

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) has attracted an increasing amount of research attention over the years. Recent advances in SOD are predominantly led by deep learning-based solutions (named deep SOD). To enable in-depth understanding of deep SOD, in this paper, we provide a comprehensive survey covering various aspects, ranging from algorithm taxonomy to unsolved 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 and analyze existing SOD datasets and evaluation metrics. Then, we benchmark a large group of representative SOD models, and provide detailed analyses of the comparison results. Moreover, we study the performance of SOD algorithms under different attribute settings, which has not been thoroughly explored previously, by constructing a novel SOD dataset with rich attribute annotations covering various salient object types, challenging factors, and scene categories. We further analyze, for the first time in the field, the robustness of SOD models to random input perturbations and adversarial attacks. We also look into the generalization and difficulty of existing SOD datasets. Finally, we discuss several open issues of SOD and outline future research directions. All the saliency prediction maps, our constructed dataset with annotations, and codes for evaluation are publicly available at <https://github.com/wenguanwang/SODsurvey>.

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

1 INTRODUCTION

SLIENT object detection (SOD) aims at highlighting visually salient object regions in images. Here, ‘visually salient’ describes the property of an object or a region to attract human observers’ attention. SOD 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], unsupervised video object segmentation [8], [9], semantic segmentation [10]–[12], person re-identification [13], [14], and video summarization [15], [16]. In computer graphics, SOD also plays an essential role in various tasks, including non-photorealistic rendering [17], [18], image cropping [19], [20], image retargeting [21], etc. Several applications in robotics, such as human-robot interaction [22], [23], and object discovery [24], [25], also benefit from SOD for better scene/object understanding.

Though inspired by eye fixation prediction (FP) [26], which originated from cognitive and psychology research communities to investigate the human attention mechanism

by predicting eye fixation positions in visual scenes, SOD differs in that it aims to detect the whole attentive object regions. Since the renaissance of deep learning techniques, significant improvement for SOD has been achieved in recent years, thanks to the powerful representation learning methods. Since the first introduction in 2015 [27]–[29], deep learning-based SOD (or *deep SOD*) algorithms have quickly shown superior performance over traditional solutions, and have continued to improve the state-of-the-art.

This paper provides a comprehensive and in-depth survey of SOD in the deep learning era. In addition to taxonomically reviewing existing deep SOD methods, it provides in-depth analyses of representative datasets and evaluation metrics, and investigates crucial but largely under-explored issues, such as the robustness and transferability of deep SOD models, their strengths and weaknesses under certain scenarios (*i.e.*, scene/salient object categories, challenging factors), as well as the generalizability and difficulty of SOD datasets. The saliency maps used for benchmarking, our constructed dataset, and evaluation codes are available at <https://github.com/wenguanwang/SODsurvey>.

1.1 History and Scope

Humans are able to quickly allocate attention to important regions in visual scenes. Understanding 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 FP and SOD. FP originated from cognitive and psychology communities [26], [51], [52], and targets at predicting where people look in images.

- 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: hblng@temple.edu)
- R. Yang is with the University of Kentucky, Lexington, KY 40507. (Email: ryang@cs.uky.edu)
- Corresponding author: Jianbing Shen

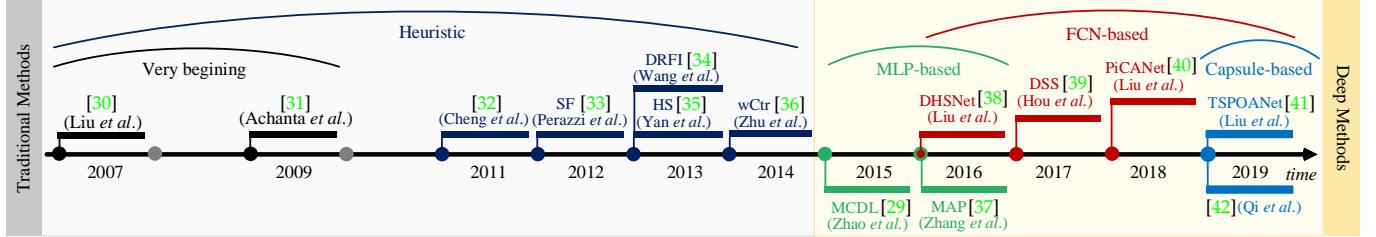


Fig. 1. A brief chronology of SOD. The very first SOD models date back to the work of Liu *et al.* [30] and Achanta *et al.* [31]. The first incorporation of deep learning techniques into SOD models was in 2015. Listed methods are milestones, which are typically highly cited. See §1.1 for more details.

TABLE 1

Summary of previous reviews. For each work, the publication information and coverage are provided. See §1.2 for more detailed descriptions.

Title	Year	Venue	Description
State-of-the-Art in Visual Attention Modeling [43]	2013	TPAMI	This paper reviews visual attention (<i>i.e.</i> fixation prediction) models before 2013.
Salient Object Detection: A Benchmark [44]	2015	TIP	This paper benchmarks 29 heuristic SOD models and 10 FP methods over 7 datasets.
Attentive Systems: A Survey [45]	2017	IJCV	This paper reviews applications that utilize visual saliency cues.
A Review of Co-Saliency Detection Algorithms: Fundamentals, Applications, and Challenges [46]	2018	TIST	This paper reviews the fundamentals, challenges, and applications of co-saliency detection.
Review of Visual Saliency Detection with Comprehensive Information [47]	2018	TCSVT	This paper reviews RGB-D SOD, co-saliency detection and video SOD.
Advanced Deep-Learning Techniques for Salient and Category-Specific Object Detection: A Survey [48]	2018	SPM	This paper reviews several sub-directions of object detection, namely objectness detection, SOD and category-specific object detection.
Saliency Prediction in the Deep Learning Era: Successes and Limitations [49]	2019	TPAMI	This paper reviews image and video fixation prediction models and analyzes specific questions.
Salient Object Detection: A Survey [50]	2019	CVM	This paper reviews 65 heuristic and 21 deep SOD models up to 2017 and discusses closely related areas like object detection, fixation prediction, segmentation, etc.

The history of SOD is relatively short and can be traced back to [30] and [31]. The rise of SOD has been driven by a wide range of object-level computer vision applications. Instead of FP models only predicting sparse eye fixation locations, SOD models aim to detect the whole entities of the visually attractive objects with precise boundaries. Most traditional, **non-deep SOD models** [36], [53] rely on low-level features and certain heuristics (*e.g.*, *color contrast* [32], *background prior* [54]). To obtain uniformly highlighted salient objects and clear object boundaries, an over-segmentation process that generates regions [55], super-pixels [56], [57], or object proposals [58] is often integrated into these models. Please see [44] for a more comprehensive overview.

With the compelling success of deep learning technologies in computer vision, more and more **deep SOD methods** have begun springing up since 2015. Earlier deep SOD models utilized multi-layer perceptron (MLP) classifiers to predict the saliency score of deep features extracted from each image processing unit [27]–[29]. Later, a more effective and efficient form, *i.e.*, fully convolutional network (FCN)-based model, became the mainstream SOD architecture. Some recent methods [41], [42] also introduced Capsule [59] into SOD to comprehensively address object property modeling. A brief chronology of SOD is shown in Fig. 1.

Scope of the survey. Despite its short history, research in deep SOD has produced hundreds of papers, making it impractical (and fortunately unnecessary) to review all of them. Instead, we comprehensively 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. Due to limitations on space and our knowledge, we apologize to those authors whose works are not included in this paper. It is worth not-

ing that we restrict this survey to *single-image SOD* methods, and leave RGB-D SOD, co-saliency detection, video SOD, *etc.*, as separate topics.

1.2 Related Previous Reviews and Surveys

Table 1 lists existing surveys that are related to ours. Among them, Borji *et al.* [44] reviewed SOD methods preceding 2015, thus do not refer to recent deep learning-based solutions. Zhang *et al.* [46] reviewed methods for co-saliency detection, *i.e.*, detecting common salient objects from multiple relevant images. Cong *et al.* [47] reviewed several extended SOD tasks including RGB-D SOD, co-saliency detection and video SOD. Han *et al.* [48] looked into several sub-directions of object detection, and outlined recent progress in objectness detection, SOD, and category-specific object detection. Borji *et al.* summarized both heuristic [43] and deep models [49] for FP. Nguyen *et al.* [45] focused on categorizing the applications of visual saliency (including both SOD and FP) in different areas. Finally, a more recently published survey [50] covers both traditional non-deep SOD methods and deep ones until 2017, and discusses their relation to several other closely-related research areas, such as special-purpose object detection and segmentation.

Different from previous SOD surveys, which focus on earlier non-deep learning SOD methods [44], other related fields [43], [47]–[49], practical applications [45] or a limited number of deep SOD models [50], this work systematically and comprehensively reviews recent advances in the field. It features in-depth analyses and discussions on various aspects, many of which, to the best of our knowledge, have never been explored in this field. In particular, we comprehensively summarize and discuss existing deep SOD methods under several proposed taxonomies (§2); review datasets (§3) and evaluation metrics (§4) with their pros

TABLE 2
Taxonomies and representative publications of deep SOD methods. See §2 for more detailed descriptions.

Category		Publications
Network Architectures (§2.1)	Multi-layer perceptron (MLP)-based	1) Super-pixel/patch-based [29], [60], [27], [61] 2) Object proposal based [28], [37], [62]
	Fully convolutional network (FCN)-based	1) Single-stream [63], [64], [65], [66], [67], [68], [69] 2) Multi-stream [70], [71], [72], [73], [74] 3) Side-fusion [39], [75], [76], [77], [78], [79], [80], [81], [82] 4) Bottom-up/top-down [38], [83], [84], [85], [86], [87], [40], [88], [89], [90], [91], [92], [93], [94], [95] 5) Branched [96], [97], [98], [99], [100], [101], [102], [103]
	Hybrid network-based	[104], [105]
	Capsule-based	[41], [42]
	Fully-supervised	All others
Level of Supervision (§2.2)	Un-/Weakly-supervised	1) Category-level [97], [68], [69], [81] 2) Pseudo pixel-level [83], [98], [67], [99]
	Single-task learning (STL)	All others
Learning Paradigm (§2.3)	Mingle-task learning (MTL)	1) Salient object subitizing [37], [77], [79] 2) Fixation prediction [96], [87] 3) Image classification [97], [98] 4) Semantic segmentation [63], [103] 5) Contour/edge detection [75], [99], [89], [91], [92], [93], [101], [82], [102] 6) Image captioning [100]
		All others
Object-/Instance-Level (§2.4)	Object-level	[37]
	Instance-level	[70]

and cons; provide a deeper understanding of SOD models through an attribute-based evaluation (§5.3); discuss the influence of input perturbation (§5.4); analyze the robustness of deep SOD models to adversarial attacks (§5.5); study the generalization and difficulty of existing SOD datasets (§5.6); and offer insight into essential open issues, challenges, and future directions (§6). We expect our survey to provide novel insight and inspiration that will facilitate the understanding of deep SOD, and foster research on the open issues raised.

1.3 Our Contributions

Our contributions in this paper are summarized as follows:

- 1) **A 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 gain a deeper understanding of the key features of deep SOD models.
- 2) **An attribute-based performance evaluation of SOD models.** We compile a hybrid dataset and provide annotated attributes for object categories, scene categories, and challenging factors. By evaluating several representative SOD models on it, we uncover the strengths and weaknesses of deep and non-deep approaches, opening up promising directions for future efforts.
- 3) **An analysis of the robustness of SOD models against general input perturbations.** To study the robustness of SOD models, we investigate the effects of various perturbations on the final performance of deep and non-deep SOD models. Some results are somewhat unexpected.
- 4) **The first known adversarial attack analysis for SOD models.** We further examine the robustness of SOD models against intentionally designed perturbations, i.e., adversarial attacks. The specially designed attacks and evaluations can serve as baselines for further studying the robustness and transferability of deep SOD models.
- 5) **Cross-dataset generalization study.** To analyze the generalization and difficulty of existing SOD datasets in

depth, we conduct a cross-dataset generalization study that quantitatively reveals the dataset bias.

- 6) **Overview of open issues and future directions.** We thoroughly look over several essential issues (i.e., model design, dataset collection, etc.), shedding light on potential directions for future research.

These contributions together comprise 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, each accompanied with one or two most representative models. §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 further discussions and presents open issues and future research directions of the field. Finally, §7 concludes the paper.

2 DEEP LEARNING BASED SOD MODELS

Before reviewing recent deep SOD models in details, we first provide 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 static images $\mathcal{I} = \{\mathbf{I}_n \in \mathbb{R}^{W \times H \times 3}\}_n$ and corresponding binary SOD ground-truth masks $\mathcal{G} = \{\mathbf{G}_n \in \{0, 1\}^{W \times H}\}_n$, the goal of learning is to find $f \in \mathcal{F}$ that minimizes the prediction error, i.e., $\sum_n \ell(S_n, G_n)$, where ℓ is a certain distance measure (e.g., defined in §4), $S_n = f(\mathbf{I}_n)$, 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 later 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 annotations, which will be discussed in §3.

In Table 2, we categorize recent deep SOD models according to four taxonomies, considering *network architecture*

TABLE 3

Summary of essential characteristics for popular SOD methods. Here, '#Training' is the number of training images, and 'CRF' denotes whether the predictions are post-processed by conditional random field [106]. See §2 for more detailed descriptions.

	Methods	Publ.	Architecture	Backbone	Level of Supervision	Learning Paradigm	Obj.-/Inst.-Level SOD	Training Dataset	#Training	CRF
2015	SuperCNN [61]	IJCV	MLP+super-pixel	-	Fully-Sup.	STL	Object	ECSSD [55]	800	
	MCDL [29]	CVPR	MLP+super-pixel	GoogleNet	Fully-Sup.	STL	Object	MSRA10K [107]	8,000	
	LEGS [28]	CVPR	MLP+segment	-	Fully-Sup.	STL	Object	MSRA-B [30]+PASCAL-S [108]	3,000+340	
	MDF [27]	CVPR	MLP+segment	-	Fully-Sup.	STL	Object	MSRA-B [30]	2,500	
2016	ELD [60]	CVPR	MLP+super-pixel	VGGNet	Fully-Sup.	STL	Object	MSRA10K [107]	~9,000	
	DHSNet [38]	CVPR	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA10K [107]+DUT-OMRON [56]	6,000+3,500	
	DCL [104]	CVPR	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA-B [30]	2,500	✓
	RACDNN [64]	CVPR	FCN	VGGNet	Fully-Sup.	STL	Object	DUT-OMRON [56]+NU2000 [109]+RGBD [110]	10,565	
	SU [96]	CVPR	FCN	VGGNet	Fully-Sup.	MTL	Object	MSRA10K [107]+SALICON [111]	10,000+15,000	✓
	MAP [37]	CVPR	MLP+obj. prop.	VGGNet	Fully-Sup.	MTL	Instance	SOS [112]	~5,500	
	SSD [62]	ECCV	MLP+obj. prop.	AlexNet	Fully-Sup.	STL	Object	MSRA-B [30]	2,500	
	CRPSD [105]	ECCV	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA10K [107]	10,000	
2017	RFCN [63]	ECCV	FCN	VGGNet	Fully-Sup.	MTL	Object	PASCAL VOC 2010 [113]+MSRA10K [107]	10,103+10,000	
	MSRNet [70]	CVPR	FCN	VGGNet	Fully-Sup.	STL	Instance	MSRA-B [30]+HKU-IS [27] (+ILSO [70])	2,500+2,500 (+500)	✓
	DSS [39]	CVPR	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA-B [30]-HKU-IS [27]	2,500	✓
	WSS [97]	CVPR	FCN	VGGNet	Weakly-Sup.	MTL	Object	ImageNet [114]	456k	✓
	DLS [65]	CVPR	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA10K [107]	10,000	
	NLDF [75]	CVPR	FCN	VGGNet	Fully-Sup.	MTL	Object	MSRA-B [30]	2,500	
	DSOS [77]	ICCV	FCN	VGGNet	Fully-Sup.	MTL	Object	SOS [112]	6,900	
	Amulet [76]	ICCV	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA10K [107]	10,000	
	FSN [72]	ICCV	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA10K [107]	10,000	
	SBF [83]	ICCV	FCN	VGGNet	Un-Sup.	STL	Object	MSRA10K [107]	10,000	
	SRM [71]	ICCV	FCN	ResNet	Fully-Sup.	STL	Object	DUTS [97]	10,553	
	UCF [66]	ICCV	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA10K [107]	10,000	
	RADF [78]	AAAI	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA10K [107]	10,000	
	ASMO [98]	AAAI	FCN	ResNet101	Weakly-Sup.	MTL	Object	MS COCO [115]+MSRA-B [30]+HKU-IS [27]	82,783+2,500+2,500	✓
2018	LICNN [68]	AAAI	FCN	VGGNet	Weakly-Sup.	STL	Object	ImageNet [114]	456k	✓
	BDMF [84]	CVPR	FCN	VGGNet	Fully-Sup.	STL	Object	DUTS [97]	10,553	
	DUS [67]	CVPR	FCN	ResNet101	Un-Sup.	MTL	Object	MSRA-B [30]	2,500	
	DGRL [85]	CVPR	FCN	ResNet50	Fully-Sup.	STL	Object	DUTS [97]	10,553	
	PAGR [86]	CVPR	FCN	VGGNet19	Fully-Sup.	STL	Object	DUTS [97]	10,553	
	RSDNet [79]	CVPR	FCN	ResNet101	Fully-Sup.	MTL	Object	PASCAL-S [108]	425	
	ASNet [87]	CVPR	FCN	VGGNet	Fully-Sup.	MTL	Object	SALICON [111]+MSRA10K [107]+DUT-OMRON [56]	15,000+10,000+5,168	
	PiCANet [40]	CVPR	FCN	VGGNet/ResNet50	Fully-Sup.	STL	Object	DUTS [97]	10,553	✓
	C25-Net [99]	ECCV	FCN	VGGNet	Weakly-Sup.	MTL	Object	MSRA10K [107]+Web	10,000+20,000	
	RAS [88]	ECCV	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA-B [30]	2,500	
2019	SuperVAE [69]	AAAI	FCN	N/A	Un-Sup.	STL	Object	N/A	N/A	
	DEF [74]	AAAI	FCN	ResNet101	Fully-Sup.	STL	Object	DUTS [97]	10,553	
	AFNet [89]	CVPR	FCN	VGGNet16	Fully-Sup.	MTL	Object	DUTS [97]	10,553	
	BASNet [90]	CVPR	FCN	ResNet-34	Fully-Sup.	STL	Object	DUTS [97]	10,553	
	CapSal [100]	CVPR	FCN	ResNet101	Fully-Sup.	MTL	Object	COCO-CapSal [100]/DUTS [97]	5,265/10,553	
	CPD-R [80]	CVPR	FCN	ResNet50	Fully-Sup.	STL	Object	DUTS [97]	10,553	
	MLSNet [91]	CVPR	FCN	VGG16	Fully-Sup.	MTL	Object	DUTS [97]	10,553	
	†MWS [81]	CVPR	FCN	N/A	Weakly-Sup.	STL	Object	ImageNet DET [114]-MS COCO [115]+ImageNet [116]+DUTS [97]	456k+82,783+300,000+10,553	
	PAGE-Net [92]	CVPR	FCN	VGGNet16	Fully-Sup.	MTL	Object	MSRA10K [107]	10,000	✓
	PS [94]	CVPR	FCN	ResNet50	Fully-Sup.	STL	Object	MSRA10K [107]	10,000	✓
	PoolNet [93]	CVPR	FCN	ResNet50	Fully-Sup.	STL/MTL	Object	DUTS [97]	10,553	
	BANet [101]	ICCV	FCN	ResNet50	Fully-Sup.	MTL	Object	DUTS [97]	10,553	
	EGNet [82]	ICCV	FCN	VGGNet/ResNet	Fully-Sup.	MTL	Object	DUTS [97]	10,553	
	HRSOD [73]	ICCV	FCN	VGGNet	Fully-Sup.	STL	Object	DUTS [97]/HRSOD [73]+DUTS [97]	10,553/12,163	
	JDFPR [95]	ICCV	FCN	VGG	Fully-Sup.	STL	Object	MSRA-B [30]	2,500	✓
	SCRN [102]	ICCV	FCN	ResNet50	Fully-Sup.	MTL	Object	DUTS [97]	10,553	
	SSNet [103]	ICCV	FCN	Desenet169	Fully-Sup.	MTL	Object	PASCAL VOC 2012 [113]+DUTS [97]	1,464+10,553	
	TSPOANet [41]	ICCV	Capsule	FLNet	Fully-Sup.	STL	Object	DUTS [97]	10,553	

(§2.1), level of supervision (§2.2), learning paradigm (§2.3), and whether they works at an *object* or *instance* level (§2.4). In the following, each category is elaborated on and exemplified by one or two most representative models. Table 3 summarizes essential characteristics of recent SOD models.

2.1 Representative Network Architectures for SOD

Based on the primary network architectures adopted, we classify deep SOD models into four categories, namely *MLP-based* (§2.1.1), *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 leverage image subunits (*i.e.*, *super-pixels/patches* [29], [60], [61] and generic *object proposals* [27], [28], [37], [62]) as processing units. They feed deep features extracted from the subunits into an MLP-classifier for saliency score prediction (Fig. 2(a)).

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

example of regular decomposition, MCDL [29] uses two pathways to extract local and global context from two super-pixel-centered windows of different sizes. The global and local feature vectors are fed into an MLP for classifying background and saliency. In contrast, SuperCNN [61] constructs two hand-crafted input feature sequences for each irregular super-pixel, and use two separate CNN columns to produce saliency scores from the feature sequences, respectively. Regular image decomposition can accelerate the processing speed, thus most of the methods in this category are based on regular decompostion.

2) Object proposal-based methods leverage object proposals [27], [28] or bounding-boxes [37], [62] as basic processing units in order to better encode object information. For instance, MAP [37] 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. Note that this kind of methods typically produce coarse SOD results due to the lack of object boundary information.

Though MLP-based SOD methods greatly outperform

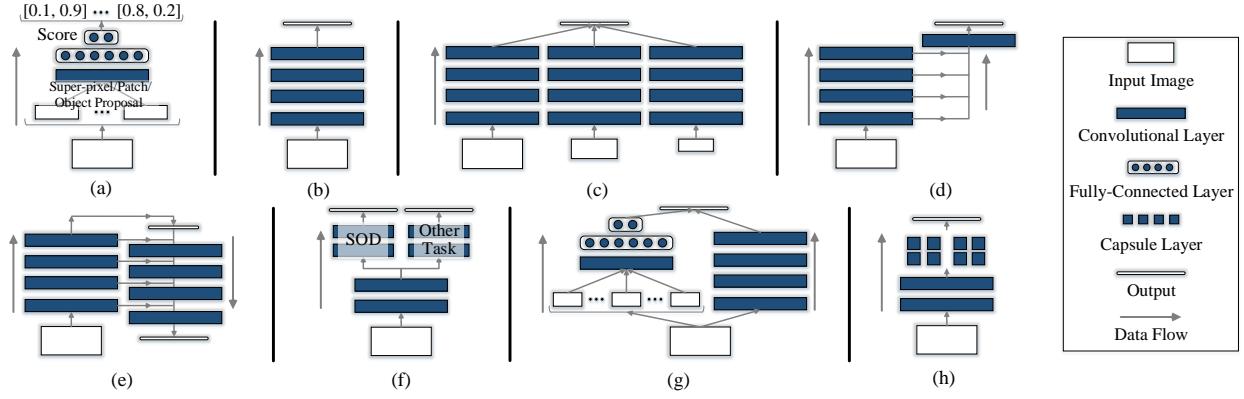


Fig. 2. Categorization of previous deep SOD models according to the adopted network architecture. (a) MLP-based methods. (b)-(f) FCN-based methods, mainly using (b) single-stream network, (c) multi-stream network, (d) side-out fusion network, (e) bottom-up/top-down network, and (f) branch network architectures.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

their non-deep counterparts, they cannot fully leverage essential spatial information and are quite time-consuming, as they need to process all visual subunits one-by-one.

2.1.2 Fully Convolutional Network (FCN)-Based Methods

To address the limitations of MLP-based methods, recent solutions adopt FCN architecture [117], leading to end-to-end spatial saliency representation learning and fast saliency prediction, within a single feed-forward process. FCN-based methods are now dominant in the field. Typical architectures can be further classified as: *single-stream*, *multi-stream*, *side-fusion*, *bottom-up/top-down*, and *branched networks*.

1) Single-stream network is the most standard architecture, having a stack of convolutional layers, interleaved with pooling and non-linear activation operations (see Fig. 2(b)). It takes a whole image as input, and directly outputs a pixel-wise probabilistic map highlighting salient objects. For example, UCF [66] makes use of an encoder-decoder network architecture for finer-resolution saliency prediction. It incorporates a reformulated dropout in the encoder to learn uncertain features, and a hybrid upsampling scheme in the decoder to avoid checkerboard artifacts.

2) Multi-stream network, as depicted in Fig. 2(c), typically consists of multiple network streams to explicitly learn multi-scale saliency features from multi-resolution inputs. Multi-stream outputs are fused to form a final prediction. DCL [104], as one of the earliest attempts towards this direction, contains two streams, which produce pixel- and region-level SOD estimations, respectively.

3) Side-fusion network fuses multi-layer responses of a backbone network together for SOD prediction, making use of the complementary information 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 [118]. As a well-known side-fusion network based SOD model, DSS [39] adds short connections from deeper side-outputs to shallower ones. In this way, higher-level features help lower side-outputs to better locate salient regions, and lower-level features can enrich deeper side-outputs with finer details.

4) Bottom-up/top-down network refines rough saliency maps in the feed-forward pass by gradually incorporating

spatial-detail-rich features from lower layers, and produces the finest saliency maps at the top-most layer (Fig. 2(e)), which resembles the U-Net [119] for semantic segmentation. This network architectures is first adopted by PiCANet [40], which hierarchically embeds global and local pixel-wise attention modules to selectively attend to informative context.

5) Branched network typically addresses multi-task learning for more robust saliency pattern modeling. They have a *single-input-multiple-output* structure, where bottom layers are shared to process a common input and top ones are specialized for different tasks (Fig. 2(f)). For example, C2S-Net [99] is constructed by adding a pre-trained contour detection model [120] to a main SOD branch. Then the two branches are alternately trained for the two tasks, *i.e.*, SOD and contour detection.

2.1.3 Hybrid Network-Based Methods

Some other models combine both MLP- and FCN-based subnets to produce edge-preserving results with multi-scale context (Fig. 2(g)). Combining pixel-level and region-level saliency cues is a promising strategy to yield improved performance, though it introduces extra computational costs. CRPSD [105] consolidates this idea. It combines pixel- and region-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 an existing SOD model [29] to image regions. Only the FCN and fusion layers are trainable.

2.1.4 Capsule-Based Methods

Recently, Hinton *et al.* [59] proposed a new family of neural networks, named *Capsules*. Capsules are made up of a group of neurons which accept and output vectors as opposed to scalar values of CNNs, allowing entity properties to be comprehensively modeled. Some researchers have thus been inspired to explore Capsules in SOD [41], [42] (Fig. 2(h)). For instance, TSPOANet [41] emphasizes part-object relations using a two-stream capsule network. The input features of capsules are extracted from a CNN, and transformed into low-level capsules. These are then assigned to high-level capsules, and finally recognized to be salient or background.

2.2 Level of Supervision

Based on the type of supervision, deep SOD models can be classified into either *fully-supervised* or *weakly-/unsupervised*.

2.2.1 Fully-Supervised Methods

Most deep SOD models are trained with large-scale pixel-level human annotations, which are time-consuming and expensive to acquire. Moreover, models trained on fine-labeled datasets tend to overfit and generalize poorly to real-life images [67]. Thus, training SOD with weaker annotations has become an increasingly popular research direction.

2.2.2 Weakly-/Unsupervised Methods

To get rid of laborious manual labeling, several weak supervision forms have been explored in SOD, including *image-level category labels* [68], [97], *object contours* [99], *image captions* [81] and *pseudo ground-truth* masks generated by non-learning SOD methods [67], [83], [98].

1) Category-level supervision. It has been shown that deep features trained with only image-level labels also provide information on object locations [121], [122], making them promising supervision signals for SOD training. WSS [97], as a typical example, first pre-trains a two-branch network, where one branch is used to predict image labels based on ImageNet [114], and the other estimates SOD maps. The estimated maps are refined by CRF and used to further fine-tune the SOD branch.

2) Pseudo pixel-level supervision. Though informative, image-level labels are weak. Some researchers therefore instead use traditional non-learning SOD methods [67], [83], [98], or contour information [99], to generate noisy yet finer-grained cues for training. For instance, SBF [83] fuses weak saliency maps from a set of prior heuristic SOD models [35], [123], [124] at intra- and inter-image levels, to generate supervision signals. C2S-Net [99] trains the SOD branch with the pixel-wise salient object masks generated from the outputs of the contour branch [125] using CEDN [120]. The contour and SOD branches alternatively update each other and progressively output finer SOD predictions.

2.3 Learning Paradigm

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

2.3.1 Single-Task Learning (STL) Based Methods

In machine learning, the standard practice is to learn one task at a time [126], *i.e.*, STL. Most deep SOD methods belong to this realm of learning, *i.e.*, they utilize supervision from a single knowledge domain (SOD or another related field such as image classification [68]) for training.

2.3.2 Multi-Task Learning (MTL) Based Methods

Inspired by the human learning process, where knowledge learned from related tasks can assist the learning of a new task, MTL [126] aims to improve the performance of multiple related tasks by learning them simultaneously. Benefiting from extra knowledge from related tasks, models can gain improved generalizability. An extra advantage lies

in the sharing of samples among tasks, which alleviates the lack of data for training heavily parameterized models. These are the core motivations of MTL based SOD models, and branched architectures (see §2.1.2) are usually adopted.

1) Salient object subitizing. The ability of humans to rapidly enumerate a small number of items is known as subitizing [112], [127]. Inspired by this, some works learn salient object subitizing and detection simultaneously [37], [77], [79]. RSDNet [79] represents the latest advance in this direction. It addresses detection, ranking and subitizing of salient objects in a unified framework.

2) Fixation prediction aims to predict human eye-fixation locations in visual scenes. Due to its close relation with SOD, learning shared knowledge from these two tasks can improve the performance of both. For example, ASNet [87] derives fixation information as a high-level understanding of the scene, from upper network layers. Then, fine-grained object-level saliency is progressively optimized under the guidance of the fixation in a top-down manner. **3) Image classification.** Image-level tags are valuable for SOD, as they provide the category information of dominant objects in the images which are very likely to be the salient regions [97]. Inspired by this, some SOD models learn image classification as an auxiliary task. For example, ASMO [98] leverages class activation maps from a neural classifier and saliency maps from previous non-learning SOD methods to train the SOD network, in an iterative manner.

4) Semantic segmentation is for per-pixel semantic prediction. Though SOD is class-agnostic, high-level semantics play a crucial role in saliency modeling. Thus, the task of semantic segmentation can also be integrated into SOD learning. A recent SOD model, SSNet [103], is developed upon this idea. It uses a saliency aggregation module to predict a saliency score of each category. Then, a segmentation network is used to produce segmentation masks of all the categories. These masks are finally aggregated (according to corresponding saliency scores) to produce a SOD map.

5) Contour/edge detection refers to the task of detecting obvious object boundaries in images, which are informative of salient objects. Thus, it is also explored in SOD modeling. For example, PAGE-Net [92] learns an edge detection module and embeds edge cues into the main SOD stream in a top-down manner, leading to better edge-preserving results.

6) Image Captioning can provide extra knowledge about the main content of visual scenes, enabling SOD models to better capture high-level semantics. This has been explored in CapSal [100], which incorporates semantic context from a captioning network with local-global visual cues to achieve improved performance for detecting salient objects.

2.4 Object-/Instance-Level SOD

According to whether or not they can identify different salient object instances, current deep SOD models can be categorized into *object-level* and *instance-level* methods.

2.4.1 Object-Level Methods

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

TABLE 4

Statistics of popular SOD datasets, including the number of images, number of salient objects per image, area ratio of the salient objects in images, annotation type, image resolution, and existence of fixation data. See §3 for more detailed descriptions.

	Dataset	Year	Publ.	#Img.	#Obj.	Obj. Area(%)	SOD Annotation	Resolution	Fix.
Early	MSRA-A [30]	2007	CVPR	1,000/20,840	1-2	-	bounding-box object-level	-	
	MSRA-B [30]	2007	CVPR	5,000	1-2	20.82 \pm 10.29	bounding-box object-level, pixel-wise object-level	$\max(w, h) = 400$, $\min(w, h) = 126$	
	SED1 [128]	2007	CVPR	100	1	26.70 \pm 14.26	pixel-wise object-level	$\max(w, h) = 465$, $\min(w, h) = 125$	
	SED2 [128]	2007	CVPR	100	2	21.42 \pm 18.41	pixel-wise object-level	$\max(w, h) = 300$, $\min(w, h) = 144$	
	ASD [31]	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	SOD [129]	2010	CVPR-W	300	1-4+	27.99 \pm 19.36	pixel-wise object-level	$\max(w, h) = 481$, $\min(w, h) = 321$	
	MSRA10K [107]	2015	TPAMI	10,000	1-2	22.21 \pm 10.09	pixel-wise object-level	$\max(w, h) = 400$, $\min(w, h) = 144$	
	ECSSD [55]	2015	TPAMI	1,000	1-4+	23.51 \pm 14.02	pixel-wise object-level	$\max(w, h) = 400$, $\min(w, h) = 139$	
	DUT-OMRON [56]	2013	CVPR	5,168	1-4+	14.85 \pm 12.15	pixel-wise object-level	$\max(w, h) = 401$, $\min(w, h) = 139$	✓
	PASCAL-S [108]	2014	CVPR	850	1-4+	24.23 \pm 16.70	pixel-wise object-level	$\max(w, h) = 500$, $\min(w, h) = 139$	✓
	HKU-IS [27]	2015	CVPR	4,447	1-4+	19.13 \pm 10.90	pixel-wise object-level	$\max(w, h) = 500$, $\min(w, h) = 100$	
	DUTS [97]	2017	CVPR	15,572	1-4+	23.17 \pm 15.52	pixel-wise object-level	$\max(w, h) = 500$, $\min(w, h) = 100$	
Special	SOS [112]	2015	CVPR	6,900	0-4+	41.22 \pm 25.35	object number, bounding-box (<i>train set</i>)	$\max(w, h) = 6132$, $\min(w, h) = 80$	
	MSO [112]	2015	CVPR	1,224	0-4+	39.51 \pm 24.85	object number, bounding-box instance-level	$\max(w, h) = 3888$, $\min(w, h) = 120$	
	ILSO [70]	2017	CVPR	1,000	1-4+	24.89 \pm 12.59	pixel-wise instance-level	$\max(w, h) = 400$, $\min(w, h) = 142$	
	XPIE [130]	2017	CVPR	10,000	1-4+	19.42 \pm 14.39	pixel-wise object-level, geographic information	$\max(w, h) = 500$, $\min(w, h) = 130$	
	SOC [131]	2018	ECCV	6,000	0-4+	21.36 \pm 16.88	pixel-wise instance-level, object category, attribute	$\max(w, h) = 849$, $\min(w, h) = 161$	
	COCO-CapSal [100]	2019	CVPR	6,724	1-4+	23.74 \pm 17.00	pixel-wise object-level, image caption	$\max(w, h) = 640$, $\min(w, h) = 480$	
	HRSOD [73]	2019	ICCV	2,010	1-4+	21.13 \pm 15.14	pixel-wise object-level	$\max(w, h) = 10240$, $\min(w, h) = 600$	

2.4.2 Instance-Level Methods

Instance-level SOD methods further identify individual object instances in the detected salient regions, which is crucial for practical applications that need finer distinctions, such as semantic segmentation [132] and multi-human parsing [133]. As an early attempt, MSRNet [70] performs salient instance detection by decomposing it into three sub-tasks, *i.e.*, pixel-level saliency prediction, salient object contour detection and salient instance identification. It jointly performs the first two sub-tasks by integrating deep features for several different scaled versions of the input image. The last sub-task is solved by multi-scale combinatorial grouping [125] to generate salient object proposals from the detected contours and filter out noisy or overlapping ones.

3 SOD DATASETS

With the rapid development of SOD, numerous datasets have been introduced. Table 4 summarizes 19 SOD datasets, which are highly representative and widely used for training or benchmarking, or collected with specific properties.

3.1 Quick Overview

In an attempt to facilitate understanding of SOD datasets, we present some main take-away points of this section.

- Compared with early datasets [30], [31], [128], recent ones [27], [56], [97], [107] are typically more advanced with less center bias, improved complexity, and increased scale. They are thus better-suited for training and evaluation, and likely to have longer life-spans.
- Some other recent datasets [70], [73], [100], [112], [130], [131] are enriched with more diverse annotations (*e.g.*, subtitizing, captioning), representing new trends in the field.

More in-depth discussions regarding generalizability and difficulty of several famous datasets will be presented in §5.6.

3.2 Early SOD Datasets

Early SOD datasets typically contain simple scenes where 1-2 salient objects stand out from a clear background.

- **MSRA-A** [30] contains 20,840 images. Each image has only one noticeable and eye-catching object, annotated by a bounding-box. As a subset of MSRA-A, MSRA-B has 5,000 images and less ambiguity w.r.t. the salient object.
- **SED** [128]¹ comprises a single-object subset and a two-object subset; each has 100 images with mask annotations.
- **ASD** [31]², also a subset of MSRA-A, has 1,000 images with pixel-wise ground-truths.

3.3 Popular Modern SOD Datasets

Recent SOD datasets tend to include more challenging and general scenes with relatively complex backgrounds and multiple salient objects. All have pixel-wise annotations.

- **SOD** [129]³ consists of 300 images, constructed from [134]. Many images have more than one salient object that is similar to the background or touches image boundaries.
- **MSRA10K** [107]⁴, also known as THUS10K, contains 10,000 images selected from MSRA-A and covers all the images in ASD. Due to its large scale, MSRA10K is widely used to train deep SOD models (see Table 3).
- **ECSSD** [55]⁵ is composed of 1,000 images with semantically meaningful but structurally complex natural contents.
- **DUT-OMRON** [56]⁶ has 5,168 images of complex backgrounds and diverse content, with pixel-wise annotations.
- **PASCAL-S** [108]⁷ comprises 850 challenging images selected from the PASCAL VOC2010 *val* set [113]. With eye-fixation records, non-binary salient-object mask annotations are provided. Note that the saliency value of a pixel is calculated as the ratio of subjects that select the segment containing this pixel as salient.
- **HKU-IS** [27]⁸ has 4,447 complex scenes that typically contain multiple disconnected objects with diverse spatial distributions and similar fore-/background appearances.

1. http://www.wisdom.weizmann.ac.il/~vision/Seg_Evaluation_DB
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/hsalient>
6. <http://saliencydetection.net/dut-omron/>
7. <http://cbi.gatech.edu/salobj/>
8. https://i.cs.hku.hk/~gbli/deep_saliency.html

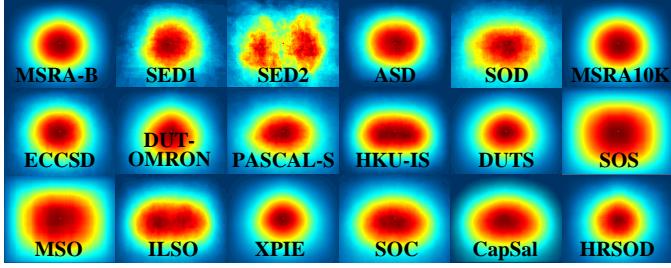


Fig. 3. Annotation distributions of SOD datasets (see §3 for details).

- **DUTS** [97]⁹ is a large-scale dataset, where the 10,553 training images were selected from the ImageNet *train/val* set [114], and the 5,019 test images are from the ImageNet *test* set and SUN [135]. Since 2017, SOD models are typically trained on DUTS (Table 3).

3.4 Other Special SOD Datasets

In addition to the above “standard” SOD datasets, some special ones have also recently been proposed, leading to new research directions.

- **SOS** [112]¹⁰ is created for SOD subtitizing [127]. It contains 6,900 images (*training* set: 5,520, *test* set: 1,380). Each image is labeled as containing 0, 1, 2, 3 or 4+ salient objects.
- **MSO** [112]¹¹ is a subset of SOS-*test* [112], covering 1,224 images. It has a more balanced distribution of the number of salient objects. Each object has a bounding-box annotation.
- **ILSO** [70]¹² contains 1,000 images with precise instance-level annotations and coarse contour labeling.
- **XPIE** [130]¹³ has 10,000 images with pixel-wise labels. It has three subsets: *Set-P* has 625 images of places-of-interest with geographic information; *Set-I* 8,799 images with object tags; and *Set-E* 576 images with eye-fixation records.
- **SOC** [131]¹⁴ consists of 6,000 images with 80 common categories. Half of the images contain salient objects, while the remaining have none. Each image containing salient objects is annotated with an instance-level ground-truth mask, object category, and challenging factors. The non-salient object subset has 783 texture images and 2,217 real-scene images.
- **COCO-CapSal** [100]¹⁵ is built from COCO [115] and SALICON [111]. Salient objects were first roughly localized using the mouse-click data in SALICON, then precisely annotated according to the instance masks in COCO. The dataset has 5,265 and 1,459 images for training and testing, respectively.
- **HRSOD** [73]¹⁶ is the first *high-resolution* dataset for SOD. It contains 1,610 training and 400 testing images collected from websites. Pixel-wise ground-truths are provided.

3.5 Discussion

As shown in Table 4, *early SOD datasets* [30], [31], [128] are comprised of simple images with 1-2 salient objects per image, and only provide rough bounding-box annotations, which are insufficient for reliable evaluations [31], [136]. Performance on these datasets has become saturated. *Modern datasets* [27], [55], [56], [97], [107] are typically large-scale and offer precise pixel-wise ground-truths. The scenes are more complex and general, and usually contain multiple salient objects. Some *special datasets* contain challenging scenes with background only [112], [131], provide more fine-grained, instance-level SOD ground-truths [70], [131] or include other annotations such as image captions [100], inspiring new research directions and applications. Fig. 3 depicts the annotation distributions of 18 SOD datasets. Here are some essential conclusions: 1) Some datasets [30], [31], [97], [107] have significant center bias; 2) Datasets [27], [70], [100] have more balanced location distributions for salient objects; and 3) MSO [112] has less center bias, as only bounding-box annotations are provided. We analyze the generalizability and difficulty of several famous SOD datasets in-depth in §5.6.

4 EVALUATION METRICS

This section reviews popular object-level SOD evaluation metrics, *i.e.*, Precision-Recall (PR), F-measure [31], Mean Absolute Error (MAE) [33], weighted F_β measure (Fbw) [137], Structural measure (S-measure) [138], and Enhanced-alignment measure (E-measure) [139].

4.1 Quick Overview

To better understand the characteristics of different metrics, a quick overview of the main conclusions for this section are provided as follows.

- PR, F-measure, MAE, and Fbw address *pixel-wise* errors, while S-measure and E-measure consider *structure cues*.
- Among pixel-level metrics, PR, F-measure, and Fbw fail to consider true negative pixels, while MAE can remedy this.
- Among structured metrics, S-measure is more favored than E-measure, as SOD addresses continuous saliency estimates.
- Considering popularity, advantages and completeness, F-measure, S-measure and MAE are the most recommended and are thus used for our performance benchmarking in §5.2.

4.2 Metric Details

- **PR** is calculated based on the binarized salient object mask and ground-truth:

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

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 binarize the prediction. Each threshold produces a pair of precision/recall values to form a PR curve for describing model performance.

- **F-measure** [31] comprehensively considers both precision and recall by computing the weighted harmonic mean:

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

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/zhanglndl/code-and-dataset-for-CapSal>

Empirically, β^2 is set to 0.3 [31] to put more emphasis on precision. Instead of plotting the whole F-measure curve, some methods only report *maximal* F_β , or binarize the predicted saliency map by an adaptive threshold, *i.e.*, twice the mean value of the saliency prediction, and report *mean* F .

- **MAE** [33] measures the average pixel-wise absolute error between normalized saliency prediction map $S \in [0, 1]^{W \times H}$ and binary ground-truth mask $G \in \{0, 1\}^{W \times H}$:

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

- **Fbw** [137] intuitively generalizes F-measure by alternating the way of calculating 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:

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

- **S-measure** [138] evaluates the structural similarity between the real-valued saliency map and the binary ground-truth. It considers object-aware (S_o) and region-aware (S_r) structure similarities:

$$S = \alpha \times S_o + (1 - \alpha) \times S_r, \quad (5)$$

where α is empirically set to 0.5.

- **E-measure** [139] considers global means of the image and local pixel matching simultaneously:

$$Q_S = \frac{1}{W \times H} \sum_{i=1}^W \sum_{j=1}^H \phi_S(i, j), \quad (6)$$

where ϕ_S is the enhanced alignment matrix, reflecting the correlation between S and G after subtracting their global means, respectively.

4.3 Discussion

These measures are typically based on *pixel-wise* errors while ignoring *structural* similarities, with S-measure and E-measure being the only exceptions. F-measure and E-measure are designed for assessing *binarized* saliency prediction maps, while PR, MAE, Fbw, and S-measure are for *non-binary* map evaluation.

Among pixel-level metrics, the PR curve is classic. However, precision and recall cannot fully assess the quality of saliency predictions, since high-precision predictions may only highlight a part of salient objects, while high-recall predictions are typically meaningless if all the pixels are predicted as being salient. In general, a high-recall response may come at the expense of reduced precision, and vice versa. F-measure and Fbw are thus used to consider precision and recall simultaneously. However, overlap-based metrics (*i.e.*, PR, F-measure, and Fbw) do not consider the true negative saliency assignments, *i.e.*, the pixels correctly marked as non-salient. Thus, these metrics favor methods that successfully assign high saliency to salient pixels but fail to detect non-salient regions [50]. MAE can remedy this, but it performs poorly when salient objects are small. For the structure-/image-level metrics, S-measure is more popular than E-measure, as SOD focuses on continuous predictions.

Considering the popularity and characteristics of existing metrics and completeness of evaluation, F-measure (*maximal* F_β), S-measure and MAE are our top recommendations.

5 BENCHMARKING AND EMPIRICAL ANALYSIS

This section provides empirical analyses to shed light on some key challenges in the field. Specifically, with our large-scale benchmarking (§5.2), we first conduct an attribute-based study to better understand the benefits and limitations of current arts (§5.3). Then, we study the robustness of SOD models against input perturbations, *i.e.*, random exerted noises (§5.4) and manually designed adversarial samples (§5.5). Finally, we quantitatively assess the generalizability and difficulty of current mainstream SOD datasets (§5.6).

5.1 Quick Overview

For ease of understanding, we compile important observations and conclusions from subsequent experiments below.

- **Overall benchmarks** (§5.2). As shown in Table 5, deep SOD models significantly outperform heuristic ones, and the performance on some datasets [27], [55] has become saturated. [82], [193], [101], [102] are current state-of-the-arts.
- **Attribute-based analysis** (§5.3). Results in Table 7 reveal that deep methods show significant advantages in detecting semantic-rich objects, such as animal. Both deep and non-deep methods face difficulties with small salient objects. For application scenarios, indoor scenes pose great challenges, highlighting potential directions for future efforts.
- **Robustness against random perturbations** (§5.4). As shown in Table 9, surprisingly, deep methods are more sensitive than heuristic ones to random input perturbations. Both types of methods demonstrate more robustness against *Rotation*, while being fragile towards *Gaussian blur* and *Gaussian noise*.
- **Adversarial attack** (§5.5). Table 10 suggests that adversarial attacks cause drastic degradation in performance for deep SOD models, and are even worse than that of random perturbations. However, attacks rarely transfers between different SOD networks.
- **Generalizability and difficulty of datasets** (§5.6). Table 11 shows that DUTS-train [97] is a good choice for training deep SOD models as it has the best generalizability, while SOC [131], DUT-OMRON [56], and DUTS-test [97] are more suitable for evaluation due to their difficulty.

5.2 Performance Benchmarking

Table 5 shows the performances of 44 state-of-the-art deep SOD models and three top-performing classic methods (suggested by [44]) on six most popular modern datasets. The performance is measured by three metrics, *i.e.*, *maximal* F_β , S-measure and MAE, as recommended in §4.3. All the benchmarked models are representative, and have publicly available implementations or saliency prediction results. For performance benchmarking, we either use saliency maps provided by the authors or run their official codes. It is worth mentioning that, for some methods, our benchmarking results are inconsistent with their reported scores. There are several reasons. First, our community long lacked an open, universally-adopted evaluation tool, while there are many implementation factors would influence the evaluation scores, such as input image resolution, threshold step, *etc.* Second, some methods [66], [69], [74], [76], [85], [100] use *mean* F-measure instead of *maximal* F-measure for performance evaluation. Third, for some methods [39], [76],

TABLE 5

Benchmarking results of 44 state-of-the-art deep SOD models and 3 top-performing classic SOD methods on 6 famous datasets (§5.2). Here max F, S, and M indicate maximal F_β , S-measure, and MAE, respectively. The three best scores are marked in red, blue, and green, respectively.

Dataset		ECSSD [55]			DUT-OMRON [56]			PASCAL-S [108]			HKU-IS [27]			DUTS-test [97]			SOD [129]			
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 [35]	.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 [36]	.684	.714	.165	.541	.653	.171	.599	.656	.196	.695	.729	.138	.522	.639	.176	.615	.638	.213	
2015	MCDL [29]	.816	.803	.101	.670	.752	.089	.706	.721	.143	.787	.786	.092	.634	.713	.105	.689	.651	.182	
	LEGS [28]	.805	.786	.118	.631	.714	.133	‡	‡	‡	.736	.742	.119	.612	.696	.137	.685	.658	.197	
	MDF [27]	.797	.776	.105	.643	.721	.092	.704	.696	.142	.839	.810	.129	.657	.728	.114	.736	.674	.160	
2016	ELD [60]	.849	.841	.078	.677	.751	.091	.782	.799	.111	.868	.868	.063	.697	.754	.092	.717	.705	.155	
	DHSNet [38]	.893	.884	.060	‡	‡	‡	.799	.810	.092	.875	.870	.053	.776	.818	.067	.790	.749	.129	
	DCL [104]	.882	.868	.075	.699	.771	.086	.787	.796	.113	.885	.877	.055	.742	.796	.149	.786	.747	.195	
	°MAP [37]	.556	.611	.213	.448	.598	.159	.521	.593	.207	.552	.624	.182	.453	.583	.181	.509	.557	.236	
	CRPSD [105]	.915	.895	.048	-	-	-	.864	.852	.064	.906	.885	.043	-	-	-	-	-	-	
	RFCN [63]	.875	.852	.107	.707	.764	.111	.800	.798	.132	.881	.859	.089	.755	.859	.090	.769	.794	.170	
2017	MSRNet [70]	.900	.895	.054	.746	.808	.073	.828	.838	.081	‡	‡	‡	.804	.839	.061	.802	.779	.113	
	DSS [39]	.906	.882	.052	.737	.790	.063	.805	.798	.093	‡	‡	‡	.796	.824	.057	.805	.751	.122	
	†WSS [97]	.879	.811	.104	.725	.730	.110	.804	.744	.139	.878	.822	.079	.878	.822	.079	.807	.675	.170	
	DLS [65]	.826	.806	.086	.644	.725	.090	.712	.723	.130	.807	.799	.069	-	-	-	-	-	-	
	NLDF [75]	.889	.875	.063	.699	.770	.080	.795	.805	.098	.888	.879	.048	.777	.816	.065	.808	.889	.125	
	Amulet [76]	.905	.894	.059	.715	.780	.098	.805	.818	.100	.887	.886	.051	.750	.804	.085	.773	.757	.142	
	FSN [72]	.897	.884	.053	.736	.802	.066	.800	.804	.093	.884	.877	.044	.761	.808	.066	.781	.755	.127	
	SBF [83]	.833	.832	.091	.649	.748	.110	.726	.758	.133	.821	.829	.078	.657	.743	.109	.740	.708	.159	
	SRM [71]	.905	.895	.054	.725	.798	.069	.817	.834	.084	.893	.887	.046	.798	.836	.059	.792	.741	.128	
	UCF [66]	.890	.883	.069	.698	.760	.120	.787	.805	.115	.874	.875	.062	.742	.782	.112	.763	.753	.165	
2018	RADF [78]	.911	.894	.049	.761	.817	.055	.800	.802	.097	.902	.888	.039	.792	.826	.061	.804	.757	.126	
	BDMP [84]	.917	.911	.045	.734	.809	.064	.830	.845	.074	.910	.907	.039	.827	.862	.049	.806	.786	.108	
	DGRL [85]	.916	.906	.043	.741	.810	.063	.830	.839	.074	.902	.897	.037	.805	.842	.050	.802	.771	.105	
	PAGR [86]	.904	.889	.061	.707	.775	.071	.814	.822	.089	.897	.887	.048	.817	.838	.056	.761	.716	.147	
	RSDDNet [79]	.880	.788	.173	.715	.644	.178	‡	‡	‡	.871	.787	.156	.798	.720	.161	.790	.668	.226	
	ASNet [87]	.925	.915	.047	‡	‡	‡	‡	.848	.861	.070	.912	.906	.041	.806	.843	.061	.801	.762	.121
	PiCANet [40]	.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 [88]	.908	.893	.056	.753	.814	.062	.800	.799	.101	.901	.887	.045	.807	.839	.059	.810	.764	.124	
2019	AFNet [89]	.924	.913	.042	.759	.826	.057	.844	.849	.070	.910	.905	.036	.838	.867	.046	.809	.774	.111	
	BASNet [90]	.931	.916	.037	.779	.836	.057	.835	.838	.076	.919	.909	.032	.838	.866	.048	.805	.769	.114	
	CapSal [100]	.813	.826	.077	.535	.674	.101	.827	.837	.073	.842	.851	.057	.772	.818	.061	.669	.694	.148	
	CPD [80]	.926	.918	.037	.753	.825	.056	.833	.848	.071	.911	.905	.034	.840	.869	.043	.814	.767	.112	
	MLSLNet [91]	.917	.911	.045	.734	.809	.064	.835	.844	.074	.910	.907	.039	.828	.862	.049	.806	.786	.108	
	†MWS [81]	.859	.827	.099	.676	.756	.108	.753	.768	.134	.835	.818	.086	.720	.759	.092	.772	.700	.170	
	PAGE-Net [92]	.926	.910	.037	.760	.819	.059	.829	.835	.073	.910	.901	.031	.816	.848	.048	.795	.763	.108	
	PS [94]	.930	.918	.041	.789	.837	.061	.837	.850	.071	.913	.907	.038	.835	.865	.048	.824	.800	.103	
	PoolNet [93]	.937	.926	.035	.762	.831	.054	.858	.865	.065	.923	.919	.030	.865	.886	.037	.831	.788	.106	
	BANet-R [101]	.939	.924	.035	.782	.832	.059	.847	.852	.070	.923	.913	.032	.858	.879	.040	.842	.791	.106	
	EGNet-R [82]	.936	.925	.037	.777	.841	.053	.841	.852	.074	.924	.918	.031	.866	.887	.039	.854	.802	.099	
	HRSOD-DH [73]	.911	.888	.052	.692	.762	.065	.810	.817	.079	.890	.877	.042	.800	.824	.050	.735	.705	.139	
	JDFPR [95]	.915	.907	.049	.755	.821	.057	.827	.841	.082	.905	.903	.039	.792	.836	.059	.792	.763	.123	
	SCRN [102]	.937	.927	.037	.772	.836	.056	.856	.869	.063	.921	.916	.034	.864	.885	.040	.826	.787	.107	
	SSNet [103]	.889	.867	.046	.708	.773	.056	.793	.807	.072	.876	.854	.041	.769	.784	.049	.713	.700	.118	
	TSPOANet [41]	.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.

the evaluation scores of finally released saliency maps are inconsistent with the ones reported in papers. We hope that our performance benchmarking, publicly released evaluation tools and SOD maps could help our community build an open and standardized evaluation system and ensure consistency and procedural correctness for results and conclusions produced by different parties.

Not surprisingly, data-driven models greatly outperform conventional heuristic ones, due to their strong learning ability for visually salient pattern modeling. In addition, the performance has gradually increased since 2015, demonstrating well the advancement of deep learning techniques. However, after 2018, the rate of improvement began decrasing, calling for more effective model designs and new machine learning technologies. We also find that the performances tend to be saturated on older SOD datasets such as ECSSD [55] and HKU-IS [27]. Hence, among the 44 famous deep SOD models, we would like to nominate

PoolNet [93], BANet [101], EGNet [82], and SCRN [102] as the four state-of-the-art methods, which consistently show promising performance over diverse datasets.

5.3 Attribute-Based Study

Although the community has witnessed the great advances made by deep SOD models, it is still unclear under which specific aspects these models perform well. As there are numerous factors affecting the performance of a SOD algorithm, such as object/scene category, occlusion, etc., it is crucial to evaluate the performance under different scenarios. This can help reveal the strengths and weaknesses of deep SOD models, identify pending challenges, and highlight future research directions towards more robust algorithms.

5.3.1 Hybrid Benchmark Dataset with Attribute Annotations

To enable a deeper analysis and understanding of the performance of an algorithm, it is essential to identify the



Fig. 4. Sample images from the hybrid benchmark consisting of images randomly selected from 6 SOD datasets. Salient regions are uniformly highlighted. Corresponding attributes are listed. See §5.3 for more detailed descriptions.

TABLE 7

Attribute-based study w.r.t. salient object categories, challenges and scene categories. (.) indicates the percentage of images with a specific attribute. *ND-avg* indicates the average score of three heuristic models: HS [35], DRFI [53] and wCtr [36]. *D-avg* indicates the average score of three deep learning models: DGRL [85], PAGR [86] and PiCANet [40]. Best in **red**, and worst with **underlined**. See §5.3 for more details.

Metric	Method	Salient object categories				Challenges								Scene categories			
		<i>Human</i> (26.61)	<i>Animal</i> (38.44)	<i>Artifact</i> (45.67)	<i>NatObj</i> (10.56)	<i>MO</i> (11.39)	<i>HO</i> (66.39)	<i>OV</i> (28.72)	<i>OC</i> (46.50)	<i>CS</i> (40.44)	<i>BC</i> (47.22)	<i>CT</i> (74.11)	<i>SO</i> (21.61)	<i>LO</i> (12.61)	<i>Indoor</i> (20.28)	<i>Urban</i> (22.22)	<i>Natural</i> (57.50)
max F↑	*HS [35]	.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 [36]	.557	.621	.624	.682	.639	.625	.605	.620	.522	.612	.606	.469	.689	.578	.613	.618
	DGRL [85]	.820	.881	.830	.728	.783	.846	.829	.830	.781	.842	.834	.724	.873	.800	.848	.840
	PAGR [86]	.834	.890	.787	.725	.743	.819	.778	.809	.770	.797	.822	.760	.802	.788	.796	.828
	PiCANet [40]	.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 6

Descriptions of attributes that often bring difficulties to SOD (see §5.3).

Attr	Description
<i>MO</i>	Multiple Objects. There exist more than two salient objects.
<i>HO</i>	Heterogeneous Object. Salient object regions have distinct colors or illuminations.
<i>OV</i>	Out-of-View. Salient objects are partially clipped by image boundaries.
<i>OC</i>	Occlusion. Salient objects are occluded by other objects.
<i>CS</i>	Complex Scene. Background regions contain confusing objects or rich details.
<i>BC</i>	Background Clutter. Foreground and background regions around the salient object boundaries have similar colors (χ^2 between RGB histograms less than 0.9).
<i>CT</i>	Complex Topology. Salient objects have complex shapes, e.g., thin parts or holes.
<i>SO</i>	Small Object. Ratio between salient object area and image is less than 0.1.
<i>LO</i>	Large Object. Ratio between salient object area and image is larger than 0.5.

key factors and circumstances influencing it [140]. To this end, we construct a *hybrid benchmark* with rich attribute annotations. It consists of 1,800 images randomly selected from six SOD datasets (300 for each), namely SOD [129], ECSSD [55], DUT-OMRON [56], PASCAL-S [108], HKU-IS [27] and DUTS test set [97]. Inspired by [108], [140], we annotate each image with an extensive set of attributes covering typical object types, challenging factors and diverse scene categories. Specifically, the annotated **salient objects** are categorized into *Human*, *Animal*, *Artifact* and *NatObj* (Natural Objects), where *NatObj* includes natural objects such as fruit, plant, mountains, icebergs, lakes, etc. The **challenging factors** describe specific situations that often bring difficulties to SOD, such as occlusions, background clutter, and complex shapes (see Table 6). The image **scenes** include *Indoor*, *Urban* and *Natural*, where the last two indicate different outdoor environments. It is worth mentioning that the attributes are not mutually exclusive. Some sample

images with attribute annotations are shown in Fig. 4. Please note that this benchmark will also be used in §5.4 and §5.5.

For the baselines in our attribute-based analysis, we choose the three top-performing heuristic models again, i.e., HS [35], DRFI [53] and wCtr [36], and three recent famous deep methods, i.e., DGRL [85], PAGR [86] and PiCANet [40]. All three deep models are trained on DUTS-train [97] and have publicly released implementations.

5.3.2 Analysis

In Table 7, we report the performance on subsets of our hybrid dataset characterized by a particular attribute. To provide better insight, in Table 8, we select images with the best-100 and worst-100 model predictions, and compare the portion distributions of attributes w.r.t. the ones over the whole dataset. Below are some important observations drawn from these experiments.

- **'Easy' and 'hard' object categories.** Deep and non-deep SOD models view object categories differently (Table 7). For the deep methods (*D-avg*), *NatObj* is clearly the most challenging one which is probably due to its small number of training samples and complex topologies. *Animal* appears to be the easiest, which can be attributed to its significant semantics. By contrast, traditional methods (*ND-avg*) struggle with *Human*, revealing their limitations in capturing high-level semantics. We are surprised to find that the deep models significantly outperform the non-deep ones over almost all the object categories, except *NatObj*. This demonstrates the value of heuristic assumptions in certain scenes and the potential of embedding human prior knowledge into current deep learning schemes.

- **Most and least challenging factors.** Table 7 shows that, interestingly, both deep and non-deep methods handle *LO* well. In addition, both types of methods face difficulties with *SO*, highlighting a promising direction for future efforts. Besides, we find that *CS* and *MO* are challenging for deep models, showing that current solutions still fall short at determining the relative importance of different objects.

TABLE 8

Attribute statistics of top and bottom 100 images based on F-measure. (·) indicates the percentage of the images with a specific attribute. $ND\text{-avg}$ indicates the average results of three heuristic models: HS [35], DRFI [53] and wCtr [36]. D-avg indicates the average results of three deep models: DGRL [85], PAGR [86] and PiCANet [40]. Two largest changes in red if positive, and blue if negative. See §5.3 for more details.

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{OC} (46.50)	\mathcal{CS} (40.44)	\mathcal{BC} (47.22)	\mathcal{CT} (74.11)	\mathcal{SO} (21.61)	\mathcal{LO} (12.61)	Indoor (20.28)	Urban (22.22)	Natural (57.50)
$ND\text{-avg}$	Best (%) change	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
		-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
$D\text{-avg}$	Worst (%) change	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
		+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
	Best (%) change	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
		-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 (%) change	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
		+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

TABLE 9

Input perturbation study on the **hybrid benchmark**. $ND\text{-avg}$ indicates the average score of three heuristic models: HS [35], DRFI [53] and wCtr [36]. $D\text{-avg}$ indicates the average score of three deep learning models: SRM [71], DGRL [85] and PiCANet [40]. Best in red and worst with underline. See §5.4 for more details.

Metric	Method	Original	Gaus. blur ($\sigma=$)		Gaus. noise (var=)		Rotation		Gray
			2	4	0.01	0.08	15°	-15°	
max F↑	*HS [35]	.600	-.012	-.096	-.022	-.057	+.015	+.009	-.104
	*DRFI [53]	.670	-.040	-.103	-.035	-.120	-.009	-.009	-.086
	*wCtr [36]	.611	+.006	-.000	-.024	-.136	-.004	-.003	-.070
	SRM [71]	.817	-.090	-.229	-.025	-.297	-.028	-.029	-.042
	DGRL [85]	.831	-.088	-.365	-.050	-.402	-.031	-.022	-.026
	PiCANet [40]	.848	-.048	-.175	-.014	-.148	-.005	-.008	-.039
	* $ND\text{-avg}$.627	-.015	-.066	-.027	-.104	-.000	-.001	-.087
	$D\text{-avg}$.832	-.075	-.256	-.041	-.282	-.021	-.020	-.037

* Non-deep learning model.

- Most and least difficult scenes.** Deep and heuristic methods perform similarly when faced with different scenes (Table 7). For both types of methods, *Natural* is the easiest, which is reasonable as the scenes are typically simple. Further, though both contain numerous objects, *Indoor* is more challenging than *Urban* as it often suffers from highly unevenly distributed illumination and more complex scenes. Our experiments also show that the utility of SOD models in real, and especially complex, environments is still limited.
- Additional advantages of deep models.** As shown in Table 7, deep models achieve great improvements on semantically rich objects (*Human*, *Animal* and *Artifact*), demonstrating advantages in semantic modeling. This is verified again by their good performance on complex object shapes (\mathcal{HO} , \mathcal{OV} , \mathcal{OC} , \mathcal{CT}). Deep models also narrow the gap between different scene categories (*Indoor* v.s. *Natural*), indicating an improved robustness against various backgrounds.
- Best and worst predictions.** From Table 8, in addition to similar conclusions drawn from Table 7, some unique and interesting observations can be made. First, for deep methods, *NatObj* spans a large range of challenge, containing both the simplest and hardest samples. Thus, future efforts should pay more attention to the hard samples in *NatObj*. In addition, after considering data distribution bias, \mathcal{CS} is the most challenging factor for deep models.

5.4 Robustness Against General Input Perturbations

The robustness of a model lies in its stability against corrupt inputs. Intuitively, the outputs of a robust SOD model should be repeatable on slightly different images with the

same content. However, the recently introduced adversarial examples, *i.e.* maliciously constructed inputs that fool machine learning models, can degrade the performance of deep image classifiers significantly. Current deep SOD models likely face a similar challenge. Therefore, in this section, we examine the robustness of SOD models by comparing their outputs for randomly perturbed inputs, such as noisy or blurred images. Then, in §5.5, we will study the robustness to manually designed adversarial examples.

The input perturbations investigated include *Gaussian blur*, *Gaussian noise*, *Rotation*, and *Gray*. For blurring, we employ Gaussian blur kernels with a sigma of 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 by $+15^\circ$ and -15° , respectively, and cut out the largest box with the original aspect ratio. The gray images are generated using the Matlab `rgb2gray` function.

As in §5.3, we include three popular heuristic models [35], [36], [53] and three deep methods [40], [71], [85] in our experiments. Table 9 shows the results. Overall, compared with deep models, heuristic methods are less sensitive towards input perturbations. The compactness and abstractness of superpixels likely explains much of this. Specifically, heuristic methods are rarely affected by *Rotation*, but perform worse under strong *Gaussian blur*, strong *Gaussian noise* and *Gray*. Deep methods suffer the most under *Gaussian blur* and strong *Gaussian noise*, which may be caused by the damage to shallow-layer features. Deep methods are relatively robust against *Rotation*, revealing the rotation invariance of DNNs brought by the pooling operation. Interestingly, we further find that, among the three deep models, PiCANet [40] demonstrates excellent robustness against a wide range of input perturbations, including *Gaussian blur*, *Gaussian noise*, and *Rotation*. We attribute this to its effective non-local operation. This reveals that effective network designs can improve the robustness to random perturbations.

5.5 Robustness Against Manually Designed Input Perturbations

Given the significant concerns with model robustness to random perturbations, this section presents an analysis focusing specifically on manually designed adversarial perturbations. Recent years have witnessed great advance in SOD driven by the progress of deep learning. However, whether the deep SOD models are as powerful as they seem is a question to worth pondering. Meanwhile, DNNs

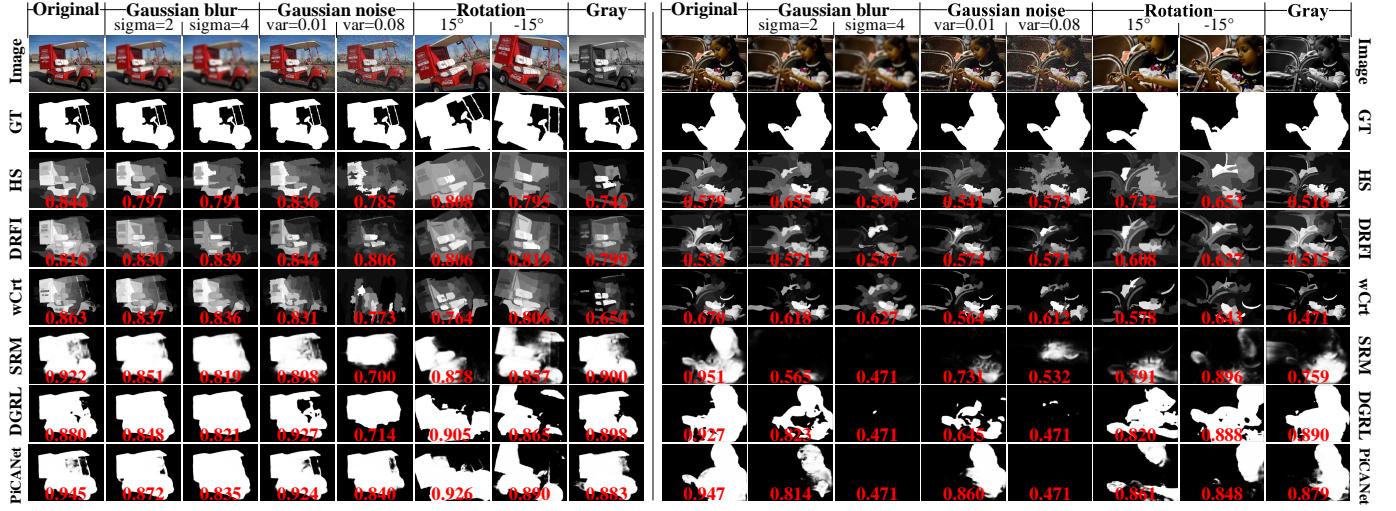


Fig. 5. Examples of saliency prediction under various input perturbations. The max F values are denoted in red. See §5.4 for more details.

have been previously found to be susceptible to adversarial attacks, where visually imperceptible perturbations lead to completely different predictions [141]. Though intensively studied in classification tasks, adversarial attacks in SOD are rarely explored. As SOD has been integrated as a critical part in many security systems and commercial projects, SOD models also have potential risks of being attacked. Specifically, SOD plays a significant role in many security systems, for detecting the candidates of interest targets from remote sensing images [142], video surveillance data [143], or sensor signals of autonomous vehicles [144]. In such situation, examining the robustness of SOD models is rather important because the insecurity of SOD modules may cause severe losses, *e.g.*, the criminals may use inconspicuous adversarial perturbations to fool SOD modules and then cheat the surveillance systems. Besides, SOD has benefited many commercial projects such as photo editing [20], and image/video compression [145]. The adversarial attacks launched by hackers on the embedded SOD modules would inevitably affect the functioning of commercial products and impacting users, causing losses for the developers and companies. Therefore, studying the robustness of SOD models is crucial for defending these applications against malicious attacks. In this section, we study the robustness against adversarial attacks and transferability of adversarial examples targeting different SOD models. Our observations are expected to shed light on adversarial attacks and defenses for SOD, providing a better understanding of vulnerabilities of deep SOD models and improving the robustness of SOD involved practical applications.

5.5.1 Robustness of SOD Against Adversarial Attacks

For measuring the robustness of deep SOD models, we adopt and modify an adversarial attack algorithm designed for semantic segmentation, *i.e.*, Dense Adversary Generation (DAG) [146]. We choose three representative deep models, *i.e.*, SRM [71], DGRL [85] and PiCANet [40] for our study. The experiment is conducted on the hybrid benchmark introduced in §5.3. Following [146], we measure the perceptibility of the adversarial examples by computing the average perceptibility of the adversarial perturbations generated

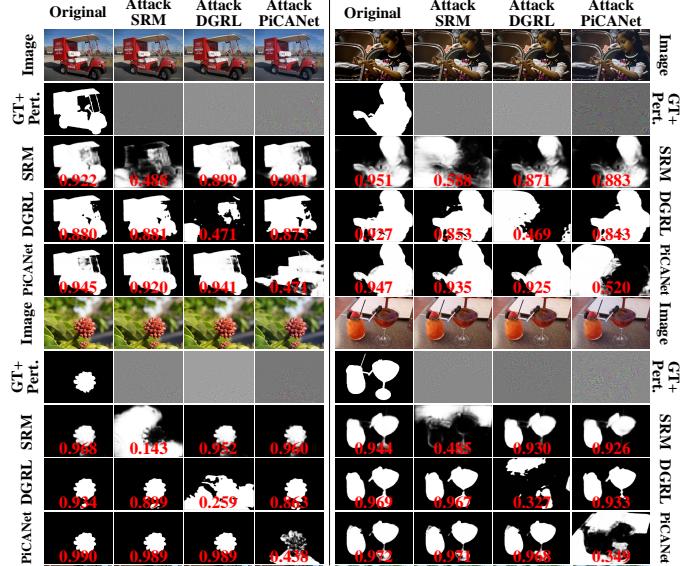


Fig. 6. Examples of SOD prediction under adversarial perturbations of different target networks. The perturbations are magnified by 10 for better visualization. Red for max F. See §5.5 for details.

from the hybrid benchmark. The values for the three models are 3.54×10^{-3} , 3.57×10^{-3} , and 3.51×10^{-3} , respectively.

Exemplar adversarial cases are shown in Fig. 6. As can be seen, the adversarial attacks can prevent the SOD models from producing reliable salient object candidates. Quantitative results are listed in Table 10. The underlined entries of Table 10 reveal that the three deep SOD models investigated are vulnerable to adversarial perturbations of the inputs. However, as can be observed by comparing Tables 9 and 10, the models are more robust to random input perturbations. These differences in robustness might be interpreted by the distance from the inputs to the decision boundary in high dimensional space. The intentionally designed adversarial inputs often lie closer to the decision boundary than the random inputs [147], and can thus more easily cause pixel-wise misclassification.

TABLE 10

Results for adversarial attack experiments. Max F↑ on the **hybrid benchmark** is presented when exerting adversarial perturbations from different models. Worst results are underline. See §5.5 for details.

Attack from	SRM [71]	DGRL [85]	PiCANet [40]
None	.817	.831	.848
SRM [71]	<u>.263</u>	.780	.842
DGRL [85]	.778	<u>.248</u>	.844
PiCANet [40]	.772	.799	<u>.253</u>

5.5.2 Transferability Across Networks

Previous research has revealed that adversarial perturbations can be transferred across networks, *i.e.* adversarial examples targeting one model can mislead another without any modification [148]. This transferability is widely used for black-box attacks against real-world systems. To investigate the transferability of perturbations for deep SOD models, we use the adversarial perturbation computed on one SOD model to attack another.

Table 10 shows the experimental results for the three models under investigation (SRM [71], DGRL [85] and PiCANet [40]). While the DAG attack leads to severe performance drops for the targeted model (see the diagonal), it causes much less degradation to other models, *i.e.*, the transferability between models of different network structures is weak for SOD task, which is similar to the transferability observed for semantic segmentation, as analyzed in [146]. This may be because the gradient directions of different models are orthogonal to each other [149], so the gradient-based attack in the experiment transfers poorly to non-targeted models. However, adversarial images generated from an ensemble of multiple models might generate non-targeted adversarial instances with better transferability [149], which would be a great threat to deep SOD models.

5.6 Cross-Dataset Generalization Evaluation

Datasets are responsible for much of the recent progress in SOD, not just as sources for training deep models, but also as means for measuring and comparing performance. Datasets are collected with the goal of representing the visual world, and to summarize the algorithm as a single number (*i.e.*, benchmark score). A concern thus arises: it is necessary to evaluate how well a particular dataset represents the real world; or, more specifically, to quantitatively measuring the dataset’s generalization ability. Unfortunately, previous studies [44] are quite limited – mainly concerning the degrees of center bias in different SOD datasets. Here, we follow [150] to assess how general SOD datasets are. We study the generalization and difficulty of several mainstream SOD datasets by performing a cross-dataset analysis, *i.e.*, training on one dataset, and testing on the others. We expect our experiments to stimulate discussion in the community regarding this essential but largely neglected issue.

We first train a typical SOD model on one dataset, and then explore how well it generalizes to a representative set of other datasets, compared with its performance on the “native” test set. Specifically, we implement the typical SOD model as a bottom-up/top-down structure, which has been the most standard and popular SOD architecture these years and is the basis of many current top-performing models [82],

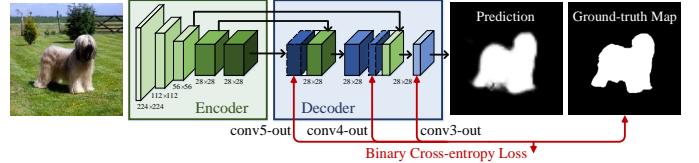


Fig. 7. Network architecture of the SOD model used in cross-dataset generalization evaluation. See §5.6 for more detailed descriptions.

TABLE 11

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 §5.6 for details.

Test on:	MSRA-10K [107]	ECSSD [55]	DUT-OMRON [56]	HKU-IS [27]	DUTS [97]	SOC [131]	Self	Mean others	Percent drop↓	
Train on:	MSRA10K [107]	.875	.818	.660	.849	.671	.617	.875	.723	17%
MSRA10K [107]	.875	.818	.660	.849	.671	.617	.875	.723	17%	
ECSSD [55]	.844	<u>.831</u>	.630	.833	.646	.616	.831	.714	14%	
DUT-OMRON [56]	.795	.752	<u>.673</u>	.779	.623	.567	.673	.703	-5%	
HKU-IS [27]	.857	.838	.695	<u>.880</u>	.719	.639	.880	.750	15%	
DUTS [97]	.857	.834	.647	.860	<u>.665</u>	.654	.665	.770	-16%	
SOC [131]	.700	.670	.517	.666	.514	<u>.593</u>	.593	.613	-3%	
Mean others	.821	.791	.637	.811	.640	.614	-	-	-	

[93], [101], [102]. As shown in Fig. 7, the encoder part is borrowed from VGG16 [151], and the decoder consists of three convolutional layers that gradually refine the saliency prediction. We pick six representative datasets [27], [55], [56], [97], [107], [131]. For each dataset, we train the SOD model with 800 randomly selected training images and test it on 200 other validation images. Please note that a total of 1,000 is the maximum possible number of images considering the size of the smallest selected dataset, ECSSD [55].

Table 11 summarizes the results of cross-dataset generalization, measured by max F. Each column corresponds to the performance when training on all the datasets separately and testing on one. Each row indicates training on one dataset and testing on all of them. Since our training/testing protocol is different from the one used in the benchmarks mentioned in previous sections, the actual performance numbers are not meaningful. Rather, it is the relative performance difference that matters. Not surprisingly, we observe that the best results are achieved when training and testing on the same dataset. By looking at the numbers across each column, we can determine how easy a dataset is for models trained on the other datasets. By looking at the numbers across one row, we can determine how good a dataset is at generalizing to the others. We find that SOC [131] is the most difficult dataset (lowest column, Mean others 0.614). MSRA10K [107] appears to be the easiest one (highest column, Mean others 0.811), and generalizes the worst (highest row, Percent drop 17%). DUTS [97] is shown to have the best generalization ability (lowest row, Percent drop -16%).

Based on these analyses, we would make the following recommendations for SOD datasets: 1) For training deep models, DUTS [97] is a good choice because it has the best generalizability. 2) For testing, SOC [131] is good for assessing the worst-case performances, since it is the most challenging dataset. DUT-OMRON [56] and DUTS-test [97] deserve more considerations as they are also very difficult.

6 MORE DISCUSSIONS

Our previous systematic review and empirical studies characterized the models (§2), datasets (§3), metrics (§4), and challenges (§5) of deep SOD. Here we further posit active research directions, and outline several open issues.

6.1 Model Design

Based on the review of deep SOD network architectures in §2.1, as well as recent advances in related fields, we here discuss several essential directions for SOD model design.

- **Network topology.** Network topology determines the within-network information flow, which directly affects model capacity and training difficulty and thus influences the best possible performance. To figure out an effective network topology for SOD, diverse architectures have been explored (§2.1), such as multi-stream networks, side-out fusion networks, as well as bottom-up/top-down networks. However, all these network architectures are hand-designed. Thus, a promising direction would be to use *automated machine learning* (AutoML) algorithms, such as *neural architecture search* [152], to automatically search for the best-performing SOD network topology.
- **Loss function.** Most deep SOD methods are trained with the standard binary cross-entropy loss, which may fail to fully capture the quality factors for the SOD task. Only a few efforts have been made to derive losses from SOD evaluation metrics [87]. Thus, it is worth exploring more effective SOD loss functions, such as the mean intersection-over-union loss [153] and affinity field matching loss [154].
- **Adaptive computation.** Currently, all deep SOD models are fixed feed-forward structures. However, most parameters model high-level features that, in contrast to low-level and many mid-level concepts, cannot be broadly shared across categories/scenes. As such, we would like to ask the following question: What if a SOD model could directly execute certain layers that can best explain the saliency patterns in a given scene? To answer this, one could leverage adaptive computation techniques [155], [156] to vary the amount of computation on-the-fly, *i.e.*, by selectively activating part of the network in an input-dependent fashion. This could bring a better trade-off between network depth and computational cost. On the other hand, adapting inference pathways for different inputs would provide finer-grained discriminative ability for various attributes. Therefore, exploring dynamic network structures in SOD is promising for improving both efficiency and effectiveness.

6.2 Data Collection

Our previous discussions (§3) and analyses (§5.3 and §5.6) on current SOD datasets revealed several factors that are essential for future dataset collection.

- **Annotation inconsistency.** Though existing SOD datasets play a critical role in training and evaluating modern SOD models, annotation inconsistencies among different SOD datasets have essentially been ignored by the community. The inconsistencies are mainly caused by separate subjects and rules/conditions during dataset annotation (see Fig. 8). To ease annotation burdens, most current SOD datasets only have a few human annotators directly identify the salient

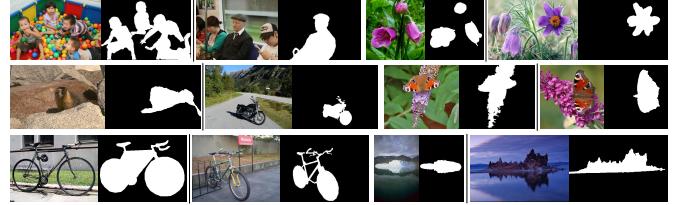


Fig. 8. Examples for annotation inconsistency. Each row shows two exemplar image pairs. See §6.2 for more detailed descriptions.

objects, instead of considering real human eye fixation behavior. Maintaining annotation consistency among newly collected datasets is an important consideration.

- **Coarse v.s. fine annotation.** Modern SOD datasets all have pixel-level annotations, which greatly boosts the performance of deep SOD models. However, pixel-wise ground-truths are very costly to collect considering the complex object boundaries and the intense data requirement. Further, the annotation qualities of different datasets are different (see bicycles in Fig. 8). Finer labels are believed to be essential for high-quality saliency prediction, but usually take more time to collect. Thus, given a limited budget, finding the optimal annotation strategy is an open problem. Some works have studied the relationship between label quality and model performance in semantic segmentation [157], highlighting a possible research direction for SOD dataset collection. In addition, current SOD models typically assume that the annotations are perfect. Thus, it would also be of value to explore robust SOD models that can learn saliency patterns from imperfectly annotated data.

- **Domain-specific SOD datasets.** SOD has shown potential in a wide range of applications, such as autonomous vehicles, video games, medical image processing, *etc.* Due to the different visual appearances and semantic components, the saliency mechanisms in these applications are quite different from that of conventional natural images. Thus, collecting domain-specific datasets might benefit the application of SOD in certain scenarios, as observed in FP for crowds [158], webpages [159] or driving [160], and better connect SOD to the biological top-down visual attention mechanism and human mental state.

6.3 Saliency Ranking and Relative Saliency

Current algorithms seem over-focused on directly regressing the saliency map to pursue a high benchmarking number, while neglecting the fact that the absolute magnitude of values in a saliency map might be less important than the relative saliency values among objects [108]. Though the relative value/rank order is rarely considered in the context of benchmarking metrics (with the exception of [79]), it is crucial for better modeling human visual attention behavior. This is, in essence, a selection process that centers our attention on certain important elements of the surroundings, while blending other relatively unimportant things into the background. This not only hints at one shortcoming of existing benchmarking paradigms and data collection strategies, but also reveals a common limitation of current methods. Current state-of-the-arts fall short at determining the relative importance of objects, such as identifying the

most important person in a crowded room. This is also evidenced by the experiments in §5.3, which show that deep models face great difficulties in complex (\mathcal{CS}), indoor (*Indoor*) or multi-object (\mathcal{MO}) scenes. In other words, deep SOD models, though good at semantic modeling, require higher-level image understanding. Exploring more powerful network designs that explicitly reason the relative saliency and revisiting classic cognitive theories are both promising directions to overcome this issue.

6.4 Linking SOD to Visual Fixations

The strong correlation between eye movements (implicit saliency) and explicit object saliency has been explored throughout history [44], [108], [161]–[163]. However, despite the deep connections between the problems of FP and SOD, the major computational models of the two tasks remain largely distinct; only a few SOD models consider both tasks simultaneously [72], [87], [96]. This is mainly due to the overemphasis on the specific setting of SOD and the design bias of current SOD datasets, which overlooks the connection to eye fixations during data annotation. As stated in [108], such dataset design bias not only creates a discomforting disconnection between FP and SOD, but also further misleads the algorithm designing. Exploring classic visual attention theories in SOD is a promising and crucial direction which could make SOD models more consistent with the visual processing of human visual system and provide better explainability. In addition, the ultimate goal of visual saliency modeling is to understand the underlying rationale of the visual attention mechanism. However, with the current focus on exploring more powerful neural network architectures and beating the latest benchmark numbers on different datasets, have we perhaps lost sight of the original purpose? The solution to these problems requires dense collaborations between the FP and SOD communities.

6.5 Learning SOD in a Weakly-/Unsupervised Manner

Deep SOD methods are typically trained in a fully-supervised manner with a plethora of finely-annotated pixel-level ground-truths. However, it is highly costly and time-consuming to construct a large-scale, well-annotated SOD dataset. Though some efforts have been made to achieve SOD with limited supervision, *i.e.*, by leveraging category-level labels [68], [69], [97] or pseudo pixel-wise annotations [67], [81], [83], [98], [99], there is still a notable gap with the fully-supervised counterparts. In contrast, humans usually learn with little or even no supervision. Since the ultimate goal of visual saliency modeling is to understand the visual attention mechanism, learning SOD in an weakly-/unsupervised manner would be of great value to both the research community and real-world applications. Further, it would also help us understand which factors truly drive our attention mechanism and saliency pattern understanding. Given the massive number of algorithmic breakthroughs over the past few years, we can expect a flurry of innovation towards this promising direction.

6.6 Pre-training with Self-Supervised Visual Features

Current deep SOD methods are typically built on ImageNet-pretrained networks, and fine-tuned on SOD datasets. It is

believed that parameters trained on ImageNet can serve as a good starting point to accelerate the convergence of training and prevent overfitting on smaller-scale SOD datasets. Besides pre-training deep SOD models on the *de facto* dataset, ImageNet, another option is to leverage self-supervised learning techniques [164] to learn effective visual features from a vast amount of unlabeled images/videos. The visual features can be learned through various pretext tasks like image inpainting [165], colorization [166], clustering [167], *etc.*, and can be generalized to other vision tasks. Fine-tuning the SOD models on parameters trained from self-supervised learning is promising to yield better performance compared to the ImageNet initialization.

6.7 Efficient SOD for Real-World Application

Current top-leading deep SOD models are designed to be complicated in order to achieve increased learning capacity and improved performance. However, more ingenuous and light-weight 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 ability caused by model scale deduction should be minimal. To facilitate the application of SOD in real-world scenarios, it is possible to utilize model compression [168] or knowledge distillation [169], [170] techniques to develop compact and fast SOD models with competitive performance. Such compression techniques have already been shown effective in improving generalization ability and alleviating underfitting for training efficient object detection models [171].

7 CONCLUSION

In this paper we present, to the best of our knowledge, the first comprehensive review of SOD focusing on deep learning techniques. We first provide novel testimonies for categorizing deep SOD models from several distinct perspectives, including network architecture, level of supervision, *etc.* We then cover the contemporary literature on popular SOD datasets and evaluation criteria, providing a thorough performance benchmarking of major SOD methods and offering recommendations for several datasets and metrics that can be used to consistently assess different models. Next, we consider several previously under-explored issues related to benchmarking and baselines. In particular, we study the strengths and weaknesses of deep and non-deep SOD models by compiling and annotating a new dataset and evaluating several representative models on it, revealing promising directions for future efforts. We also study the robustness of SOD methods by analyzing the effects of various perturbations on the final performance. Moreover, for the first time in the field, we investigate the robustness of deep SOD models to maliciously designed adversarial perturbations and the transferability of these adversarial examples, providing baselines for future research. In addition, we analyze the generalization and difficulty of existing SOD datasets through a cross-dataset generalization study, and quantitatively reveal the dataset bias. We finally introduce several open issues and challenges of SOD in the deep learning era, providing insightful discussions and identifying a number of potentially fruitful directions forward.

In conclusion, SOD has achieved notable progress thanks to the striking development of deep learning techniques. However, there are still under-explored problems on achieving more efficient model designs, training, and inference for both academic research and real-world applications. We expect this survey to provide an effective way to understand current state-of-the-arts and, more importantly, insight for the future exploration of 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] 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.
- [12] G. Sun, W. Wang, J. Dai, and L. Van Gool, "Mining cross-image semantics for weakly supervised semantic segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 347–365.
- [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, Y. Yu, and K.-L. Ma, "Stereoscopic thumbnail creation via efficient stereo saliency detection," *IEEE Trans. Visualization and Comput. Graphics*, vol. 23, no. 8, pp. 2014–2027, 2016.
- [20] 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.
- [21] S. Avidan and A. Shamir, "Seam carving for content-aware image resizing," in *ACM Trans. Graph.*, vol. 26, no. 3, 2007, p. 10.
- [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] A. M. Treisman and G. Gelade, "A feature-integration theory of attention," *Cognitive psychology*, vol. 12, no. 1, pp. 97–136, 1980.
- [27] G. Li and Y. Yu, "Visual saliency based on multiscale deep features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 5455–5463.
- [28] 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.
- [29] 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.
- [30] 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.
- [31] 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.
- [32] 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.
- [33] 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.
- [34] 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.
- [35] Q. Yan, L. Xu, J. Shi, and J. Jia, "Hierarchical saliency detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 1155–1162.
- [36] 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.
- [37] 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.
- [38] 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.
- [39] 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.
- [40] 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.
- [41] 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.
- [42] 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.
- [43] 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.
- [44] 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.
- [45] T. V. Nguyen, Q. Zhao, and S. Yan, "Attentive systems: A survey," *Int. J. Comput. Vis.*, vol. 126, no. 1, pp. 86–110, 2018.

- [46] 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.
- [47] 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.
- [48] 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.
- [49] A. Borji, "Saliency prediction in the deep learning era: Successes and limitations," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2019.
- [50] A. Borji, M.-M. Cheng, Q. Hou, H. Jiang, and J. Li, "Salient object detection: A survey," *Computational Visual Media*, pp. 1–34, 2019.
- [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] Y. Wei, F. Wen, W. Zhu, and J. Sun, "Geodesic saliency using background priors," *Proc. Eur. Conf. Comput. Vis.*, pp. 29–42, 2012.
- [55] 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.
- [56] 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.
- [57] 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.
- [58] 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.
- [59] G. E. Hinton, A. Krizhevsky, and S. D. Wang, "Transforming auto-encoders," in *Proc. Int. Conf. Artificial Neural Netw.*, 2011, pp. 44–51.
- [60] 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.
- [61] 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.
- [62] 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.
- [63] 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.
- [64] 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.
- [65] 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.
- [66] 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.
- [67] 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.
- [68] 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.
- [69] 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.
- [70] 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.
- [71] 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.
- [72] 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.
- [73] 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.
- [74] 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.
- [75] 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.
- [76] 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.
- [77] S. He, J. Jiao, X. Zhang, G. Han, and R. W. Lau, "Delving into salient object subitzing and detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 1059–1067.
- [78] 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.
- [79] M. Amirul Islam, M. Kalash, and N. D. B. Bruce, "Revisiting salient object detection: Simultaneous detection, ranking, and subitzing of multiple salient objects," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018.
- [80] 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.
- [81] 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.
- [82] 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.
- [83] 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.
- [84] 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.
- [85] 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.
- [86] 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.
- [87] 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. 1171–11720.
- [88] 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.
- [89] 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.
- [90] X. Qin, Z. Zhang, C. Huang, C. Gao, M. Dehghan, and M. Jager sand, "Basnet: Boundary-aware salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 7479–7489.
- [91] 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.
- [92] 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.
- [93] 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.
- [94] 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.
- [95] 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.
- [96] 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.
- [97] 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.
- [98] G. Li, Y. Xie, and L. Lin, "Weakly supervised salient object detection using image labels," in *AAAI Conference on Artificial Intelligence*, 2018.
- [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] 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.
- [101] 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.
- [102] 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.
- [103] 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.
- [104] G. Li and Y. Yu, "Deep contrast learning for salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 478–487.
- [105] 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.
- [106] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected crfs with gaussian edge potentials," in *Proc. Advances Neural Inf. Process. Syst.*, 2011, pp. 109–117.
- [107] 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.
- [108] 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.
- [109] 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.
- [110] 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.
- [111] 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.
- [112] 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.
- [113] 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.
- [114] 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.
- [115] 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.
- [116] 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.
- [117] 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.
- [118] S. Xie and Z. Tu, "Holistically-nested edge detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1395–1403.
- [119] 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.
- [120] 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.
- [121] J. L. Long, N. Zhang, and T. Darrell, "Do convnets learn correspondence?" in *Proc. Advances Neural Inf. Process. Syst.*, 2014, pp. 1601–1609.
- [122] 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.
- [123] 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.
- [124] 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.
- [125] 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.
- [126] R. Caruana, "Multitask learning," *Machine learning*, vol. 28, no. 1, pp. 41–75, 1997.
- [127] 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.
- [128] 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.
- [129] 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.
- [130] 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.
- [131] 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.
- [132] R. Fan, Q. Hou, M.-M. Cheng, G. Yu, R. R. Martin, and S.-M. Hu, "Associating inter-image salient instances for weakly supervised semantic segmentation," in *The Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 367–383.
- [133] J. Zhao, J. Li, H. Liu, S. Yan, and J. Feng, "Fine-grained multi-human parsing," *Int. J. Comput. Vis.*, pp. 1–19, 2019.
- [134] 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.
- [135] 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.
- [136] 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.
- [137] 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.
- [138] 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.
- [139] 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.
- [140] 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.
- [141] 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.
- [142] C. Li, R. Cong, J. Hou, S. Zhang, Y. Qian, and S. Kwong, "Nested network with two-stream pyramid for salient object detection in

- optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 11, pp. 9156–9166, 2019.
- [143] I. Mahmood, M. Sajjad, W. Ejaz, and S. W. Baik, "Saliency-directed prioritization of visual data in wireless surveillance networks," *Information Fusion*, vol. 24, pp. 16–30, 2015.
- [144] Z. Zhang, S. Fidler, and R. Urtasun, "Instance-level segmentation for autonomous driving with deep densely connected mrf's," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 669–677.
- [145] C. Guo and L. Zhang, "A novel multiresolution spatiotemporal saliency detection model and its applications in image and video compression," *IEEE Trans. Image Process.*, vol. 19, no. 1, pp. 185–198, 2009.
- [146] 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.
- [147] A. Fawzi, S.-M. Moosavi-Dezfooli, and P. Frossard, "Robustness of classifiers: from adversarial to random noise," in *Proc. Advances Neural Inf. Process. Syst.*, 2016, pp. 1632–1640.
- [148] 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.
- [149] Y. Liu, X. Chen, C. Liu, and D. Song, "Delving into transferable adversarial examples and black-box attacks," in *Proc. Int. Conf. Learn. Representations*, 2017.
- [150] A. Torralba and A. A. Efros, "Unbiased look at dataset bias," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 1521–1528.
- [151] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Representations*, 2015.
- [152] B. Zoph and Q. V. Le, "Neural architecture search with reinforcement learning," in *Proc. Int. Conf. Learn. Representations*, 2017.
- [153] 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.
- [154] T.-W. Ke, J.-J. Hwang, Z. Liu, and S. X. Yu, "Adaptive affinity fields for semantic segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 587–602.
- [155] 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.
- [156] A. Veit and S. Belongie, "Convolutional networks with adaptive inference graphs," in *Proc. Eur. Conf. Comput. Vis.*, 2018.
- [157] 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.
- [158] M. Jiang, J. Xu, and Q. Zhao, "Saliency in crowd," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 17–32.
- [159] Q. Zheng, J. Jiao, Y. Cao, and R. W. Lau, "Task-driven webpage saliency," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 287–302.
- [160] A. Palazzi, F. Solera, S. Calderara, S. Alletto, and R. Cucchiara, "Learning where to attend like a human driver," in *IEEE Intelligent Vehicles Symposium*, 2017, pp. 920–925.
- [161] A. K. Mishra, Y. Aloimonos, L. F. Cheong, and A. Hassim, "Active visual segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 639–653, 2012.
- [162] 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.
- [163] 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.
- [164] L. Jing and Y. Tian, "Self-supervised visual feature learning with deep neural networks: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2020.
- [165] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2536–2544.
- [166] G. Larsson, M. Maire, and G. Shakhnarovich, "Colorization as a proxy task for visual understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6874–6883.
- [167] M. Caron, P. Bojanowski, A. Joulin, and M. Douze, "Deep clustering for unsupervised learning of visual features," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 132–149.
- [168] 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.
- [169] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," in *Proc. Advances Neural Inf. Process. Syst. - workshops*, 2014.
- [170] 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.
- [171] 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 a postdoc scholar at ETH Zurich, Switzerland. From 2016 to 2018, he was a visiting Ph.D. student in University of California, Los Angeles. From 2018 to 2019, he was a senior scientist at Inception Institute of Artificial Intelligence, UAE. His current research interests include computer vision 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 in 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 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, UAE. His research interests include computer vision and medical image analysis. He is an Associate Editor of IEEE TMI and IEEE Access.

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 *TPAMI*, *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.