-forward process. Typical architectures include five categories: *Single-stream network*, *Multi-stream network*, *Side-fusion network*, *Bottom-up/top-down network*, and *Branched network*.

1) Single-stream network is a standard architecture consisting of a sequential cascade of convolution layers, pooling layers and non-linear activation operations (see Fig. 2 (b)).

• **RFCN [77]** recurrently refines the saliency prediction based on the input image and the saliency priors from heuristic calculation or prediction of previous time step. It can be viewed as a cascaded structure after being unrolled.

• **RACDNN [69]** produces a coarse saliency map using an encoder-decoder stream, and progressively refines different local object regions. It utilizes a spatial transformer [121] to attend to an image region at each iteration for refinement.

• **DLS [81]** utilizes a stack of convolution and dilated

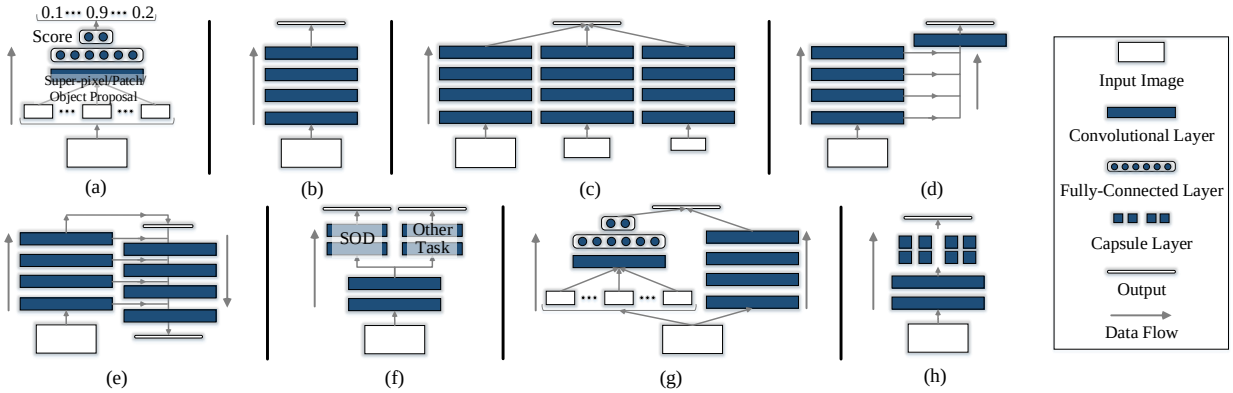

Fig. 2. Category of previous deep SOD models. (a) MLP-based methods. (b)-(f) FCN-based methods, mainly using (b) single-stream network, (c) multi-stream network, (d) side-out fusion network, (e) bottom-up/top-down network, and (f) branch network architectures. (g) Hybrid network-based methods. (h) Capsule-based methods. See §2.1 for more detailed descriptions.

convolution layers to produce an initial saliency map, and then refines it at super-pixel level. A level set loss function is used to aid the learning of the binary segmentation map.

• **UCF [88]** uses an encoder-decoder architecture to produce finer-resolution predictions. It learns uncertainty through a reformulated dropout in the decoder, and avoids artifacts by using a hybrid up-sampling scheme in the decoder.

• **DUS [94]** is based on the Deeplab [122] algorithm, which is an FCN with dilated convolution layers on the top. It learns the latent saliency and noise pattern by pixel-wise supervision from several heuristic saliency methods.

• **LICNN [92]** generates 'post-hoc' saliency maps by combining top-5 category-specific attention maps of a pre-trained image classification network. The lateral inhibition enhances the discriminative ability of the attention maps, releasing it from the need of SOD annotations.

• **SuperVAE [101]** trains a variational autoencoder (VAE) with background super-pixels. The final saliency map is estimated using the reconstruction errors of each super-pixel, as the VAE is expected to reconstruct the background with smaller errors than that of the foreground objects.

2) Multi-stream network, as depicted in Fig. 2 (c), typically has multiple network streams for explicitly learning multi-scale saliency features from inputs of varied resolutions or with different structures. Some multi-stream networks handle different tasks at separate pathways. The outputs from different streams are combined to form final prediction.

• **MSRNet [63]** consists of three streams of bottom-up/top-down network structure to process three scaled versions of the input image. The three outputs are finally fused through a learnable attention module.

• **SRM [87]** refines saliency features by passing them stage-wisely from a coarser stream to a finer one. The top-most feature of each stream is supervised with the ground-truth salient object mask. The pyramid pooling module further facilitates multi-stage saliency fusion and refinement.

• **FSN [85]**, in view of that salient objects typically gain most of human eye-fixations [66], fuses the outputs of a fixation stream [123] and a semantic stream [119] into an inception-segmentation module to predict the salient object scores.

• **HRSOD [114]** captures global semantics and local high-resolution details by two bottom-up/top-down pathways for handling the entire image and the attended image patches, respectively. The patch sampling is guided by the saliency results of the entire image.

• **DEF [102]** embeds multi-scale features from the backbone into a metric space following the initial saliency map at top. The embedded features later guide the top-down pathway to generate the salient map stage-wisely.

3) Side-fusion network fuses multi-layer responses of a backbone network together for SOD prediction, making use of the inherent multi-scale representations of the CNN hierarchy (Fig. 2 (d)). Side-outputs are typically supervised by the ground-truth, leading to a *deep supervision* strategy [124].

• **DSS [38]** adds several short connections from deeper side-outputs to shallower ones. In this way, higher-level features can help lower side-outputs to better locate the salient regions, while lower-level features can help enrich the higher-level side-outputs with finer details.

• **NLDF [82]** generates a local saliency map by fusing multi-level features and contrast features in a top-down manner, then integrates it with a global one yielded by the top layer to produce the final prediction. The contrast features are obtained by subtracting the feature from its average pooling.

• **Amulet [84]** aggregates multi-level features into multiple resolutions. The multiple aggregated features are further refined in a top-down manner. A boundary refinement is introduced at each aggregated feature before final fusion.

• **DSOS [83]** uses two subnets for detecting salient objects and subitizing the result, respectively. The detection subnet is a U-net [125] with side-fusions, whose bottle-neck parameters are dynamically determined by the other subnet.

• **RADF [89]** utilizes the integrated side-features to refine themselves, and such process is repeated to gradually yield finer saliency predictions.

• **RSDNet-R [97]** combines an initial coarse representation with finer features at earlier layers under a gating mechanism to stage-wisely refine the side-outputs. Maps from all the stages are fused to obtain the overall saliency map.

• **CPD [105]** integrates features of deeper layers in the backbone to get an initial saliency map, which is used to refine the features of the backbone after going though a Holistic Attention Module to generate the final map.

• **MWS [107]** adds together the four saliency maps from the four dilated convolution layers of the feature extractor, which is deconvolutioned to the size of the input image.

• **EGNet [113]** extracts multi-scale features using a bottom-up/top-down structure enhanced with downward location propagation, which are then guided by the salient edge features to make side-outputs and fused into the final map.

4) Bottom-up/top-down network refines the rough saliency estimation in the feed-forward pass by progressively incorporating spatial-detail-rich features from lower layers, and produces the final map at the top-most layer (see Fig. 2 (e)).

• **DHSNet [37]** refines the coarse saliency map by gradually combining shallower features using recurrent layers. All the intermediate maps are supervised by the ground truth [124].

• **SBF [86]** borrows the network architecture of DHSNet [37], but is trained under the weak ground truth provided by several un-supervised heuristic SOD methods.

• **BDMP [93]** refines multi-level features using convolution layers with various reception fields, and enables inter-level message exchange through a gated bi-directional pathway. The refined features are fused in a top-down manner.

• **RLN [95]** uses an inception-like module to purify the low-level features. A recurrent mechanism in the top-down pathway further refines the combined features. The saliency output is enhanced by a boundary refinement network.

• **PAGR [96]** enhances the learning ability of the feature extraction pathway by incorporating multi-path recurrent connections to transfer higher-level semantics to lower layers. The top-down pathway is embedded with several channel-spatial attention modules for refining the features.

• **ASNet [98]** learns a coarse fixation map in the feed-forward pass, then utilizes a stack of convLSTMs [126] to iteratively infer pixel-wise saliency map by incorporating multi-level features from successively shallower layers.

• **PiCANet [39]** hierarchically embeds global and local pixel-wise contextual attention modules into the top-down pathway of a U-Net [125] structure.

• **RAS [100]** embeds *reverse attention* (RA) blocks in the top-down pathway to guide residual saliency learning. The RA blocks emphasize the non-object areas using the complement of deeper-level output.

• **AFNet [103]** globally refines the top-most saliency feature with a Global Perceptron Module, and utilize the Attentive Feedback Modules (AFMs) to pass messages between the corresponding encoder and the decoder blocks. A Boundary Enhancement Loss is applied in the last two AFMs.

• **BASNet [62]** first makes a coarse saliency prediction with a deeply-supervised bottom-up/top-down structure, then refines the residual of the saliency map with another bottom-up/top-down module. A hybrid loss considering hierarchical cues implicitly aids accurate boundary prediction.

• **MLSLNet [106]** embeds mutual learning modules and edge modules hierarchically, each is deeply supervised by saliency and edge cues, respectively. The bottom-up and top-down pathways are trained in an intertwined manner.

• **PAGE-Net [109]** equips the typical bottom-up/top-down structure with pyramid attention module and edge detection module. The former enlarges the receptive field and provides multi-scale cues. The latter locates and sharpens saliency object boundaries with explicit edge information.

• **PoolNet [111]** generates global guidance from the top-most feature to guide the salient object localization in the top-down pathway. The feature aggreagation modules embedded in the top-down pathway further refines the fused features, resulting in finer prediction maps.

• **PS [110]** proposes an iterative top-down and bottom-up framework for saliency inference. Specially, the RNNs are used as the building blocks of the inference pathways to optimize the features of static images iteratively.

• **JDFPR [115]** embeds CRF blocks at each level of structure to refine features and prediction maps jointly, from coarser scale to finer ones. The CRF block allows message-passing of feature-feature, feature-prediction and prediction-prediction, enhancing features for prediction.

5) Branched network is a *single-input-multiple-output* structure, where the bottom layers are shared to process a common input and the top layers are specialized for different outputs. Its core scheme is shown in Fig. 2 (f).

• **SU [72]** performs eye-fixation prediction (FP) and SOD in a branched network. The shared layers capture the semantics and global saliency contexts. The FP branch learns to infer fixations from the top feature, while the SOD branch aggregates side-features to better preserve spatial cues.

• **WSS [79]** consists of an image classification branch and an SOD branch. The SOD branch benefits from the features trained under image-level supervision, and produces initial saliency maps in a top-down scheme which are refined by an iterative CRF and used for fine-tuning the SOD branch.

• **ASMO [90]** performs the same tasks with WSS [79] and is also trained under weak supervision. The main difference is that the shared network in ASMO uses a multi-stream structure to handle different scales of an input image.

• **C2S-Net [99]** is constructed by adding an SOD branch to a pre-trained contour detection model, *i.e.*, CEND [127]. The two branches are trained under an alternating scheme with the supervision signals provided by each other.

• **CapSal [104]** contains an image captioning subnet and a SOD subnet with shared backbone, where the latter consists of a local perception and a global perception branch whose outputs are fused for final prediction.

• **BANet [112]** tackles the *selectivity-invariance dilemma* [128] in SOD by handling boundaries and interiors of salient objects at separate streams to fulfill different feature requirements. A third stream complements the possible failures at the transitional regions between the two.

• **SCRN [116]** extracts two separate multi-scale features with a shared backbone for SOD and edge detection, which are gradually refined by a stack of Cross Refinement Units that allow bidirectional message passing between the two tasks.

• **SSNet [117]** consists of a semantic segmentation branch and a SOD branch sharing the feature extractor. The former is supervised by the image-level and the refined pseudo pixel-wise labels in tandem. The latter is fully-supervised.

2.1.3 Hybrid Network-based Methods

Some deep SOD methods combine both MLP- and FCN-based subnets, aiming to produce edge-preserving detection with multi-scale context (see Fig. 2 (g)).

• **DCL [68]** combines a pixel-wise prediction of a side-fusion FCN stream and a segment-level map produced by classifying multi-scale super-pixels based on deep features. The two branches share the same feature extraction network, and are alternatively optimized during training.

• **CRPSD [76]** also combines pixel-level and super-pixel-level saliency. The former is generated by fusing the last and penultimate side-output features of an FCN, while the latter is obtained by applying MCDL [28] to adaptively generated regions. Only the FCN and the fusion layers are trainable.

2.1.4 Capsule-based Methods

Recently, Hinton *et al*. [61], [129], [130] propose a new type of network, named *Capsule*. Capsules are made up of a group of neurons which accept and output vectors as opposed to CNNs' scalar values, allowing comprehensively modeling of entity properties. Such advantage inspires some researchers explore Capsule in SOD [40], [41] (Fig. 2 (h)).

• **TSPOANet [40]** emphasizes the part-object relationships with a two-stream capsule networks. The input deep features of the capsules are extracted using a bottom-up/top-down convolutional network, which are transformed into low-level capsules and assigned to high-level ones, and finally recognized to be salient or background.

2.2 Level of Supervision

Based on whether human-annotated salient object masks are used for training, deep SOD methods can be classified into *fully-supervised methods* and *un-/weakly-supervised methods*.

2.2.1 Fully-Supervised Methods

Most deep SOD models are trained with large-scale pixel-wise human annotations. However, for an SOD task, manually labeling a large amount of pixel-wise saliency annotations is time-consuming and requires heavy and intensive human labeling. Moreover, models trained on fine-labeled datasets tend to overfit and generalize poorly to real-life images. Thus, how to train SOD with less human annotations becomes an increasingly popular research direction.

2.2.2 Un-/Weakly-Supervised Methods

Un-/Weakly supervised learning refers to learning without task-specific ground-truth supervision. To get rid of the laborious manual labeling, some SOD methods make efforts to predict saliency using *image-level* categorical labels [79], [92], or pseudo *pixel-wise* saliency annotations generated by heuristic un-supervised SOD methods [86], [90], [94] or from other applications [99], [107].

1) Category-level supervision. It has been shown that the hierarchical deep features trained with image-level labels have the ability to locate the regions containing objects [131], [132], which is promising to provide useful cues for detecting salient objects in the scene.

• **WSS [79]** first pre-trains a two-branch network to predict image labels at one branch using ImageNet [80], while estimating saliency maps at the other. The estimated maps are refined by CRF and used to fine-tune the SOD branch.

• **LICNN [92]** turns to an ImageNet-pretrained image classification network to generate 'post-hoc' saliency maps. It does not need explicit training with any other SOD annotations thanks to the lateral inhibition mechanism.

• **SuperVAE [101]** trains the super-pixel-wise VAE through the perceptual loss, where the learned hidden features are forced to be consist with a pre-trained model.

2) Pseudo pixel-level supervision. Though being informative, image-level labels are too sparse to yield precise pixel-wise saliency masks. Some researchers propose to utilize traditional un-supervised SOD methods [86], [90], [94], contour information [99] or multi-source cues [107] to generate noisy saliency maps, which are refined and used for training.

• **SBF [86]** generates saliency predictions through a fusion process that integrates the weak saliency maps yielded by several classical un-supervised salient object detectors [34], [133], [134] at intra- and inter-image levels.

• **ASMO [90]** trains a multi-task FCN with image categorical labels and noisy maps of heuristic un-supervised SOD methods. The coarse saliency and the average map of the top-3 class activation maps [132] are fed into a CRF model to obtain finer maps for fine-tuning the SOD sub-net.

• **DUS [94]** jointly learns latent saliency and noise patterns from noisy saliency maps generated by several traditional un-supervised SOD methods [34], [35], [135], [136], and produces finer saliency maps for next training iteration.

• **C2S-Net [99]** generates pixel-wise salient object masks from contours [137] using CEDN [127] and trains the SOD branch. The contour and SOD branches alternatively update each other and progressively output finer SOD predictions.

• **MWS [107]** learns saliency prediction under multi-source supervision from an image classification network and a caption generation network.

2.3 Learning Paradigm

From the perspective of different learning paradigms, SOD networks can be divided into methods of *single-task learning (STL)* and *multi-task learning (MTL)*.

2.3.1 Single-Task Learning (STL) based Methods

In machine learning, the standard methodology is to learn one task at a time, *i.e.*, single-task learning [138]. Most deep SOD methods belong to this realm of learning paradigm. They utilize supervision from a single knowledge domain to train the SOD models, using either the SOD domain, or other related domains such as image classification [92].

2.3.2 Multi-Task Learning (MTL) based Methods

Inspired by human learning process where the knowledge learned from related tasks can be used to help learning a new task, Multi-Task Learning (MTL) [138] aims to learn multiple related tasks simultaneously. By incorporating domain-specific information from extra training signals of related tasks, the generalization ability of the model gets improved. The sharing of samples among tasks also alleviates the lack of data for training heavy-parameterized models.

1) Salient object subitizing [74]. The ability of human to rapidly enumerate a small number of items is referred to as subitizing [139]. Some SOD methods learn salient object subitizing and detection simultaneously.

• **MAP [36]** first outputs a set of scored bounding boxes that match the number and locations of the salient objects, then performs a subset optimization formulation based on *maximum a posteriori* to jointly optimize the number and locations of the salient object proposals.

• **DSOS [83]** uses an auxiliary network to learn salient object subitizing, which affects the SOD subnet by alternating the parameters of its adaptive weight layer.
• **RSDNet [97]**, different from above methods that explicitly model salient object subitizing as a classification problem, applies a stack of saliency-level-aware ground-truth masks to train the network that implicitly learns to figure out the number of salient objects as well as their relative saliency.

2) Fixation prediction aims to predict the locations of human eye-fixations. Due to its close relation with SOD, learning shared knowledge from the two closely related tasks is promising to improve the performances of both.
• **SU [72]** performs eye-fixation prediction and SOD in a branched network. The shared layers learn to capture the semantics and global saliency contexts. The branched layers are distinctively trained to handle task-specific problems.
• **ASNet [98]** learns SOD by jointly training a bottom-up pathway to predict fixations. A top-down pathway refines the object-level saliency estimation progressively by incorporating multi-level features guided by the fixation cues.

3) Image classification. The image category labels can help localize the discriminative regions [131], [132], [140], which often contain salient object candidates. Some methods thus leverage image-category classification to assist SOD task.
• **WSS [79]** learns a *foreground inference network* (FIN) for predicting image categories and estimating foreground maps for all categories. FIN is further fine-tuned with the CRF-refined foreground maps to predict saliency map.
• **ASMO [90]** learns to predict the saliency map and the image categories simultaneously under the supervision of category labels and pseudo ground-truth saliency maps from traditional un-supervised SOD methods.

4) Noise pattern modeling learns the noise pattern out of the noisy saliency maps generated by existing heuristic un-supervised SOD methods, aiming at extracting 'pure' saliency maps for supervising SOD training.
• **DUS [94]** proposes to model the noise pattern of the noisy supervision from traditional un-supervised SOD methods instead of denoising. SOD and noise pattern modeling tasks are jointly optimized under a unified loss.

5) Semantic segmentation is to assign each image pixel a label from a set of predefined categories. SOD can be viewed as a class-agnostic semantic segmentation where each pixel is classified as either belongs to a salient object or not. High-level semantics play an important role in distinguishing salient objects from backgrounds.
• **RFCN [77]** is first trained on a segmentation dataset [78] to learn semantics, and then fine-tuned on an SOD dataset to predict foreground and background maps. The saliency map is a softmax combination of the two kinds of maps.
• **SSNet [117]** obtains the saliency maps by fusing the segmentation masks of all the categories with the predicted class-wise saliency scores. The saliency aggregation module and the segmnetation network are trained jointly.

6) Contour/edge detection responds to edges belonging to objects without considering background boundaries, which can help localizing and segmenting salient object regions.
• **NLDF [82]** computes an IoU loss between the predicted and the real boundary pixels. This penalty loss term contributes significantly to finer boundaries.
• **C2S-Net [99]** encodes the common features of contour and SOD at shared bottom layers, and performs the two tasks at distinct branches. The former is fine-tuned from pre-trained model, while the latter is trained from scratch.
• **AFNet [103]** calculates the Boundary Enhancement Loss between the edge map obtained from the saliency map and the ground-truth without explicitly predicting the edges.
• **MLSLNet [106]** outputs edge detections from the edge modules connected to the shallower layers of the encoder pathway. Besides, the contour supervision is also applied at decoder pathway alternatively with the saliency cues.
• **PAGE-Net [109]** embeds edge detection modules hierarchically, which are explicitly trained for detecting salient object boundaries. Such extra supervision emphasizes the saliency boundary alignment.
• **PoolNet [111]** further improves the prediction performance by training with high-quality edge datasets [145], [146] to capture more details of the saleint objects.
• **BANet [112]** supervises the boundary localization stream with the boundary map to emphasize feature selectivities in detecting boundaries. The interior perception stream and the final prediction are supervised by the saliency maps.
• **EGNet [113]** applies explicit edge supervision at the final salient edge features, which guide the salient features for better segmentation and localization of the salient objects.
• **SCRN [116]** levarages the complementary cues between the two tasks of edge detection and SOD by exchanging messages between the two task-specific feature sets and to gradually refine the features for prediction.

7) Image Captioning can provide extra supervision for learning high-level semantics of the major objects.
• **CapSal [104]** jointly trains an image captioning subnet with the SOD subnet. The semantic context from the captioning subnet is incorporated with local and global visual cues for boosting the saliency prediction performance.

2.4 Object-/Instance-Level SOD

The goal of SOD is to locate and segment the most noticeable object regions in images. If the output mask only denotes the saliency of each pixel without distinguishing different objects, the method belongs to object-level SOD methods; otherwise, it is an instance-level SOD method.

2.4.1 Object-Level Methods

Most SOD methods are object-level methods, *i.e.*, designed to detect pixels that belong to the salient objects without being aware of the individual instances.

2.4.2 Instance-Level Methods

Instance-level SOD methods produce saliency masks with distinct object labels to perform more detailed parsing of the salient regions. The instance-level information is crucial for many practical applications that need finer distinctions.
• **MAP [36]** emphasizes instance-level SOD in unconstrained images. It first generates numerous object candidates, and then selects the top-ranking ones as the outputs.
• **MSRNet [63]** decomposes salient instance detection into three sub-tasks, *i.e.*, pixel-level saliency prediction, salient object contour detection and salient instance identification.

TABLE 3
Statistics of popular SOD datasets. See §3 for more detailed descriptions.

	#	Dataset	Year	Publ.	#Img.	#Obj.	Obj. Area(%)	SOD Annotation	Resolution	Fix.
Early	1	MSRA-A [29]	2007	CVPR	1,000/20,840	1-2	-	bounding-box object-level	-	
	2	MSRA-B [29]	2007	CVPR	5,000	1-2	$20.82_{\pm 10.29}$	bounding-box object-level, pixel-wise object-level	$\max(w,h)=400$, $\min(w,h)=126$	
	3	SED1 [141]	2007	CVPR	100	1	$26.70_{\pm 14.26}$	pixel-wise object-level	$\max(w,h)=465$, $\min(w,h)=125$	
	4	SED2 [141]	2007	CVPR	100	2	$21.42_{\pm 18.41}$	pixel-wise object-level	$\max(w,h)=300$, $\min(w,h)=144$	
	5	ASD [30]	2009	CVPR	1,000	1-2	$19.89_{\pm 9.53}$	pixel-wise object-level	$\max(w,h)=400$, $\min(w,h)=142$	
Modern&Popular	1	SOD [142]	2010	CVPR-W	300	1-4+	$27.99_{\pm 19.36}$	pixel-wise object-level	$\max(w,h)=481$, $\min(w,h)=321$	
	2	MSRA10K [65]	2015	TPAMI	10,000	1-2	$22.21_{\pm 10.09}$	pixel-wise object-level	$\max(w,h)=400$, $\min(w,h)=144$	
	3	ECSSD [57]	2015	TPAMI	1,000	1-4+	$23.51_{\pm 14.02}$	pixel-wise object-level	$\max(w,h)=400$, $\min(w,h)=139$	
	4	DUT-OMRON [58]	2013	CVPR	5,168	1-4+	$14.85_{\pm 12.15}$	pixel-wise object-level	$\max(w,h)=401$, $\min(w,h)=139$	✓
	5	PASCAL-S [66]	2014	CVPR	850	1-4+	$24.23_{\pm 16.70}$	pixel-wise object-level	$\max(w,h)=500$, $\min(w,h)=139$	✓
	6	HKU-IS [26]	2015	CVPR	4,447	1-4+	$19.13_{\pm 10.90}$	pixel-wise object-level	$\max(w,h)=500$, $\min(w,h)=100$	
	7	DUTS [79]	2017	CVPR	15,572	1-4+	$23.17_{\pm 15.52}$	pixel-wise object-level	$\max(w,h)=500$, $\min(w,h)=100$	
Special	1	SOS [74]	2015	CVPR	6,900	0-4+	$41.22_{\pm 25.35}$	object number, bounding-box (*train* set)	$\max(w,h)=6132$, $\min(w,h)=80$	
	2	MSO [74]	2015	CVPR	1,224	0-4+	$39.51_{\pm 24.85}$	object number, bounding-box instance-level	$\max(w,h)=3888$, $\min(w,h)=120$	
	3	ILSO [63]	2017	CVPR	1,000	1-4+	$24.89_{\pm 12.59}$	pixel-wise instance-level	$\max(w,h)=400$, $\min(w,h)=142$	
	4	XPIE [143]	2017	CVPR	10,000	1-4+	$19.42_{\pm 14.39}$	pixel-wise object-level, geographic information	$\max(w,h)=500$, $\min(w,h)=130$	✓
	5	SOC [144]	2018	ECCV	6,000	0-4+	$21.36_{\pm 16.88}$	pixel-wise instance-level, object category, attribute	$\max(w,h)=849$, $\min(w,h)=161$	
	6	COCO-CapSal [104]	2019	CVPR	6,724	1-4+	$23.74_{\pm 17.00}$	pixel-wise object-level, image caption	$\max(w,h)=640$, $\min(w,h)=480$	
	7	HRSOD [114]	2019	ICCV	2,010	1-4+	$21.13_{\pm 15.14}$	pixel-wise object-level	$\max(w,h)=10240$, $\min(w,h)=600$	

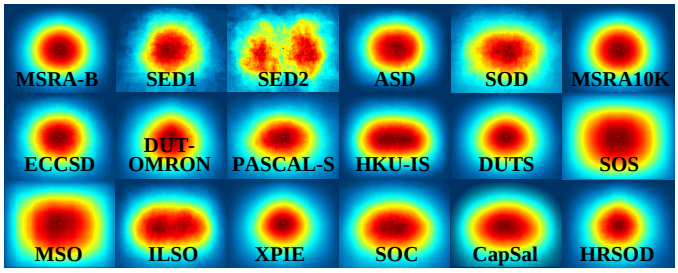

Fig. 3. Ground-truth annotation distributions of representative SOD datasets. See §3 for more detailed descriptions.

3 SOD Datasets

With the rapid development of SOD, numerous datasets have been introduced. Table 3 summarizes 19 representative datasets. Fig. 3 shows the annotation distribution of 18 available datasets.

3.1 Early SOD Datasets

Early SOD datasets typically contain simple scenes where 1∼2 salient objects stand out from simple backgrounds.

• **MSRA-A [29]** contains 20,840 images collected from various image forums and image search engines. Each image has a clear, unambiguous object and the corresponding annotation is the "majority agreement" of the bounding boxes provided by three users.

• **MSRA-B [29]**, as a subset of MSRA-A, has 5,000 images that are relabeled by nine users using bounding boxes. Compared with MSRA-A, MSRA-B has less ambiguity w.r.t. the salient object. The performances on MSRA-A and MSRA-B become saturated since most of the images only include a single and clear salient object around the center position.

• **SED [141]**[1] comprises of a single-object subset SED1 and a two-object subset SED2, each of which contains 100 images and has pixel-wise annotations. The objects in the images differ from their surroundings by various low-level cues such as intensity, texture, *etc*. Each image was segmented by three subjects and vote for the foreground.

• **ASD [30]**[2] contains 1,000 images with pixel-wise ground-truths. The images are selected from the MSRA-A dataset [29], where only the bounding boxes around salient regions are provided. The accurate salient masks in ASD are created based on object contours.

3.2 Modern Popular SOD Datasets

Recently emerged datasets tend to include more challenging and general scenes with relatively complex backgrounds and contain multiple salient objects. In this section, we review seven most popular and widely-used ones.

• **SOD [142]**[3] contains 300 images from the Berkeley segmentation dataset [147]. Each image is labeled by seven subjects. Many images have more than one salient objects that have low color contrast to the background or touch image boundaries. Pixel-wise annotations are available.

• **MSRA10K [65]**[4], also known as THUS10K, contains 10,000 images selected from MSRA [29] and covers all the 1,000 images in ASD [30]. The images have consistent bounding box labeling, and are further augmented with pixel-level annotations. Due to its large scale and precise annotations, it is widely used to train deep SOD models (see Table 2).

• **ECSSD [57]**[5] is composed by 1,000 images with semantically meaningful but structurally complex natural contents. The ground-truth masks are annotated by 5 participants.

• **DUT-OMRON [58]**[6] contains 5,168 images of relatively complex backgrounds and high content variety. Each image is accompanied with pixel-level ground-truth annotation.

• **PASCAL-S [66]**[7] consists of 850 challenging images selected from the *val* set of PASCAL VOC 2010 [78]. In addition to eye-fixation records, the pixel-wise non-binary salient-object annotations are provided, where the saliency value of a pixel is calculated as the ratio of subjects that select the segment containing this pixel as salient.

• **HKU-IS [26]**[8] contains 4,447 complex scenes that typically contain multiple disconnected objects with relatively diverse spatial distribution, *i.e.*, at least one salient object touches the image boundary. Besides, the similar fore-/back-ground appearance makes this dataset more difficult.

1. http://www.wisdom.weizmann.ac.il/~vision/Seg_Evaluation_DB/dl.html
2. https://ivrlwww.epfl.ch/supplementary_material/RK_CVPR09/
3. http://elderlab.yorku.ca/SOD/
4. https://mmcheng.net/zh/msra10k/
5. http://www.cse.cuhk.edu.hk/leojia/projects/hsaliency
6. http://saliencydetection.net/dut-omron/
7. http://cbi.gatech.edu/salobj/
8. https://i.cs.hku.hk/~gbli/deep_saliency.html

• **DUTS** [79][9], the largest SOD dataset, contains 10,553 training and 5,019 test images. The training images are selected from the ImageNet DET *train/val* set [80], and the test images from the ImageNet *test* set [80] and the SUN dataset [148]. Since 2017, more and more deep SOD models are trained on the training set of DUTS (see Table 2).

3.3 Other Special SOD Datasets

Beyond the "standard" SOD datasets, there are some special ones proposed recently, which are useful to capture different aspects in SOD and lead to related research directions.

• **SOS** [74][10] is created for SOD subitizing [139], *i.e.*, to predict the number of salient objects without an expensive detection process. It contains 6,900 images selected from [78], [80], [91], [148]. Each image is labeled as containing 0, 1, 2, 3 or 4+ salient objects. SOS is randomly split into a *training* (5,520 images) and a *test* set (1,380 images).

• **MSO** [74][11] is a subset of the SOS-*test* covering 1,224 images. It has a more balanced distribution of the number of salient objects. Each object has a bounding box annotation.

• **ILSO** [63][12] has 1,000 images with pixel-wise instance-level saliency annotations and coarse contour labeling, where the benchmark results are generated using MSRNet [63]. Most ILSO images are selected from [26], [34], [58], [74] to reduce ambiguity over the salient object regions.

• **XPIE** [143][13] contains 10,000 images with unambiguous salient objects, which are annotated with pixel-wise ground-truths. It has three subsets: *Set-P* contains 625 images of places-of-interest with geographic information; *Set-I* contains 8,799 images with object tags; and *Set-E* includes 576 images with eye-fixation annotations.

• **SOC** [144][14] has 6,000 images with 80 common categories. Half of the images contain salient objects and the others contain none. Each salient-object-contained image is annotated with instance-level SOD ground-truth, object category, and challenging factors. The non-salient object subset has 783 texture images and 2,217 real-scene images.

• **COCO-CapSal** [104][15] is built from MS COCO [91] and SALICON [73]. The rough salient regions are localized using the human gaze data in SALICON. The 5,265 training images and 1,459 testing images are selected to contain salient regions whose categories are covered by COCO categories and are consistent with the captions. The final salient object mask come from the instance masks in COCO whose IoU scores with the gaze annotations exceed certain thresholds.

• **HRSOD** [114][16] is the first *high-resolution* dataset for SOD. It contains 1,610 training images and 400 testing images collected from website. The pixel-wise ground-truths are labeled by 40 subjects.

4 Evaluation Metrics

There are several ways to measure the agreement between model predictions and human annotations. In this section we review several universally-agreed and popularly adopted measures for SOD model evaluation.

• **Precision-Recall (PR)** is calculated based on the binarilized salient object mask and the ground-truth:

$$\text{Precision} = \frac{\text{TP}}{\text{TP}+\text{FP}}, \quad \text{Recall} = \frac{\text{TP}}{\text{TP}+\text{FN}}, \tag{1}$$

where TP, TN, FP, FN denote true-positive, true-negative, false-positive, and false-negative, respectively. A set of thresholds ($[0-255]$) is applied to binarilize the prediction. Each threshold produces a pair of Precision/Recall value to form a PR curve for describing model performance.

• **F-measure** [30] comprehensively considers both Precision and Recall by computing the weighted harmonic mean:

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

β^2 is empirically set to 0.3 [30] to emphasize more on precision. Instead of reporting the whole F-measure plot, some methods directly use the *maximal* F_β values from the plot, and some others use an adaptive threshold [30], *i.e.*, twice the mean value of the predicted saliency map, to binarilize the saliency map and report the *mean F* value.

• **Mean Absolute Error (MAE)** [32]. Despite their popularity, the above two metrics fail to consider the true negative pixels. MAE is used to remedy this problem by measuring the average pixel-wise absolute error between normalized map $\mathbf{S} \in [0,1]^{W \times H}$ and saliency mask $\mathbf{G} \in \{0,1\}^{W \times H}$:

$$\text{MAE} = \frac{1}{W \times H} \sum_{i=1}^{W} \sum_{j=1}^{H} |\mathbf{G}(i,j) - \mathbf{S}(i,j)|. \tag{3}$$

• **Weighted F_β measure (Fbw)** [149] intuitively generalizes F-measure by alternating the way to calculate the Precision and Recall. It extends the four basic quantities TP, TN, FP and FN to real values, and assigns different weights (ω) to different errors at different locations considering the neighborhood information, defined as:

$$F_\beta^\omega = \frac{(1+\beta^2)\text{Precision}^\omega \times \text{Recall}^\omega}{\beta^2 \text{Precision}^\omega + \text{Recall}^\omega}. \tag{4}$$

• **Structural measure (S-measure)** [150], instead of only address pixel-wise errors, evaluates structural similarity between the real-valued saliency map and the binary ground-truth. S-measure (S) considers object-aware (S_o) and region-aware (S_r) structure similarities:

$$S = \alpha \times S_o + (1-\alpha) \times S_r, \tag{5}$$

where α is empirically set to 0.5.

• **Enhanced-alignment measure (E-measure)** [151] considers global means of the image and local pixel matching simultaneously:

$$Q_\mathbf{S} = \frac{1}{W \times H} \sum_{i=1}^{W} \sum_{j=1}^{H} \phi_\mathbf{S}(i,j), \tag{6}$$

where $\phi_\mathbf{S}$ is the enhanced alignment matrix, which reflects the correlation between $\mathbf{S}$ and $\mathbf{G}$ after subtracting their global means, respectively.

• **Salient Object Ranking (SOR)** [97] is designed for evaluating the rank order of salient objects in salient object subitizing, which is calculated as the normalized Spearman's Rank-Order Correlation between the ground-truth rank order $rg_\mathbf{G}$ and the predicted rank order $rg_\mathbf{S}$:

$$\text{SOR} \triangleq \rho_{rg_\mathbf{G}, rg_\mathbf{S}} = \frac{\text{cov}(rg_\mathbf{G}, rg_\mathbf{S})}{\sigma_{rg_\mathbf{G}} \sigma_{rg_\mathbf{S}}}, \tag{7}$$

9. http://saliencydetection.net/duts/
10. http://cs-people.bu.edu/jmzhang/sos.html
11. http://cs-people.bu.edu/jmzhang/sos.html
12. http://www.sysu-hcp.net/instance-level-salient-object-segmentation/
13. http://cvteam.net/projects/CVPR17-ELE/ELE.html
14. http://mmcheng.net/SOCBenchmark/
15. https://github.com/yi94code/HRSOD
16. https://github.com/zhangludl/code-and-dataset-for-CapSal

TABLE 4
Benchmarking results of 44 state-of-the-art deep SOD models and 3 top-performing classic SOD methods on 6 famous datasets (See §5.1). Here max F, S, and M indicate maximal F-measure, S-measure, and MAE, respectively.

	Dataset	ECSSD [57]			DUT-OMRON [58]			PASCAL-S [66]			HKU-IS [26]			DUTS-test [79]			SOD [142]		
	Metric	max F↑	S↑	M↓	max F↑	S↑	M↓	max F↑	S↑	M↓	max F↑	S↑	M↓	max F↑	S↑	M↓	max F↑	S↑	M↓
2013-14	*HS [34]	.673	.685	.228	.561	.633	.227	.569	.624	.262	.652	.674	.215	.504	.601	.243	.756	.711	.222
	*DRFI [53]	.751	.732	.170	.623	.696	.150	.639	.658	.207	.745	.740	.145	.600	.676	.155	.658	.619	.228
	*wCtr [35]	.684	.714	.165	.541	.653	.171	.599	.656	.196	.695	.729	.138	.522	.639	.176	.615	.638	.213
2015	MCDL [28]	.816	.803	.101	.670	.752	.089	.706	.721	.143	.787	.786	.092	.634	.713	.105	.689	.651	.182
	LEGS [27]	.805	.786	.118	.631	.714	.133	‡	‡	‡	.736	.742	.119	.612	.696	.137	.685	.658	.197
	MDF [26]	.797	.776	.105	.643	.721	.092	.704	.696	.142	.839	.810	.129	.657	.728	.114	.736	.674	.160
2016	ELD [67]	.849	.841	.078	.677	.751	.091	.782	.799	.111	.868	.868	.063	.697	.754	.092	.717	.705	.155
	DHSNet [37]	.893	.884	.060	‡	‡	‡	.799	.810	.092	.875	.870	.053	.776	.818	.067	.790	.749	.129
	DCL [68]	.882	.868	.075	.699	.771	.086	.787	.796	.113	.885	.877	.055	.742	.796	.149	.786	.747	.195
	◇MAP [36]	.556	.611	.213	.448	.598	.159	.521	.593	.207	.552	.624	.182	.453	.583	.181	.509	.557	.236
	CRPSD [76]	.915	.895	.048	-	-	-	.864	.852	.064	.906	.885	.043	-	-	-	-	-	-
	RFCN [77]	.875	.852	.107	.707	.764	.111	.800	.798	.132	.881	.859	.089	.755	.859	.090	.769	.794	.170
2017	MSRNet [63]	.900	.895	.054	.746	.808	.073	.828	.838	.081	‡	‡	‡	.804	.839	.061	.802	.779	.113
	DSS [38]	.906	.882	.052	.737	.790	.063	.805	.798	.093	‡	‡	‡	.796	.824	.057	.805	.751	.122
	†WSS [79]	.879	.811	.104	.725	.730	.110	.804	.744	.139	.878	.822	.079	.878	.822	.079	.807	.675	.170
	DLS [81]	.826	.806	.086	.644	.725	.090	.712	.723	.130	.807	.799	.069	-	-	-	-	-	-
	NLDF [82]	.889	.875	.063	.699	.770	.080	.795	.805	.098	.888	.879	.048	.777	.816	.065	.808	.889	.125
	Amulet [84]	.905	.894	.059	.715	.780	.098	.805	.818	.100	.887	.886	.051	.750	.804	.085	.773	.757	.142
	FSN [85]	.897	.884	.053	.736	.802	.066	.800	.804	.093	.884	.877	.044	.761	.808	.066	.781	.755	.127
	SBF [86]	.833	.832	.091	.649	.748	.110	.726	.758	.133	.821	.829	.078	.657	.743	.109	.740	.708	.159
	SRM [87]	.905	.895	.054	.725	.798	.069	.817	.834	.084	.893	.887	.046	.798	.836	.059	.792	.741	.128
	UCF [88]	.890	.883	.069	.698	.760	.120	.787	.805	.115	.874	.875	.062	.742	.782	.112	.763	.753	.165
2018	RADF [89]	.911	.894	.049	.761	.817	.055	.800	.802	.097	.902	.888	.039	.792	.826	.061	.804	.757	.126
	BDMP [93]	.917	.911	.045	.734	.809	.064	.830	.845	.074	.910	.907	.039	.827	.862	.049	.806	.786	.108
	DGRL [95]	.916	.906	.043	.741	.810	.063	.830	.839	.074	.902	.897	.037	.805	.842	.050	.802	.771	.105
	PAGR [96]	.904	.889	.061	.707	.775	.071	.814	.822	.089	.897	.887	.048	.817	.838	.056	.761	.716	.147
	RSDNet [97]	.880	.788	.173	.715	.644	.178	‡	‡	‡	.871	.787	.156	.798	.720	.161	.790	.668	.226
	ASNet [98]	.925	.915	.047	‡	‡	‡	.848	.861	.070	.912	.906	.041	.806	.843	.061	.801	.762	.121
	PiCANet [39]	.929	.916	.035	.767	.825	.054	.838	.846	.064	.913	.905	.031	.840	.863	.040	.814	.776	.096
	†C2S-Net [99]	.902	.896	.053	.722	.799	.072	.827	.839	.081	.887	.889	.046	.784	.831	.062	.786	.760	.124
	RAS [100]	.908	.893	.056	.753	.814	.062	.800	.799	.101	.901	.887	.045	.807	.839	.059	.810	.764	.124
2019	AFNet [103]	.924	.913	.042	.759	.826	.057	.844	.849	.070	.910	.905	.036	.838	.867	.046	.809	.774	.111
	BASNet [62]	.931	.916	.037	.779	.836	.057	.835	.838	.076	.919	.909	.032	.838	.866	.048	.805	.769	.114
	CapSal [104]	.813	.826	.077	.535	.674	.101	.827	.837	.073	.842	.851	.057	.772	.818	.061	.669	.694	.148
	CPD [105]	.926	.918	.037	.753	.825	.056	.833	.848	.071	.911	.905	.034	.840	.869	.043	.814	.767	.112
	MLSLNet [106]	.917	.911	.045	.734	.809	.064	.835	.844	.074	.910	.907	.039	.828	.862	.049	.806	.786	.108
	†MWS [107]	.859	.827	.099	.676	.756	.108	.753	.768	.134	.835	.818	.086	.720	.759	.092	.772	.700	.170
	PAGE-Net [109]	.926	.910	.037	.760	.819	.059	.829	.835	.073	.910	.901	.031	.816	.848	.048	.795	.763	.108
	PS [110]	.930	.918	.041	.789	.837	.061	.837	.850	.071	.913	.907	.038	.835	.865	.048	.824	.800	.103
	PoolNet [111]	.937	.926	.035	.762	.831	.054	.858	.865	.065	.923	.919	.030	.865	.886	.037	.831	.788	.106
	BANet [112]	.939	.924	.035	.782	.832	.059	.847	.852	.070	.923	.913	.032	.858	.879	.040	.842	.791	.106
	EGNet-R [113]	.936	.925	.037	.777	.841	.053	.841	.852	.074	.924	.918	.031	.866	.887	.039	.854	.802	.099
	HRSOD-DH [114]	.911	.888	.052	.692	.762	.065	.810	.817	.079	.890	.877	.042	.800	.824	.050	.735	.705	.139
	JDFPR [115]	.915	.907	.049	.755	.821	.057	.827	.841	.082	.905	.903	.039	.792	.836	.059	.792	.763	.123
	SCRN [116]	.937	.927	.037	.772	.836	.056	.856	.869	.063	.921	.916	.034	.864	.885	.040	.826	.787	.107
	SSNet [117]	.889	.867	.046	.708	.773	.056	.793	.807	.072	.876	.854	.041	.769	.784	.049	.713	.700	.118
	TSPOANet [40]	.919	.907	.047	.749	.818	.061	.830	.842	.078	.909	.902	.039	.828	.860	.049	.810	.772	.118

* Non-deep learning model. † Weakly-supervised model. ◇ Bounding-box output. ‡ Training on subset. - Results not available.

where $\mathrm{cov}(\cdot)$ calculates the covariance, and $\sigma_{\{\cdot\}}$ denotes the standard deviation.

5 Benchmarking and Analysis

5.1 Overall Performance Benchmarking Results

Table 4 shows performances of 44 state-of-the-art deep SOD models and 3 top-performing classic SOD methods on 6 most popular datasets. Three evaluation metrics, *i.e. maximal* F_β [30], S-measure [150] and MAE [32] are used for assessing pixel-wise saliency prediction accuracy and the structure similarity of salient regions. All the 47 benchmarked models are representative, and have publicly available implementations or saliency prediction results on the 6 selected datasets.

• **Deep v.s. Non-deep learning.** Comparing the 3 top-performing heuristic SOD methods with deep ones in Table 4, we observe that deep models consistently improve the prediction performances by a large margin. This confirms the strong learning ability of deep neural networks.

• **Performance evaluation of deep SOD.** The performances of visual saliency computation models gradually increased over time since 2015. Among the deep models, MAP [36] proposed in 2016 performs least impressive, since it only outputs the bounding boxes of the salient objects. This demonstrates the need for accurate annotations for more effective training and more reliable evaluations [30], [152].

5.2 Attribute-based Evaluation

Applying DNN on SOD has brought significant performance gain, while the challenges associated with foreground and background attributes remain to be conquered. In this section, we conduct a detailed attribute-based analysis on the performance of selected SOD approaches.

5.2.1 Models, Benchmark and Attributes

We choose three top-performing heuristic models, *i.e.* HS [34], DRFI [53] and wCtr [35], and three recent famous deep methods, *i.e.* DGRL [95], PAGR [96] and PiCANet [39]

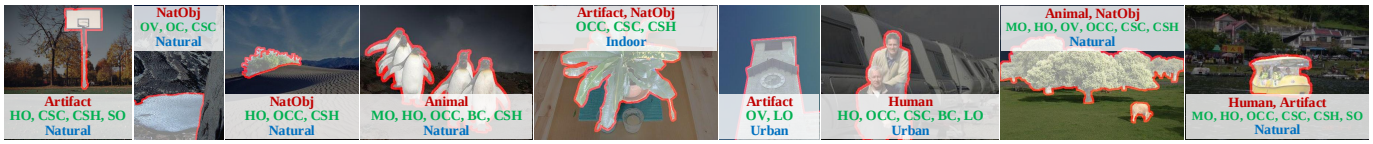

Fig. 4. Sample images from the hybrid benchmark consisting of images randomly selected from 6 SOD datasets. Saliently regions are uniformly highlighted. Corresponding attributes are listed. See §5.2 for more detailed descriptions.

TABLE 6
Attribute-based study w.r.t. salient object categories, challenges and scene categories. $(\cdot)$ indicates the percentage of the images with a specific attribute. *ND-avg* indicates the average score of three heuristic models: HS [34], DRFI [53] and wCtr [35]. *D-avg* indicates the average score of three deep learning models: DGRL [95], PAGR [96] and PiCANet [39]. (Best in **red**, worst with **underline**; See §5.2 for details).

Metric	Method	Salient object categories				Challenges									Scene categories		
		Human (26.61)	*Animal* (38.44)	*Artifact* (45.67)	*NatObj* (10.56)	$\mathcal{MO}$ (11.39)	$\mathcal{HO}$ (66.39)	$\mathcal{OV}$ (28.72)	$\mathcal{OCC}$ (46.50)	$\mathcal{CSC}$ (40.44)	$\mathcal{BC}$ (47.22)	$\mathcal{CSH}$ (74.11)	$\mathcal{SO}$ (21.61)	$\mathcal{LO}$ (12.61)	*Indoor* (20.28)	*Urban* (22.22)	*Natural* (57.50)
max $\mathbb{F}\uparrow$	*HS [34]	.587	.650	.636	.704	.663	.637	.631	.645	.558	.647	.629	.493	.737	.594	.627	.650
	*DRFI [53]	.635	.692	.673	.713	.674	.688	.658	.675	.599	.662	.677	.566	.747	.609	.661	.697
	*wCtr [35]	.557	.621	.624	.682	.639	.625	.605	.620	.522	.612	.606	.469	.689	.578	.613	.618
	DGRL [95]	.820	.881	.830	.728	.783	.846	.829	.830	.781	.842	.834	.724	.873	.800	.848	.840
	PAGR [96]	.834	.890	.787	.725	.743	.819	.778	.809	.770	.797	.822	.760	.802	.788	.796	.828
	PiCANet [39]	.840	.897	.846	.669	.791	.861	.843	.845	.797	.848	.850	.763	.889	.806	.862	.859
	*ND-avg	**.593**	.654	.644	**.700**	**.659**	.650	.631	.647	.560	.640	.637	**.509**	.724	**.594**	.634	**.655**
	D-avg	.831	**.889**	.821	**.708**	.772	**.842**	.817	.828	.783	.829	.836	**.749**	.855	**.798**	.836	**.842**

* Non-deep learning model.

TABLE 5
Descriptions of attributes. See §5.2 for more details.

Attr	Description
$\mathcal{MO}$	**Multiple Objects.** There exist more than two salient objects.
$\mathcal{HO}$	**Heterogeneous Object.** Salient object regions have distinct colors or illuminations.
$\mathcal{OV}$	**Out-of-view.** Salient object is partially clipped by the image boundaries.
$\mathcal{OCC}$	**Occlusion.** Salient object is occluded by other objects.
$\mathcal{CSC}$	**Complex Scene.** Background region contains confusing objects or rich details.
$\mathcal{BC}$	**Background Clutter.** Foreground and background regions around the salient object boundaries have similar colors (χ^2 between RGB histograms less than 0.9).
$\mathcal{CSH}$	**Complex Shape.** Salient object contains thin parts or holes.
$\mathcal{SO}$	**Small Object.** Ratio between salient object area and image area is less than 0.1.
$\mathcal{LO}$	**Large Object.** Ratio between salient object area and image area is larger than 0.5.

to perform attribute-based analysis. All of the deep models are trained on the same dataset, *i.e.*, DUTS [79].

We construct a *hybrid benchmark* consists of 1,800 images randomly selected from 6 SOD datasets (300 each), namely SOD [142], ECSSD [57], DUT-OMRON [58], PASCAL-S [66], HKU-IS [26] and the test set of DUTS [79]. Please be noted that this benchmark will also be used in §5.3 and §5.4.

Inspired by [66], [144], [153], we annotate each image with a rich set of attributes considering salient object categories, challenges and scene categories. The **salient objects** are categorized into *Human*, *Animal*, *Artifact* and *NatObj* (Natural Objects), where *NatObj* includes natural objects such as fruit, plant, mountains, icebergs, water (*e.g.* lakes, streaks), *etc*. The **challenges** describe factors that often bring difficulties to SOD, such as occlusion, background cluster, complex shape and object scale, as summarized in Table 5. The **scene** of images includes *Indoor*, *Urban* and *Natural*, where the last two indicate different outdoor environments. Please note that the attributes are not mutually exclusive. Some sample images are shown in Fig. 4.

5.2.2 Analysis

• **'Easy' and 'Hard' object categories.** Deep and non-deep SOD models view object categories differently (Table 6). For deep SOD methods, *NatObj* is clearly the most challenging one which is probably due to small amount of training samples. *Animal* appears to be the easiest even though the portion is not the largest, mainly due to its specific semantics. By contrast, heuristic methods are generally good at segmenting dominant *NatObj*, and are short at *Human*, which may be caused by the lack of high-level semantic learning.

• **Most and least challenging factors.** Table 6 shows that deep methods predict $\mathcal{HO}$ with higher precision thanks to the powerful ability of DNN to extract high-level semantics. Heuristic methods perform well for $\mathcal{MO}$, since hand-craft local features contribute to distinguishing the boundaries of different objects. Both deep and non-deep methods achieve lower performance for $\mathcal{SO}$ due to the inherent difficulty to precisely label small scale objects.

• **Most and least difficult scenes.** Deep and heuristic methods perform similarly when facing different scenes (Table 6). For both types of methods, *Natural* is the easiest, which is reasonable since it takes up more than half of the samples. *Indoor* is harder than *Urban* since the former usually contains a plunge of objects within a limited space, and often suffers from highly unevenly distributed illuminations.

• **Additional advantages of deep models.** First, as show in Table 6, deep models achieve great improvement on two object categories, *Animal* and *Artifact*, showing its ability to learn from large amount examples. Second, deep models are less sensitive to incomplete object shape ($\mathcal{HO}$ and $\mathcal{OV}$), since they learn high-level semantics. Third, deep models narrow the gap between different scene categories (*Indoor* v.s. *Natural*), showing robustness against various backgrounds.

• **Top and bottom predictions.** From Table 7, heuristic methods perform better for hybrid natural objects (*NatObj*) than for *Human*. On the contrary, deep methods seem to suffer from *NatObj* besides *Animal*. For challenge factors, both deep and heuristic methods meet problems at handling complex scenes ($\mathcal{CSC}$) and small objects ($\mathcal{SO}$). Lastly, heuristic methods perform worst on outdoor scenes (*i.e.*, *Urban* and *Natural*), while deep ones are relatively bad at predicting saliency for *Indoor* scene.

TABLE 7
Attribute statistics of top and bottom 100 images based on F-measure. $(\cdot)$ indicates the percentage of the images with a specific attribute. *ND-avg* indicates the average results of three heuristic models: HS [34], DRFI [53] and wCtr [35]. *D-avg* indicates the average results of three deep models: DGRL [95], PAGR [96] and PiCANet [39]. (Two largest changes in by **red** if positive, **blue** if negative; See §5.2)

Method	Cases	Salient object categories				Challenges									Scene categories		
		Human (26.61)	*Animal* (38.44)	*Artifact* (45.67)	*NatObj* (10.56)	$\mathcal{MO}$ (11.39)	$\mathcal{HO}$ (66.39)	$\mathcal{OV}$ (28.72)	$\mathcal{OCC}$ (46.50)	$\mathcal{CSC}$ (40.44)	$\mathcal{BC}$ (47.22)	$\mathcal{CSH}$ (74.11)	$\mathcal{SO}$ (21.61)	$\mathcal{LO}$ (12.61)	*Indoor* (20.28)	*Urban* (22.22)	*Natural* (57.50)
ND-avg	Best (%)	13.00	25.00	46.00	27.00	5.00	61.00	12.00	26.00	10.00	20.00	63.00	5.00	18.00	17.00	6.00	12.00
	change	**-13.61**	-13.44	+0.33	**+14.44**	-6.39	-5.39	-16.72	-20.50	**-30.44**	**-27.22**	-11.11	-16.61	+5.39	-3.28	**-16.22**	**-45.50**
	Worst (%)	36.00	30.00	41.00	5.00	6.00	54.00	15.00	34.00	70.00	31.00	71.00	76.00	0.00	22.00	37.00	37.00
	change	**+9.39**	**-8.44**	-4.67	-5.56	-5.39	-12.39	-13.72	-12.50	**+29.56**	-16.22	-3.11	**+54.39**	-12.61	+1.72	**+14.78**	**-20.50**
D-avg	Best (%)	24.00	30.00	49.00	17.00	3.00	69.00	33.00	28.00	26.00	35.00	49.00	2.00	18.00	24.00	23.00	53.00
	change	-2.61	**-8.44**	+3.33	**+6.44**	-8.39	+2.61	+4.28	-18.50	-14.44	-12.22	**-25.11**	**-19.61**	+5.39	**+3.72**	+0.78	**-4.50**
	Worst (%)	30.00	10.00	49.00	33.00	20.00	52.00	28.00	46.00	70.00	42.00	59.00	50.00	3.00	32.00	23.00	45.00
	change	+3.39	**-28.44**	+3.33	**+22.44**	+8.61	-14.39	-0.72	-0.50	**+29.56**	-5.22	-15.11	**+28.39**	-9.61	**+11.72**	+0.78	**-12.50**

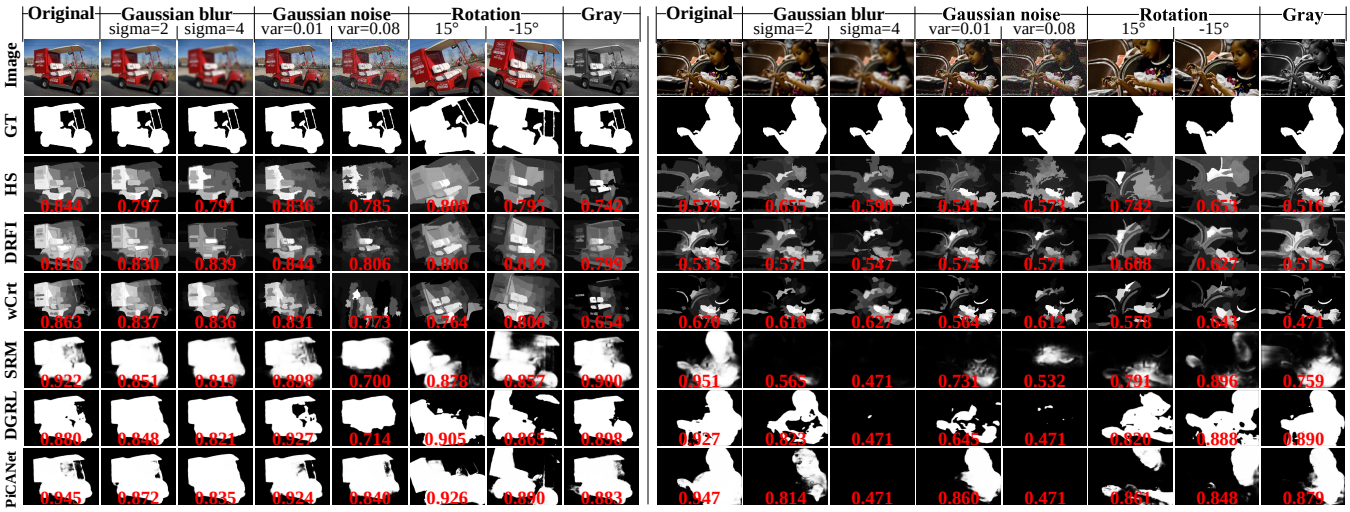

Fig. 5. Examples of saliency prediction under various input perturbations. The max F values are denoted using red. See §5.3 for more details.

TABLE 8
Input perturbation study on the **hybrid benchmark**. *ND-avg* indicates the average score of three heuristic models: HS [34], DRFI [53] and wCtr [35]. *D-avg* indicates the average score of three deep learning models: SRM [87], DGRL [95] and PiCANet [39]. See §5.3 for details. (Best in **red**, worst with **underline**).

Metric	Method	*Original*	*Gaus. blur* (σ=)		*Gaus. noise* (var=)		*Rotation*		*Gray*
			2	4	0.01	0.08	15°	−15°	
max $F\uparrow$	*HS [34]	.600	-.012	-.096	-.022	-.057	+.015	+.009	-.104
	*DRFI [53]	.670	-.040	-.103	-.035	-.120	-.009	-.009	-.086
	*wCtr [35]	.611	+.006	-.000	-.024	-.136	-.004	-.003	-.070
	SRM [87]	.817	-.090	-.229	-.025	-.297	-.028	-.029	-.042
	DGRL [95]	.831	-.088	-.365	-.050	-.402	-.031	-.022	-.026
	PiCANet [39]	.848	-.048	-.175	-.014	-.148	-.005	-.008	-.039
	**ND-avg*	.627	-.015	-.066	-.027	**-.104**	**-.000**	-.001	-.087
	D-avg	.832	-.075	-.256	-.041	**-.282**	-.021	**-.020**	-.037

* Non-deep learning model.

5.3 Influences of Input Perturbations

The robustness of a model lies in its stability against corrupted inputs. Randomly perturbed images, such as images with additive noise or blurriness, would possibly result in worse outputs. On the other hand, the recently emerged adversarial examples, *i.e.* the maliciously constructed inputs that fool the trained models, can degrade the performance of the deep image classification models significantly. In this section, we analyze the influences of general input perturbations. In §5.4, we will look into how the manually designed adversarial examples affect the deep SOD models.

The experimented input perturbations include *Gaussian blur*, *Gaussian noise*, *Rotation*, and *Gray*. For blurring, we blur the images using Gaussian kernels with sigma being 2 or 4. For noise, we select two variance values, *i.e.* 0.01 and 0.08 to cover both tiny and medium magnitudes. For rotation, we rotate the images for $+15°$ and $-15°$, respectively, and cut out the largest box with the original aspect ratio. The gray images are generated using Matlab `rgb2gray` function.

As in §5.2, we choose three popular heuristic models [34], [35], [53] and three deep methods [39], [87], [95] for studying the input perturbation influences. Table 8 shows the results. Overall, the heuristic methods are less sensitive towards input perturbations compared with deep methods, largely due to the robustness of hand-craft super-pixel level features. Specifically, heuristic methods are rarely affected by *Rotation*, but perform worse for strong *Gaussian blur*, strong *Gaussian noise* and the *Gray* effect. Deep methods suffer the most for *Gaussian blur* and strong *Gaussian noise*, which greatly impair the of information in the reception fields of shallow layers. Deep methods are relatively robust against *Rotation* due to spatial pooling in feature hierarchy.

5.4 Adversarial Attacks Analysis

DNN models have achieved dominant performance in various tasks, including SOD. However, modern DNNs are shown to be surprisingly susceptible to adversarial attacks, where visually imperceptible perturbations of input im-

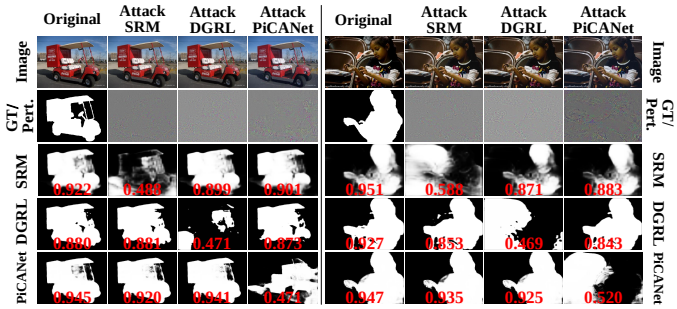

Fig. 6. Adversarial examples for saliency prediction under adversarial perturbations of different target networks. Adversarial perturbations are magnified by 10 for better visualization. Red for max F. See §5.4.

TABLE 9
Results for adversarial attack experiments. max F↑ on the **hybrid benchmark** is presented when exerting adversarial perturbations from different models. See § 5.4 for details. (Worst with **underline**)

Attack from	SRM [87]	DGRL [95]	PiCANet [39]
None	.817	.831	.848
SRM [87]	**.263**	.780	.842
DGRL [95]	.778	**.248**	.844
PiCANet [39]	.772	.799	**.253**

ages would lead to completely different predictions [154]. Though being intensively studied in classification tasks, adversarial attacks in SOD are significantly under-explored.

In this section, we study the robustness of deep SOD methods by performing adversarial attack on three representative deep models. We also analyze the transferability of the adversarial examples targeted on different SOD models. We expect our observations to shed light on the adversarial attacks and defenses of SOD, and lead to better understanding of model vulnerabilities.

5.4.1 Robustness of SOD against Adversarial Attacks

We choose three representative deep models, *i.e.* SRM [87], DGRL [95] and PiCANet [39], to study the robustness against adversarial attack. All the three models are trained on DUTS [79]. We experiment with the ResNet [120] backbone version of the three models. The experiment is conducted on the hybrid benchmark introduced in §5.2.

Since SOD can be viewed as a special case of semantic segmentation with two predefined categories, we resort to an adversarial attack algorithm designed for semantic segmentation, *Dense Adversary Generation* (DAG) [155], for measuring the robustness of deep SOD models.

Exemplar adversarial cases are shown in Fig. 6. Quantitative results are listed in Table 9. As can be seen, small adversarial perturbations can cause drastic performance drops for all of the three models. More often than not, such adversarial examples result in worse predictions compared with random exerted noises (See Tables 8 and 9).

5.4.2 Transferability across Networks

Transferability refers to the ability of adversarial examples generated against one model to mislead another model without any modification [156], which is widely used for black-box attack against real-world system. Given this, we analyze the transferability in SOD by attacking one model using the adversarial perturbations generated for another.

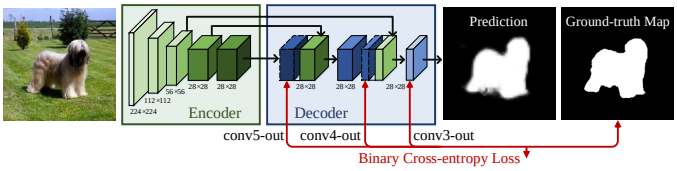

Fig. 7. Network architecture of the SOD model used in cross-dataset generalization evaluation. See §5.5 for more detailed descriptions.

TABLE 10
Results for cross-dataset generalization experiment. max F↑ for saliency prediction when training on one dataset (rows) and testing on another (columns). "Self" refers to training and testing on the same dataset (same as diagonal). "Mean Others" indicates average performance on all except self. See §5.5 for details.

Train on: \ Test on:	MSRA-10K [65]	ECSSD [57]	DUT-OMRON [58]	HKU-IS [26]	DUTS [79]	SOC [144]	Self	Mean others	Percent drop↓
MSRA10K [65]	**.875**	.818	.660	.849	.671	.617	.875	.723	17%
ECSSD [57]	.844	**.831**	.630	.833	.646	.616	.831	.714	14%
DUT-OMRON [58]	.795	.752	**.673**	.779	.623	.567	.673	.703	-5%
HKU-IS [26]	.857	.838	.695	**.880**	.719	.639	.880	.750	15%
DUTS [79]	.857	.834	.647	.860	**.665**	.654	.665	.770	-16%
SOC [144]	.700	.670	.517	.666	.514	**.593**	.593	.613	-3%
Mean others	.821	.791	.637	.811	.640	.614	-	-	-

The evaluation of transferability among 3 studied models (SRM [87], DGRL [95] and PiCANet [39]) is shown in Table 9. It shows that the DAG attack rarely transfers among different SOD networks. This may be because that the spatial distributions of the attacks are very distinctive among different SOD models.

5.5 Cross-dataset Generalization Evaluation

Datasets is important for training and evaluating deep models. In this section, we study the generalization and hardness of several main-stream SOD datasets by performing cross-dataset analysis [157], *i.e.*, to train a representative simple SOD model on one dataset, and test it on the other.

The simple SOD model is implemented as a popular bottom-up/top-down architecture, where the encoder part is borrowed from VGG16 [119], and the decoder part consists of three convolutional layers for gradually making more precise pixel-wise saliency predictions. To increase the output resolution, the strides of the max-pooling layer in the 4-th block is decreased to 1, the dilation rates of the 5-th convolutional block is modified to 2, and the *pool5* layer is excluded. The side output is obtained by a $Conv(1\times1, 1)$ layer with *Sigmoid* activation and deeply supervised. The final map comes from the 3-rd decoder layer. See Fig. 7.

For this study we pick six representative datasets [26], [57], [58], [65], [79], [144]. For each dataset, we train the SOD model with 800 randomly selected training images and test it on 200 other validation images. Please be noted that $1,000$ is the maximum possible total number considering the size of the smallest selected dataset, ECSSD [57].

Table 10 summarizes the results of cross-dataset generalization using max F. Each column shows the performance of all the trained models testing on one dataset, indicating the hardness of the tested dataset. Each row shows the performance of one trained model testing on all the datasets, indicating the generalization ability of the training dataset. We find that SOC [144] is the most difficult dataset

(lowest column *Mean others* 0.619). This may be because that SOC [144] is collected to have distinctive location distributions compared with other datasets, and may contain extremely large or small salient objects. MSRA10K [65] appears to be the easiest dataset (highest column *Mean others* 0.811), and generalizes the worst (highest row *Percent drop* 17%). DUTS [79] is shown to have the best generalization ability (lowest row *Percent drop* −16%).

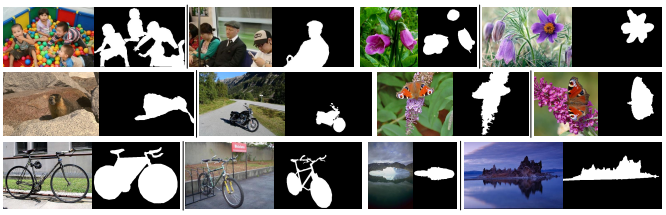

Fig. 8. Examples for annotation inconsistency. Each row shows two exemplar image pairs. See §6.2 for more detailed descriptions.

6 Discussions

6.1 Model Design

In the following we discuss several factors and directions that are important for SOD model design.

• **Feature aggregation.** Efficient aggregation of hierarchical deep features are significant for pixel-wise labeling tasks since it is believed to be beneficial to integrate 'multi-scale' abstracted information. Existing SOD methods have brought various strategies for feature aggregation, such as multi-stream/multi-resolution fusion [63], top-down bottom-up fusion [37] or side-output fusion [38], [84], [89]. Fusion with features from other domains, *e.g.* fixation prediction, may also enhance the feature representation [85]. It's also promising to learn from the feature aggregation methodology of other closely related research tasks such as semantic segmentation [158]–[160].

• **Loss function.** The elaborate design of loss functions also plays an important role in training more effective models. In [98], loss functions derived from SOD evaluation metrics are used for capturing quality factors and have been empirically shown to improve saliency prediction performance. Other recent works [62], [161] proposes to directly optimize the mean intersection-over-union loss. Designing suitable loss functions for SOD is an important consideration for further improving model performance.

• **Network topology.** Network topology determines the within-network information flow that directly affects training difficulty and parameter usage, such as various skip connection designs [120], [162]. As an evidence, for a same SOD method, using ResNet [120] as the backbone usually obtains better performance than the one based on VGG [119]. Besides manually determining the network topology all the way up, a promising direction is automated machine learning (AutoML), which aims to find the best performing algorithms with least possible human intervention. For example, *Neural Architecture Search (NAS)* [163] is able to generate competitive models for image classification and language modeling from scratch. The existing well-designed network topologies and the AutoML technologies all provide insights for constructing novel and effective SOD architectures in future.

• **Dynamic inference.** The rich redundancy among DNN features facilitates its robustness against perturbed inputs, while inevitably introducing extra computational cost during inference. Besides using some *static* methods such as kernel decomposition [164] or parameter pruning [165], some studies investigate on varying the amount of computation *dynamically* during testing, either by selectively activating part of the network [166], [167], or performing early stop [168]. Compared with *static* methods, these dynamic ones improve the efficiency without decreasing network parameters, thus is prone to be robust against basic adversarial attacks [167]. For SOD model design, incorporating reasonable and effective dynamic structure is promising for improving both efficiency and performance.

6.2 Dataset Collection

• **Data selection bias.** Most existing SOD datasets collect images that contain salient objects in relatively clean background, while discarding images that do not contain any salient objects, or whose backgrounds are too clustered. However, real-world applications usually face with much more complicated situations, which can cause serious trouble to SOD models trained on these datasets. Thus, creating datasets to faithfully reflect the real world challenges is crucial for improving the generalization ability of SOD [43].

• **Annotation inconsistency.** Though existing SOD datasets play an important role in training and evaluating modern SOD models, the inconsistencies among different SOD datasets shall not be neglected or overlooked. The intra-dataset inconsistencies are mainly due to separate subjects and rules/conditions during dataset annotation. See Fig. 8.

• **Coarse v.s. fine annotation.** For data-driven learning, the labeling quality is crucial for training reliable SOD models and evaluating faithfully them. The first improvement of SOD annotation quality is to replace the bounding-boxes with pixel-wise masks for denoting the salient objects [30], [152], which greatly boost the performance of SOD models. However, the precision of the pixel-wise labeling varied across datasets (see the bicycle in Fig. 8). Some researchs have focus on the effect of fine-labeled training data on model performance [144], [169], [170].

• **Domain-specific SOD datasets.** SOD has wide applications such as autonomous vehicles, video games, medical image processing, *etc*. Due to different visual appearances and semantic components, the saliency mechanism in these applications can be quite different from that in conventional natural images. Thus, domain-specific datasets might benefit SOD in certain applications, as have been observed in FP for crowds [171], webpages [172] or during driving [173].

6.3 Saliency Ranking and Relative Saliency

Traditionally, the salient object generally refers to the most salient object or region in a scene. However, this 'simple' definition may be confusing for images with multiple salient objects. Thus, how to assess the saliency of co-existing objects or regions is import for designing SOD models and annotating SOD datasets. One possible solution is to rank the saliency of objects or regions using fixation data [66]. Another solution is to vote the relative saliency of multiple salient instances by several observers [97].

6.4 Relation with Fixations

Both FP and SOD closely relate to the concept of *visual saliency* in the field of computer vision. FP dates back to early 1990s [52] which aims to predict the fixation points that would the focus of the first glance by human viewers. SOD has a slightly shorter history dating back to [29], [30], and attempts to identify and segment the salient object(s) in the scene. FP is directly derived from the cognition and psychology community, while SOD appears more 'computer vision' driven by object-level applications. The generated saliency maps of the two are actually remarkably different due to the distinct purposes in saliency detection.

The strong correlation between FP and SOD has been explored in history [43], [66], [174]–[176]. While only a few models consider FP and SOD tasks simultaneously [12], [66], [72], [85]. Exploring the rationale behind the relation of SOD and FP is a promising direction as it helps to better understand the visual selective mechanism of humans.

6.5 Improving SOD with Semantics

Semantics is of crucial importance in high-level vision tasks such as semantic segmentation, object detection, object class discovery, *etc*. Some efforts have been devoted to facilitate SOD with semantic information [77], [117]. Besides pre-training SOD models with segmentation dataset [77], or utilizing multi-task learning to concurrently train SOD with semantic segmentation [117], a promising direction is to enhance saliency features by incorporating segmentation features as done in some object detection methods, either through concatenation [177] or using activation [178].

6.6 Learning SOD in a Un-/Weakly-Supervised Manner

Most of the modern deep SOD methods are trained in a fully-supervised manner with a plethora of fine-annotated pixel-wise ground-truths. However, it is highly-costly and time-consuming to construct a large-scale SOD dataset. Thus, learning SOD in an un-supervised or weakly-supervised manner is of great value in both research and real-world application. Though a few efforts have been made, *i.e.*, resorting to category-level labels [79], [92], [101], leveraging pseudo pixel-wise annotations [86], [90], [94], [99], [107], there is still a large gap regarding the fully-supervised ones. Therefore we can expect a flurry of innovation towards this direction in the upcoming years.

6.7 Applying SOD in Real-World Scenarios

DNNs are generally designed to be deep and complicated in order to increase the model capacity and achieve better performance in various tasks. However, more ingenuous and light-weighted network architectures are required to fulfill the requirements of mobile and embedded applications such as robotics, autonomous driving, augmented reality, *etc*. The degradation of accuracy and generalization capability due to model scale deduction is desired to be minimum.

To facilitate the application of SOD in real-world scenarios, it is considerable to utilize model compression [179] or knowledge distillation [180], [181] techniques to learn compact and fast SOD models with competitive prediction accuracy. Such compression techniques have shown the effectiveness in improving the generalization and alleviating under-fitting when training faster models for object detection [182], a more challenging task than image classification. It is worthy of exploring compressing SOD models with these techniques for fast and accurate saliency prediction.

7 Conclusion

In this paper we present, to the best of our knowledge, the first comprehensive review of SOD with focus on deep learning techniques. We first carefully review and organize deep learning-based SOD models from several different perspectives, including network architecture, level of supervision, *etc*. We then summarize popular SOD datasets and evaluation criteria, and compile a thorough performance benchmarking of major SOD methods. Next, we investigate several previously under-explored issues with novel efforts on benchmarking and baselines. In particular, we perform attribute-based performance analysis by compiling and annotating a new dataset and testing several representative SOD methods. We also study the robustness of SOD methods w.r.t. input perturbations. Moreover, for the first time in SOD, we investigate the robustness and transferability of deep SOD models w.r.t. adversarial attacks. In addition, we assess the generalization and hardness of existing SOD datasets through cross-dataset generalization experiment. We finally look through several open issues and challenges of SOD in deep learning era, and provide insightful discussions on possible research directions in future.

All the saliency prediction maps, our constructed dataset, annotations, and codes for evaluation are made publicly available at https://github.com/wenguanwang/SODsurvey. In conclusion, SOD has achieved notable progress thanks to the striking development of deep learning techniques, yet it still has significant room for improvement. We expect this survey to provide an effective way to understand state-of-the-arts and, more importantly, insights for future exploration in SOD.

References

[1] J.-Y. Zhu, J. Wu, Y. Xu, E. Chang, and Z. Tu, "Unsupervised object class discovery via saliency-guided multiple class learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 4, pp. 862–875, 2015.

[2] F. Zhang, B. Du, and L. Zhang, "Saliency-guided unsupervised feature learning for scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 4, pp. 2175–2184, 2015.

[3] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *Proc. ACM Int. Conf. Mach. Learn.*, 2015, pp. 2048–2057.

[4] H. Fang, S. Gupta, F. Iandola, R. K. Srivastava, L. Deng, P. Dollár, J. Gao, X. He, M. Mitchell, J. C. Platt *et al.*, "From captions to visual concepts and back," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1473–1482.

[5] A. Das, H. Agrawal, L. Zitnick, D. Parikh, and D. Batra, "Human attention in visual question answering: Do humans and deep networks look at the same regions?" *Computer Vision and Image Understanding*, vol. 163, pp. 90–100, 2017.

[6] Z. Ren, S. Gao, L.-T. Chia, and I. W.-H. Tsang, "Region-based saliency detection and its application in object recognition." *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 5, pp. 769–779, 2014.

[7] D. Zhang, D. Meng, L. Zhao, and J. Han, "Bridging saliency detection to weakly supervised object detection based on self-paced curriculum learning," in *International Joint Conferences on Artificial Intelligence*, 2016.

[8] W. Wang, J. Shen, R. Yang, and F. Porikli, "Saliency-aware video object segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 1, pp. 20–33, 2018.

[9] H. Song, W. Wang, S. Zhao, J. Shen, and K.-M. Lam, "Pyramid dilated deeper convlstm for video salient object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2018.

[10] Y. Wei, J. Feng, X. Liang, M.-M. Cheng, Y. Zhao, and S. Yan, "Object region mining with adversarial erasing: A simple classification to semantic segmentation approach," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017.

[11] Y. Wei, X. Liang, Y. Chen, X. Shen, M.-M. Cheng, J. Feng, Y. Zhao, and S. Yan, "STC: A simple to complex framework for weakly-supervised semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 11, pp. 2314–2320, 2017.

[12] X. Wang, S. You, X. Li, and H. Ma, "Weakly-supervised semantic segmentation by iteratively mining common object features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018.

[13] R. Zhao, W. Ouyang, and X. Wang, "Unsupervised salience learning for person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 3586–3593.

[14] S. Bi, G. Li, and Y. Yu, "Person re-identification using multiple experts with random subspaces," *Journal of Image and Graphics*, vol. 2, no. 2, 2014.

[15] Y.-F. Ma, L. Lu, H.-J. Zhang, and M. Li, "A user attention model for video summarization," in *Proc. ACM Int. Conf. Multimedia*, 2002, pp. 533–542.

[16] D. Simakov, Y. Caspi, E. Shechtman, and M. Irani, "Summarizing visual data using bidirectional similarity," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2008, pp. 1–8.

[17] J. Han, E. J. Pauwels, and P. De Zeeuw, "Fast saliency-aware multi-modality image fusion," *Neurocomputing*, vol. 111, pp. 70–80, 2013.

[18] P. L. Rosin and Y.-K. Lai, "Artistic minimal rendering with lines and blocks," *Graphical Models*, vol. 75, no. 4, pp. 208–229, 2013.

[19] W. Wang, J. Shen, and H. Ling, "A deep network solution for attention and aesthetics aware photo cropping," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2018.

[20] S. Avidan and A. Shamir, "Seam carving for content-aware image resizing," in *ACM Trans. Graph.*, vol. 26, no. 3, 2007, p. 10.

[21] J. Sun and H. Ling, "Scale and object aware image retargeting for thumbnail browsing," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011.

[22] Y. Sugano, Y. Matsushita, and Y. Sato, "Calibration-free gaze sensing using saliency maps," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 2667–2674.

[23] A. Borji and L. Itti, "Defending yarbus: Eye movements reveal observers' task," *Journal of Vision*, vol. 14, no. 3, pp. 29–29, 2014.

[24] A. Karpathy, S. Miller, and L. Fei-Fei, "Object discovery in 3d scenes via shape analysis," in *Proc. IEEE Conf. Robot. Autom.*, 2013, pp. 2088–2095.

[25] S. Frintrop, G. M. García, and A. B. Cremers, "A cognitive approach for object discovery," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 2329–2334.

[26] G. Li and Y. Yu, "Visual saliency based on multiscale deep features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 5455–5463.

[27] L. Wang, H. Lu, X. Ruan, and M.-H. Yang, "Deep networks for saliency detection via local estimation and global search," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3183–3192.

[28] R. Zhao, W. Ouyang, H. Li, and X. Wang, "Saliency detection by multi-context deep learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1265–1274.

[29] T. Liu, J. Sun, N.-N. Zheng, X. Tang, and H.-Y. Shum, "Learning to detect a salient object," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2007, pp. 1–8.

[30] 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.

[31] M.-M. Cheng, G.-X. Zhang, N. J. Mitra, X. Huang, and S.-M. Hu, "Global contrast based salient region detection," *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011.

[32] 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.* IEEE, 2012, pp. 733–740.

[33] J. Wang, H. Jiang, Z. Yuan, M.-M. Cheng, X. Hu, and N. Zheng, "Salient object detection: A discriminative regional feature integration approach," *Int. J. Comput. Vis.*, vol. 123, no. 2, pp. 251–268, 2017.

[34] Q. Yan, L. Xu, J. Shi, and J. Jia, "Hierarchical saliency detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 1155–1162.

[35] W. Zhu, S. Liang, Y. Wei, and J. Sun, "Saliency optimization from robust background detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 2814–2821.

[36] J. Zhang, S. Sclaroff, Z. Lin, X. Shen, B. Price, and R. Mech, "Unconstrained salient object detection via proposal subset optimization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 5733–5742.

[37] N. Liu and J. Han, "DHSNet: Deep hierarchical saliency network for salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 678–686.

[38] Q. Hou, M.-M. Cheng, X. Hu, A. Borji, Z. Tu, and P. Torr, "Deeply supervised salient object detection with short connections," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 3203–3212.

[39] 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.

[40] Y. Liu, Q. Zhang, D. Zhang, and J. Han, "Employing deep part-object relationships for salient object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 1232–1241.

[41] Q. Qi, S. Zhao, J. Shen, and K.-M. Lam, "Multi-scale capsule attention-based salient object detection with multi-crossed layer connections," in *IEEE International Conference on Multimedia and Expo*, 2019, pp. 1762–1767.

[42] A. Borji and L. Itti, "State-of-the-art in visual attention modeling," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 185–207, 2013.

[43] A. Borji, M.-M. Cheng, H. Jiang, and J. Li, "Salient object detection: A benchmark," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 5706–5722, 2015.

[44] T. V. Nguyen, Q. Zhao, and S. Yan, "Attentive systems: A survey," *Int. J. Comput. Vis.*, vol. 126, no. 1, pp. 86–110, 2018.

[45] D. Zhang, H. Fu, J. Han, A. Borji, and X. Li, "A review of co-saliency detection algorithms: fundamentals, applications, and challenges," *ACM Trans. Intell. Syst. Technol.*, vol. 9, no. 4, p. 38, 2018.

[46] R. Cong, J. Lei, H. Fu, M.-M. Cheng, W. Lin, and Q. Huang, "Review of visual saliency detection with comprehensive information," *IEEE Trans. Circuits Syst. Video Technol.*, 2018.

[47] J. Han, D. Zhang, G. Cheng, N. Liu, and D. Xu, "Advanced deep-learning techniques for salient and category-specific object detection: a survey," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 84–100, 2018.

[48] A. Borji, "Saliency prediction in the deep learning era: Successes and limitations," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2019.

[49] A. Borji, M.-M. Cheng, Q. Hou, H. Jiang, and J. Li, "Salient object detection: A survey," *Computational Visual Media*, pp. 1–34, 2019.

[50] A. M. Treisman and G. Gelade, "A feature-integration theory of attention," *Cognitive psychology*, vol. 12, no. 1, pp. 97–136, 1980.

[51] C. Koch and S. Ullman, "Shifts in selective visual attention: Towards the underlying neural circuitry," *Human neurobiology*, vol. 4, no. 4, p. 219, 1985.

[52] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 11, pp. 1254–1259, 1998.

[53] H. Jiang, J. Wang, Z. Yuan, Y. Wu, N. Zheng, and S. Li, "Salient object detection: A discriminative regional feature integration approach," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 2083–2090.

[54] H. Peng, B. Li, H. Ling, W. Hu, W. Xiong, and S. J. Maybank, "Salient object detection via structured matrix decomposition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 818–832, 2017.

[55] R. Cong, J. Lei, H. Fu, Q. Huang, X. Cao, and C. Hou, "Co-saliency detection for RGBD images based on multi-constraint feature matching and cross label propagation," *IEEE Trans. Image Process.*, vol. 27, no. 2, pp. 568–579, 2018.

[56] Y. Wei, F. Wen, W. Zhu, and J. Sun, "Geodesic saliency using background priors," *Proc. Eur. Conf. Comput. Vis.*, pp. 29–42, 2012.

[57] J. Shi, Q. Yan, L. Xu, and J. Jia, "Hierarchical image saliency detection on extended cssd," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 4, pp. 717–729, 2015.

[58] C. Yang, L. Zhang, H. Lu, X. Ruan, and M. H. Yang, "Saliency detection via graph-based manifold ranking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 3166–3173.

[59] W. Wang, J. Shen, L. Shao, and F. Porikli, "Correspondence driven saliency transfer," *IEEE Trans. Image Process.*, vol. 25, no. 11, pp. 5025–5034, 2016.

[60] F. Guo, W. Wang, J. Shen, L. Shao, J. Yang, D. Tao, and Y. Y. Tang, "Video saliency detection using object proposals," *IEEE Trans. Cybernetics*, 2017.

[61] G. E. Hinton, A. Krizhevsky, and S. D. Wang, "Transforming auto-encoders," in *Proc. Int. Conf. Artificial Neural Netw.*, 2011, pp. 44–51.

[62] X. Qin, Z. Zhang, C. Huang, C. Gao, M. Dehghan, and M. Jagersand, "Basnet: Boundary-aware salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 7479–7489.

[63] G. Li, Y. Xie, L. Lin, and Y. Yu, "Instance-level salient object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 247–256.

[64] S. He, R. W. Lau, W. Liu, Z. Huang, and Q. Yang, "Supercnn: A superpixelwise convolutional neural network for salient object detection," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 330–344, 2015.

[65] M.-M. Cheng, N. J. Mitra, X. Huang, P. H. S. Torr, and S.-M. Hu, "Global contrast based salient region detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 569–582, 2015.

[66] Y. Li, X. Hou, C. Koch, J. M. Rehg, and A. L. Yuille, "The secrets of salient object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 280–287.

[67] G. Lee, Y.-W. Tai, and J. Kim, "Deep saliency with encoded low level distance map and high level features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 660–668.

[68] G. Li and Y. Yu, "Deep contrast learning for salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 478–487.

[69] J. Kuen, Z. Wang, and G. Wang, "Recurrent attentional networks for saliency detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3668–3677.

[70] R. Ju, Y. Liu, T. Ren, L. Ge, and G. Wu, "Depth-aware salient object detection using anisotropic center-surround difference," *Signal Processing: Image Communication*, vol. 38, pp. 115–126, 2015.

[71] 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.*, 2014, pp. 92–109.

[72] S. S. Kruthiventi, V. Gudisa, J. H. Dholakiya, and R. Venkatesh Babu, "Saliency unified: A deep architecture for simultaneous eye fixation prediction and salient object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 5781–5790.

[73] M. Jiang, S. Huang, J. Duan, and Q. Zhao, "SALICON: Saliency in context," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1072–1080.

[74] J. Zhang, S. Ma, M. Sameki, S. Sclaroff, M. Betke, Z. Lin, X. Shen, B. Price, and R. Mech, "Salient object subitizing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 4045–4054.

[75] J. Kim and V. Pavlovic, "A shape-based approach for salient object detection using deep learning," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 455–470.

[76] Y. Tang and X. Wu, "Saliency detection via combining region-level and pixel-level predictions with cnns," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 809–825.

[77] L. Wang, L. Wang, H. Lu, P. Zhang, and X. Ruan, "Saliency detection with recurrent fully convolutional networks," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 825–841.

[78] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, 2010.

[79] L. Wang, H. Lu, Y. Wang, M. Feng, D. Wang, B. Yin, and X. Ruan, "Learning to detect salient objects with image-level supervision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017.

[80] 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.

[81] P. Hu, B. Shuai, J. Liu, and G. Wang, "Deep level sets for salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 540–549.

[82] Z. Luo, A. Mishra, A. Achkar, J. Eichel, S. Li, and P.-M. Jodoin, "Non-local deep features for salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6593–6601.

[83] S. He, J. Jiao, X. Zhang, G. Han, and R. W. Lau, "Delving into salient object subitizing and detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 1059–1067.

[84] P. Zhang, D. Wang, H. Lu, H. Wang, and X. Ruan, "Amulet: Aggregating multi-level convolutional features for salient object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 202–211.

[85] X. Chen, A. Zheng, J. Li, and F. Lu, "Look, perceive and segment: Finding the salient objects in images via two-stream fixation-semantic CNNs," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 1050–1058.

[86] D. Zhang, J. Han, and Y. Zhang, "Supervision by fusion: Towards unsupervised learning of deep salient object detector," in *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 1, no. 2, 2017, p. 3.

[87] T. Wang, A. Borji, L. Zhang, P. Zhang, and H. Lu, "A stagewise refinement model for detecting salient objects in images," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 4039–4048.

[88] P. Zhang, D. Wang, H. Lu, H. Wang, and B. Yin, "Learning uncertain convolutional features for accurate saliency detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 212–221.

[89] X. Hu, L. Zhu, J. Qin, C.-W. Fu, and P.-A. Heng, "Recurrently aggregating deep features for salient object detection." in *AAAI Conference on Artificial Intelligence*, 2018.

[90] G. Li, Y. Xie, and L. Lin, "Weakly supervised salient object detection using image labels," in *AAAI Conference on Artificial Intelligence*, 2018.

[91] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.

[92] C. Cao, Y. Hunag, Z. Wang, L. Wang, N. Xu, and T. Tan, "Lateral inhibition-inspired convolutional neural network for visual attention and saliency detection," in *AAAI Conference on Artificial Intelligence*, 2018.

[93] L. Zhang, J. Dai, H. Lu, Y. He, and G. Wang, "A bi-directional message passing model for salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1741–1750.

[94] J. Zhang, T. Zhang, Y. Dai, M. Harandi, and R. Hartley, "Deep unsupervised saliency detection: A multiple noisy labeling perspective," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 9029–9038.

[95] T. Wang, L. Zhang, S. Wang, H. Lu, G. Yang, X. Ruan, and A. Borji, "Detect globally, refine locally: A novel approach to saliency detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3127–3135.

[96] X. Zhang, T. Wang, J. Qi, H. Lu, and G. Wang, "Progressive attention guided recurrent network for salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 714–722.

[97] M. Amirul Islam, M. Kalash, and N. D. B. Bruce, "Revisiting salient object detection: Simultaneous detection, ranking, and subitizing of multiple salient objects," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018.

[98] W. Wang, J. Shen, X. Dong, and A. Borji, "Salient object detection driven by fixation prediction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1171–1720.

[99] X. Li, F. Yang, H. Cheng, W. Liu, and D. Shen, "Contour knowledge transfer for salient object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 370–385.

[100] S. Chen, X. Tan, B. Wang, and X. Hu, "Reverse attention for salient object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 236–252.

[101] B. Li, Z. Sun, and Y. Guo, "Supervae: Superpixelwise variational autoencoder for salient object detection," in *AAAI Conference on Artificial Intelligence*, 2019, pp. 8569–8576.

[102] Y. Zhuge, Y. Zeng, and H. Lu, "Deep embedding features for salient object detection," in *AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 9340–9347.

[103] M. Feng, H. Lu, and E. Ding, "Attentive feedback network for boundary-aware salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1623–1632.

[104] L. Zhang, J. Zhang, Z. Lin, H. Lu, and Y. He, "Capsal: Leveraging captioning to boost semantics for salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 6024–6033.

[105] Z. Wu, L. Su, and Q. Huang, "Cascaded partial decoder for fast and accurate salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3907–3916.

[106] R. Wu, M. Feng, W. Guan, D. Wang, H. Lu, and E. Ding, "A mutual learning method for salient object detection with intertwined multi-supervision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 8150–8159.

[107] Y. Zeng, Y. Zhuge, H. Lu, L. Zhang, M. Qian, and Y. Yu, "Multi-source weak supervision for saliency detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 6074–6083.

[108] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Advances Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[109] W. Wang, S. Zhao, J. Shen, S. C. Hoi, and A. Borji, "Salient object detection with pyramid attention and salient edges," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1448–1457.

[110] W. Wang, J. Shen, M.-M. Cheng, and L. Shao, "An iterative and cooperative top-down and bottom-up inference network for salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 5968–5977.

[111] J.-J. Liu, Q. Hou, M.-M. Cheng, J. Feng, and J. Jiang, "A simple pooling-based design for real-time salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3917–3926.

[112] J. Su, J. Li, Y. Zhang, C. Xia, and Y. Tian, "Selectivity or invariance: Boundary-aware salient object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 3799–3808.

[113] J.-X. Zhao, J.-J. Liu, D.-P. Fan, Y. Cao, J. Yang, and M.-M. Cheng, "Egnet: Edge guidance network for salient object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 8779–8788.

[114] Y. Zeng, P. Zhang, J. Zhang, Z. Lin, and H. Lu, "Towards high-resolution salient object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 7234–7243.

[115] Y. Xu, D. Xu, X. Hong, W. Ouyang, R. Ji, M. Xu, and G. Zhao, "Structured modeling of joint deep feature and prediction refinement for salient object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 3789–3798.

[116] Z. Wu, L. Su, and Q. Huang, "Stacked cross refinement network for edge-aware salient object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 7264–7273.

[117] Y. Zeng, Y. Zhuge, H. Lu, and L. Zhang, "Joint learning of saliency detection and weakly supervised semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 7223–7233.

[118] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3431–3440.

[119] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Representations*, 2015.

[120] 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.

[121] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu, "Spatial transformer networks," in *Proc. Advances Neural Inf. Process. Syst.*, 2015, pp. 2017–2025.

[122] 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, 2018.

[123] S. S. Kruthiventi, K. Ayush, and R. V. Babu, "Deepfix: A fully convolutional neural network for predicting human eye fixations," *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4446–4456, 2017.

[124] S. Xie and Z. Tu, "Holistically-nested edge detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1395–1403.

[125] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2015, pp. 234–241.

[126] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. Advances Neural Inf. Process. Syst.*, 2015, pp. 802–810.

[127] J. Yang, B. Price, S. Cohen, H. Lee, and M.-H. Yang, "Object contour detection with a fully convolutional encoder-decoder network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 193–202.

[128] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[129] S. Sabour, N. Frosst, and G. E. Hinton, "Dynamic routing between capsules," in *Proc. Advances Neural Inf. Process. Syst.*, 2017, pp. 3856–3866.

[130] G. E. Hinton, S. Sabour, and N. Frosst, "Matrix capsules with em routing," in *Proc. Int. Conf. Learn. Representations*, 2018, pp. 1–15.

[131] J. L. Long, N. Zhang, and T. Darrell, "Do convnets learn correspondence?" in *Proc. Advances Neural Inf. Process. Syst.*, 2014, pp. 1601–1609.

[132] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2921–2929.

[133] J. Zhang, S. Sclaroff, Z. Lin, X. Shen, B. Price, and R. Mech, "Minimum barrier salient object detection at 80 fps," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1404–1412.

[134] J. Zhang and S. Sclaroff, "Exploiting surroundedness for saliency detection: a boolean map approach," *IEEE Trans. Pattern Anal. Mach. Intell.*, no. 5, pp. 889–902, 2016.

[135] B. Jiang, L. Zhang, H. Lu, C. Yang, and M.-H. Yang, "Saliency detection via absorbing markov chain," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 1665–1672.

[136] X. Li, H. Lu, L. Zhang, X. Ruan, and M.-H. Yang, "Saliency detection via dense and sparse reconstruction," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 2976–2983.

[137] P. Arbeláez, J. Pont-Tuset, J. T. Barron, F. Marques, and J. Malik, "Multiscale combinatorial grouping," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 328–335.

[138] R. Caruana, "Multitask learning," *Machine learning*, vol. 28, no. 1, pp. 41–75, 1997.

[139] E. L. Kaufman, M. W. Lord, T. W. Reese, and J. Volkmann, "The discrimination of visual number," *The American Journal of Psychology*, vol. 62, no. 4, pp. 498–525, 1949.

[140] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Object detectors emerge in deep scene cnns," in *Proc. Int. Conf. Learn. Representations*, 2015.

[141] S. Alpert, M. Galun, R. Basri, and A. Brandt, "Image segmentation by probabilistic bottom-up aggregation and cue integration," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2007, pp. 1–8.

[142] V. Movahedi and J. H. Elder, "Design and perceptual validation of performance measures for salient object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. - Workshops*, 2010.

[143] C. Xia, J. Li, X. Chen, A. Zheng, and Y. Zhang, "What is and what is not a salient object? learning salient object detector by ensembling linear exemplar regressors," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4321–4329.

[144] D.-P. Fan, M.-M. Cheng, J.-J. Liu, S.-H. Gao, Q. Hou, and A. Borji, "Salient objects in clutter: Bringing salient object detection to the foreground," in *The Proc. Eur. Conf. Comput. Vis.*, 2018.

[145] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 898–916, 2010.

[146] R. Mottaghi, X. Chen, X. Liu, N.-G. Cho, S.-W. Lee, S. Fidler, R. Urtasun, and A. Yuille, "The role of context for object detection and semantic segmentation in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 891–898.

[147] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2, 2001, pp. 416–423.

[148] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba, "Sun database: Large-scale scene recognition from abbey to zoo," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 3485–3492.

[149] R. Margolin, L. Zelnik-Manor, and A. Tal, "How to evaluate foreground maps?" in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 248–255.

[150] 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.

[151] D.-P. Fan, M.-M. Cheng, J.-J. Liu, S.-H. Gao, Q. Hou, and A. Borji, "Enhanced-alignment measure for binary foreground map evaluation," in *International Joint Conferences on Artificial Intelligence*, 2018.

[152] Z. Wang and B. Li, "A two-stage approach to saliency detection in images," in *Proc. IEEE Conf. Acoust. Speech Signal Process.*, 2008, pp. 965–968.

[153] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung, "A benchmark dataset and evaluation

methodology for video object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 724–732.
[154] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," in *Proc. Int. Conf. Learn. Representations*, 2014.
[155] C. Xie, J. Wang, Z. Zhang, Y. Zhou, L. Xie, and A. Yuille, "Adversarial examples for semantic segmentation and object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1369–1378.
[156] N. Papernot, P. McDaniel, and I. Goodfellow, "Transferability in machine learning: from phenomena to black-box attacks using adversarial samples," *arXiv preprint arXiv:1605.07277*, 2016.
[157] A. Torralba and A. A. Efros, "Unbiased look at dataset bias," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 1521–1528.
[158] H. Ding, X. Jiang, B. Shuai, A. Qun Liu, and G. Wang, "Context contrasted feature and gated multi-scale aggregation for scene segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 2393–2402.
[159] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "Learning a discriminative feature network for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018.
[160] F. Yu, D. Wang, E. Shelhamer, and T. Darrell, "Deep layer aggregation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018.
[161] M. Berman, A. Rannen Triki, and M. B. Blaschko, "The lovász-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4413–4421.
[162] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2261–2269.
[163] B. Zoph and Q. V. Le, "Neural architecture search with reinforcement learning," in *Proc. Int. Conf. Learn. Representations*, 2017.
[164] M. Jaderberg, A. Vedaldi, and A. Zisserman, "Speeding up convolutional neural networks with low rank expansions," in *Proceedings of the British Machine Vision Conference*, 2014.
[165] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding," in *Proc. Int. Conf. Learn. Representations*, 2016.
[166] E. Bengio, P.-L. Bacon, J. Pineau, and D. Precup, "Conditional computation in neural networks for faster models," in *Proc. Int. Conf. Learn. Representations*, 2016.
[167] A. Veit and S. Belongie, "Convolutional networks with adaptive inference graphs," in *Proc. Eur. Conf. Comput. Vis.*, 2018.
[168] S. Teerapittayanon, B. McDanel, and H. Kung, "Branchynet: Fast inference via early exiting from deep neural networks," in *Proc. Int. Conf. Pattern Recognit.*, 2016, pp. 2464–2469.
[169] A. Zlateski, R. Jaroensri, P. Sharma, and F. Durand, "On the importance of label quality for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018.
[170] Q. Hou, M.-M. Cheng, X. Hu, A. Borji, Z. Tu, and P. H. Torr, "Deeply supervised salient object detection with short connections," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2018.
[171] M. Jiang, J. Xu, and Q. Zhao, "Saliency in crowd," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 17–32.
[172] Q. Zheng, J. Jiao, Y. Cao, and R. W. Lau, "Task-driven webpage saliency," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 287–302.
[173] A. Palazzi, F. Solera, S. Calderara, S. Alletto, and R. Cucchiara, "Where should you attend while driving?" *arXiv preprint arXiv:1611.08215*, 2016.
[174] A. K. Mishra, Y. Aloimonos, L. F. Cheong, and A. Kassim, "Active visual segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 639–653, 2012.
[175] C. M. Masciocchi, S. Mihalas, D. Parkhurst, and E. Niebur, "Everyone knows what is interesting: Salient locations which should be fixated," *Journal of Vision*, vol. 9, no. 11, pp. 25–25, 2009.
[176] A. Borji, "What is a salient object? A dataset and a baseline model for salient object detection," *IEEE Trans. Image Process.*, vol. 24, no. 2, pp. 742–756, 2015.
[177] S. Gidaris and N. Komodakis, "Object detection via a multi-region and semantic segmentation-aware cnn model," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1134–1142.
[178] Z. Zhang, S. Qiao, C. Xie, W. Shen, B. Wang, and A. L. Yuille, "Single-shot object detection with enriched semantics," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 5813–5821.
[179] C. Bucilua, R. Caruana, and A. Niculescu-Mizil, "Model compression," in *Proceedings of SIGKDD international conference on Knowledge discovery and data mining*, 2006, pp. 535–541.
[180] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," in *Proc. Advances Neural Inf. Process. Syst. - workshops*, 2014.
[181] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio, "Fitnets: Hints for thin deep nets," *arXiv preprint arXiv:1412.6550*, 2014.
[182] G. Chen, W. Choi, X. Yu, T. Han, and M. Chandraker, "Learning efficient object detection models with knowledge distillation," in *Proc. Advances Neural Inf. Process. Syst.*, 2017, pp. 742–751.

Wenguan Wang received his Ph.D. degree from Beijing Institute of Technology in 2018. He is currently working with Prof. Luc Van Gool, as a postdoc scholar at ETH Zurich, Switzerland. From 2016 to 2018, he was a joint Ph.D. candidate in University of California, Los Angeles, directed by Prof. Song-Chun Zhu. From 2018 to 2019, he was a senior scientist at Inception Institute of Artificial Intelligence, UAE. His current research interests include computer vision, image processing and deep learning.

Qiuxia Lai received the B.E. and M.S. degrees in the School of Automation from Huazhong University of Science and Technology in 2013 and 2016, respectively. She is currently pursuing the Ph.D. degree with the Department of Computer Science and Engineering, The Chinese University of Hong Kong. Her research interests include image/video processing and deep learning.

Huazhu Fu (SM'18) received the Ph.D. degree in computer science from Tianjin University, China, in 2013. He was a Research Fellow with Nanyang Technological University, Singapore for two years. From 2015 to 2018, he was a Research Scientist with the Institute for Infocomm Research, Agency for Science, Technology and Research, Singapore. He is currently a Senior Scientist with Inception Institute of Artificial Intelligence, Abu Dhabi, United Arab Emirates. His research interests include computer vision, image processing, and medical image analysis. He is an Associate Editor of IEEE Access and BMC Medical Imaging.

Jianbing Shen (M'11-SM'12) is a Professor with the School of Computer Science, Beijing Institute of Technology. He has published about 100 journal and conference papers such as *IEEE Trans. on PAMI*, *CVPR*, and *ICCV*. He has obtained many honors including the Fok Ying Tung Education Foundation from Ministry of Education, the Program for Beijing Excellent Youth Talents from Beijing Municipal Education Commission, and the Program for New Century Excellent Talents from Ministry of Education. His research interests include computer vision and deep learning. He is an Associate Editor of IEEE TNNLS, IEEE TIP and Neurocomputing.

Haibin Ling received the PhD degree from University of Maryland in 2006. From 2000 to 2001, he was an assistant researcher at Microsoft Research Asia. From 2006 to 2007, he worked as a postdoc at University of California Los Angeles. After that, he joined Siemens Corporate Research as a research scientist. Since 2008, he has been with Temple University where he is now an Associate Professor. He received the Best Student Paper Award at the ACM UIST in 2003, and the NSF CAREER Award in 2014. He is an Associate Editor of IEEE TPAMI, PR, and CVIU, and served as Area Chairs for CVPR 2014, 2016 and 2019.

Ruigang Yang is currently a full professor of Computer Science at the University of Kentucky. His research interests span over computer vision and computer graphics, in particular in 3D reconstruction and 3D data analysis. He has received a number of awards, including the US National Science Foundation Faculty Early Career Development (CAREER) Program Award in 2004, and the best Demonstration Award at CVPR 2007. He is currently an associate editor of IEEE TPAMI and a senior member of IEEE.