

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

Wenguan Wang, Qiuxia Lai, Huazhu Fu, Jianbing Shen, Haibin Ling

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

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

## 1 INTRODUCTION

SLIENT object detection (SOD) aims at highlighting salient object regions in images. Different from fixation prediction (FP), which is originated from cognitive and psychology research communities, SOD is driven by and applied to a widely spectrum of object-level applications in various areas. In computer vision, sampled applications of SOD include image understanding [1], [2], image captioning [3]–[5], object detection [6], [7], un-supervised video object segmentation [8], [9], semantic segmentation [10]–[12], person re-identification [13], [14], etc. In computer graphics, SOD has been used tasks such as non-photo-realistic rendering [15], [16], automatic image cropping [17], image retargeting [18], [19], video summarization [20], [21], etc. Example applications in robotics, such as human-robot interaction [22], [23], and object discovery [24], [25] also benefit from SOD for scene understanding.

Significant improvement for SOD has been witnessed in recent years with the renaissance of deep learning techniques, thanks to the powerful representation learning methods. Since the first introduction in 2015 [26]–[28], deep

learning-based SOD (or *deep SOD*) algorithms have soon shown superior performance over traditional solutions, and kept residing the top of various benchmarking leaderboards. On the other hand, hundreds of research papers have been produced on deep SOD, making it non-trivial for effectively understanding the state-of-the-arts.

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

### 1.1 History and Scope

Compared with other computer vision tasks, the history of SOD is relatively short and can be traced back to the pioneer works in [29] and [30]. Most of **non-deep learning SOD models** [35], [48]–[50] are based on low-level features and rely on certain heuristics (e.g., *color contrast* [31], *background prior* [51]). For obtaining uniformly highlighted salient objects and clear object boundaries, an over-segmentation process that generates regions [34], super-pixels [52], [53], or object proposals [54] is often integrated into above models. Please see [41] for a comprehensive overview.

- W. Wang is with Inception Institute of Artificial Intelligence, UAE. (Email: [wenguanwang.ai@gmail.com](mailto:wenguanwang.ai@gmail.com))
- Q. Lai is with the Department of Computer Science and Engineering, the Chinese University of Hong Kong, Hong Kong. (Email: [qxlai@cse.cuhk.edu.hk](mailto:qxlai@cse.cuhk.edu.hk))
- H. Fu is with Inception Institute of Artificial Intelligence, UAE. (Email: [hzfu@ieee.org](mailto: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](mailto:shenjianbing@bit.edu.cn))
- H. Ling is with the Department of Computer and Information Sciences, Temple University, Philadelphia, PA, USA. (Email: [hbling@temple.edu](mailto:hbling@temple.edu))
- First two authors contribute equally.
- Corresponding author: Jianbing Shen

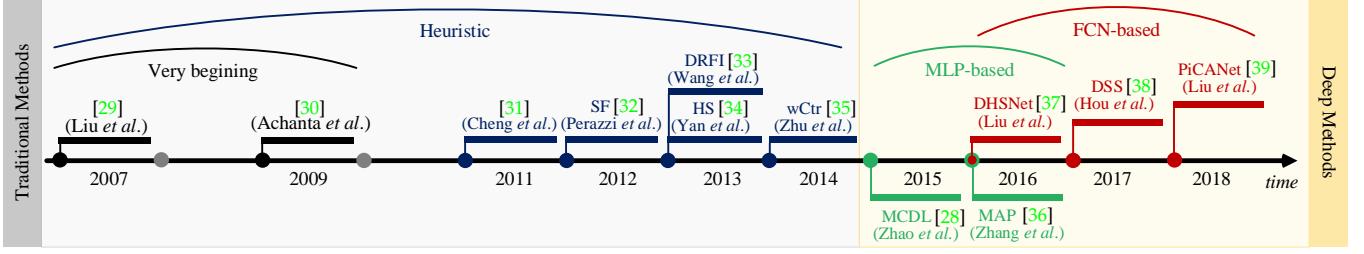


Fig. 1. A brief chronology of salient object detection (SOD). The very first SOD models date back to the work of Liu *et al.* [29] and Achanta *et al.* [30]. The first incorporation of deep learning techniques in SOD models is from 2015. See §1.1 for more detailed descriptions.

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

#	Title	Year	Venue	Description
1	State-of-the-Art in Visual Attention Modeling [40]	2013	TPAMI	A survey for visual attention ( <i>i.e.</i> fixation prediction) models before 2013.
2	Salient Object Detection: A Benchmark [41]	2015	TIP	This paper benchmarks 29 heuristic SOD models and 10 FP methods over 7 datasets.
3	Attentive Systems: A Survey [42]	2017	IJCV	A review of applications that utilize visual saliency cues.
4	Salient Object Detection: A Survey [43]	2018	arXiv	Reviews both heuristic and deep SOD models (65 and 21, respectively), and includes comparisons and discussions w.r.t. some closely related areas such as object detection, fixation prediction, segmentation <i>etc.</i>
5	A Review of Co-Saliency Detection Algorithms: Fundamentals, Applications, and Challenges [44]	2018	TIST	This paper reviews the fundamentals, challenges, and applications of co-saliency detection.
6	Review of Visual Saliency Detection with Comprehensive Information [45]	2018	TCSVT	This paper reviews RGB-D saliency detection, co-saliency detection and video saliency detection.
7	Advanced Deep-Learning Techniques for Salient and Category-Specific Object Detection: A survey [46]	2018	TSPM	This paper reviews several sub-directions of object detection, namely objectness detection, salient object detection and category-specific object detection.
8	Saliency Prediction in the Deep Learning Era: An Empirical Investigation [47]	2018	arXiv	A review of image and video fixation prediction models along with analysis of specific questions.

With the compelling success of deep learning technologies in computer vision, more and more **deep learning-based SOD methods** have been springing up since 2015. Earlier deep SOD models typically utilize multi-layer perceptron (MLP) classifiers to predict the saliency score of deep features extracted from each image processing unit [26]–[28]. Later, a more effective and efficient form, *i.e.*, fully convolutional network (FCN)-based network, becomes the mainstream of SOD architecture. Different deep models have different levels of supervision, and may use different learning paradigm during training. Specially, some SOD methods further distinguish individual instances among all the detected salient objects [36], [55]. A brief chronology is shown in Fig. 1.

**Scope of the survey.** Despite having a short history, research in deep SOD has produced hundreds of papers, making it impractical (and fortunately unnecessary) to review all of them. Instead, we carefully and thoroughly select influential or important papers published in, but not limited to, prestigious journals and conferences. This survey mainly focuses on the major progress in the last five years; but for completeness and better readability, some early related works are also included. It is worth noting that we restrict this survey to *single image object-level SOD* methods, and leave instance-level SOD, RGB-D saliency detection, co-saliency detection, video SOD, FP, social gaze prediction, *etc.*, as separate topics.

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

## 1.2 Related Previous Reviews and Surveys

Table 1 lists existing surveys that are closely related to our paper. Among these works, Borji *et al.* [41] comprehensively review SOD methods preceding 2015, thus does not refer to recent deep learning-based solutions. More recently, the review is extended in [43] that covers both traditional non-deep methods and recent deep ones, and discusses the relation w.r.t. several other closely-related research areas such as special-purpose object detection, fixation prediction and segmentation. Zhang *et al.* [44] review methods for co-segmentation, a branch of visual saliency that detects and segments common and salient foregrounds from more than one relevant images. Cong *et al.* [45] review several extended SOD tasks including RGB-D SOD, co-saliency detection and video SOD. Han *et al.* [46] look into the sub-directions of object detection, and conclude the recent progress in objectness detection, SOD, and category-specific object detection (COD). Borji *et al.* [40] and [47] summarize (both heuristic and deep) models for FP, another important branch of visual saliency, and analyze several special issues. Nguyen *et al.* [42] mainly focuses on categorizing the applications of visual saliency (including both SOD and FP) in different areas.

Different from previous SOD surveys, in this paper we systematically and comprehensively review *deep learning-based SOD* methods. Our survey is featured by in-depth analysis and discussion in various aspects, many of which, to the best of our knowledge, are the first time in this field. In particular, we summarize existing deep SOD methods based on several proposed taxonomies, gain deeper understanding of SOD models through attribute-based evaluation, discuss on the influence of input perturbation, analyze the robustness of deep SOD models w.r.t. adversarial attacks, study the generalization and hardness of existing

SOD datasets, and offer insights for essential open issues, challenges, and future directions. We expect our survey to provide novel insight and inspiration for facilitating the understanding of deep SOD, and to inspire research on the raised open issues such as the adversarial attacks to SOD.

### 1.3 Our Contributions

Our contributions in this paper are summarized as follows:

- 1) **Systematic review of deep SOD models from various perspectives.** We categorize and summarize existing deep SOD models according to network architecture, level of supervision, learning paradigm, etc. The proposed taxonomies aim to help researchers with deeper understanding of the key features of SOD in the deep learning era.
- 2) **A novel attribute-based performance evaluation of deep SOD models.** We compile a hybrid benchmark and provide annotated attributes considering object categories, scene categories and challenge factors. Based on the dataset, we evaluate the performance of six popular SOD models, and discuss how these attributes affect different algorithms and the improvements brought by deep learning techniques.
- 3) **Discussion regarding the influence of input perturbations.** We investigate the effects of various types of image perturbation on six representative SOD algorithms. The study is expected to provide informative suggestions regarding real-world applications where noises frequently appear.
- 4) **The first known adversarial attack analysis on SOD models.** DNNs have been shown to be surprisingly vulnerable to visually imperceptible adversarial attacks for typical tasks such as recognition, though how such attacks affect SOD models remains unexplored. We provide the first study on this issue with carefully designed baseline attacks and evaluations, which could serve as baselines for future study of the robustness and transferability of deep SOD models.
- 5) **Cross-dataset generalization study.** SOD datasets are often collected with certain bias [41], hence, we conduct a cross-dataset generalization study of existing SOD datasets with a representative baseline model.
- 6) **Overview of open issues and future directions.** We thoroughly look over several essential issues for model design, dataset collection, and the relation of SOD with other topics, which shed light on potential directions for future research.

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

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

## 2 DEEP LEARNING BASED SALIENT OBJECT DETECTION (SOD) MODELS

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

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

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

### 2.1 Representative Network Architectures for SOD

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

#### 2.1.1 Multi-Layer Perceptron (MLP)-based Methods

MLP-based methods typically extract deep features for each processing unit of an image to train an MLP-classifier for saliency score prediction, as shown in Fig. 2 (a). Commonly adopted processing units include *super-pixels/patches* [28], [56], [60], and generic *object proposals* [26], [27], [36], [68].

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

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

- **ELD** [60] concatenates deep convolution features and an *encoded low level distance map* (ELD-map) to construct a feature vector for each super-pixel. The ELD-map is generated from the initial hand-crafted feature distance maps of the queried super-pixel using CNN.

- **SuperCNN** [56] constructs two hand-crafted input feature sequences for each super-pixel, which are further processed

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

	#	Methods	Publ.	Architecture	Backbone	Level of Supervision	Learning Paradigm	Obj.-/Inst.-Level SOD	Training Dataset	#Training	CRF
2015	1	SuperCNN [56]	IJCV	MLP+super-pixel	GoogleNet [57]	Fully-Sup.	STL	Object	ECSSD [34]	800	
	2	MCDL [28]	CVPR	MLP+super-pixel		Fully-Sup.	STL	Object	MSRA10K [58]	8,000	
	3	LEGS [27]	CVPR	MLP+segment		Fully-Sup.	STL	Object	MSRA-B [29]+PASCAL-S [59]	3,000+340	
	4	MDF [26]	CVPR	MLP+segment		Fully-Sup.	STL	Object	MSRA-B [29]	2,500	
2016	1	ELD [60]	CVPR	MLP+super-pixel	VGGNet	Fully-Sup.	STL	Object	MSRA10K [58]	~9,000	
	2	DHSNet [37]	CVPR	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA10K [58]+DUT-OMRON [52]	6,000+3,500	
	3	DCL [61]	CVPR	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA-B [29]	2,500	✓
	4	RACDNN [62]	CVPR	FCN	VGGNet	Fully-Sup.	STL	Object	DUT-OMRON [52]+NJU2000 [63]+RGBD [64]	10,565	
	5	SU [65]	CVPR	FCN	VGGNet	Fully-Sup.	MTL	Object	MSRA10K [58]+SALICON [66]	10,000+15,000	✓
	6	MAP [36]	CVPR	MLP+obj. prop.	VGGNet	Fully-Sup.	MTL	Object	SOS [67]	~5,500	
	7	SSD [68]	ECCV	MLP+obj. prop.	AlexNet	Fully-Sup.	STL	Object	MSRA-B [29]	2,500	
	8	CRPSD [69]	ECCV	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA10K [58]	10,000	
	9	RFCN [70]	ECCV	FCN	VGGNet	Fully-Sup.	MTL	Object	PASCAL VOC 2010 [71]+MSRA10K [58]	10,103+10,000	
	10	MDS [72]	TIP	FCN	VGGNet	Fully-Sup.	MTL	Object	MSRA10K [58]	10,000	
2017	1	MSRNet [55]	CVPR	FCN	VGGNet	Fully-Sup.	STL	Instance	MSRA-B [29]+HKU-IS [26] (+ILSO [55])	2,500+2,500 (+500)	✓
	2	DSS [38]	CVPR	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA-B [29]+HKU-IS [26]	2,500	✓
	3	WSS [73]	CVPR	FCN	VGGNet	Weakly-Sup.	MTL	Object	ImageNet [74]	456k	✓
	4	DLS [75]	CVPR	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA10K [58]	10,000	
	5	NLDF [76]	CVPR	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA-B [29]	2,500	
	6	DSOS [77]	ICCV	FCN	VGGNet	Fully-Sup.	MTL	Object	SOS [67]	6,900	
	7	Amulet [78]	ICCV	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA10K [58]	10,000	
	8	FSN [79]	ICCV	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA10K [58]	10,000	
	9	SBF [80]	ICCV	FCN	VGGNet	Un-Sup.	STL	Object	MSRA10K [58]	10,000	
	10	SRM [81]	ICCV	FCN	ResNet	Fully-Sup.	STL	Object	DUTS [73]	10,553	
	11	UCF [82]	ICCV	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA10K [58]	10,000	
2018	1	RADF [83]	AAAI	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA10K [58]	10,000	✓
	2	ASMO [84]	AAAI	FCN	ResNet101	Weakly-Sup.	MTL	Object	Microsoft COCO+MSRA-B [29]+HKU-IS [26]	82,783+2,500+2,500	✓
	3	LICNN [85]	AAAI	FCN	VGGNet	Weakly-Sup.	STL	Object	ImageNet [74]	456k	✓
	4	BDMP [86]	CVPR	FCN	VGGNet	Fully-Sup.	STL	Object	DUTS [73]	10,553	
	5	DUS [87]	CVPR	FCN	ResNet101	Un-Sup.	MTL	Object	MSRA-B [29]	2,500	
	6	DGRL [88]	CVPR	FCN	ResNet50	Fully-Sup.	STL	Object	DUTS [73]	10,553	
	7	PAGR [89]	CVPR	FCN	VGGNet19	Fully-Sup.	STL	Object	DUTS [73]	10,553	
	8	RSDNet [90]	CVPR	FCN	ResNet101	Fully-Sup.	MTL	Object	PASCAL-S [59]	425	
	9	ASNNet [91]	CVPR	FCN	VGGNet	Fully-Sup.	MTL	Object	SALICON [66]+MSRA10K [58]+DUT-OMRON [52]	15,000+10,000+5,168	
	10	PiCANet [39]	CVPR	FCN	VGGNet/ResNet50	Fully-Sup.	STL	Object	DUTS [73]	10,553	✓
	11	C2S-Net [92]	ECCV	FCN	VGGNet	Weakly-Sup.	MTL	Object	MSRA10K [58]+Web	10,000+20,000	
	12	RAS [93]	ECCV	FCN	VGGNet	Fully-Sup.	STL	Object	MSRA-B [29]	2,500	

by two CNN columns separately to produce binary scores using 1D convolution instead of fully connected layers.

**2) Object proposal-based methods** leverage object proposals [26], [27] or bounding-boxes [36], [68] as basic processing units that naturally encode object information.

- **LEGS** [27] constructs segment-level feature vectors out of pixel-level deep features, it then uses an MLP to predict saliency scores from the segment-level features. The final saliency map is the weighted sum over all segment masks.
- **MDF** [26] constructs feature vectors for each image segment by feeding three nested rectangle regions into a pre-trained image classification DNN. An MLP is trained to regress the segment-level saliency. The final saliency map is the linear combination of three resulted saliency maps.
- **MAP** [36] uses a CNN model to generate a set of scored bounding boxes, then selects an optimized compact subset of bounding boxes for multiple salient objects.
- **SSD** [68] first generates region proposals and then uses a CNN to classify each proposal into a pre-defined shape class with standard binary map. The final saliency map is averaged over the binary maps of all the proposals.

### 2.1.2 Fully Convolutional Network (FCN)-based Methods

Though having outperformed previous non-deep learning SOD models and heuristic ones with deeply learned features, the MLP-based SOD models can not capture well critical spatial information and are time-consuming as they need to process all visual sub-units one by one. Inspired by the great success of Fully Convolutional Network (FCN) [94] in semantic segmentation, latest deep SOD solutions adapt popular classification models, *e.g.*, VGGNet [95] and ResNet [96], into fully convolutional ones to directly

output spatial maps instead of classification scores. This way, these deep SOD solutions benefit from end-to-end spatial saliency representation learning and efficiently predict saliency maps in a single feed-forward process. Typical architectures can be divided into five categories: *Single-stream network*, *Multi-stream network*, *Side-fusion network*, *Bottom-up/top-down network*, and *Branched network*.

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

- **RFCN** [70] recurrently refines the saliency prediction based on the input image and the saliency priors from heuristic calculation or prediction of previous time step. It can be viewed as a cascaded structure after being unrolled.
- **RACDNN** [62] produces a coarse saliency map using an encoder-decoder stream, and progressively refines different local object regions. It utilizes a spatial transformer [97] to attend to an image region at each iteration for refinement.
- **DLS** [75] utilizes a stack of convolution and dilated convolution layers to produce an initial saliency map, and then refines it at super-pixel level. A level set loss function is used to aid the learning of the binary segmentation map.
- **UCF** [82] uses an encoder-decoder architecture to produce finer-resolution predictions. It learns uncertainty through a reformulated dropout in the decoder, and avoids artifacts by using a hybrid up-sampling scheme in the decoder.
- **DUS** [87] is based on the Deeplab [98] algorithm, which is an FCN with dilated convolution layers on the top. It learns the latent saliency and noise pattern by pixel-wise supervision from several heuristic saliency methods.
- **LICNN** [85] generates ‘post-hoc’ saliency maps by combining top-5 category-specific attention maps of a pre-

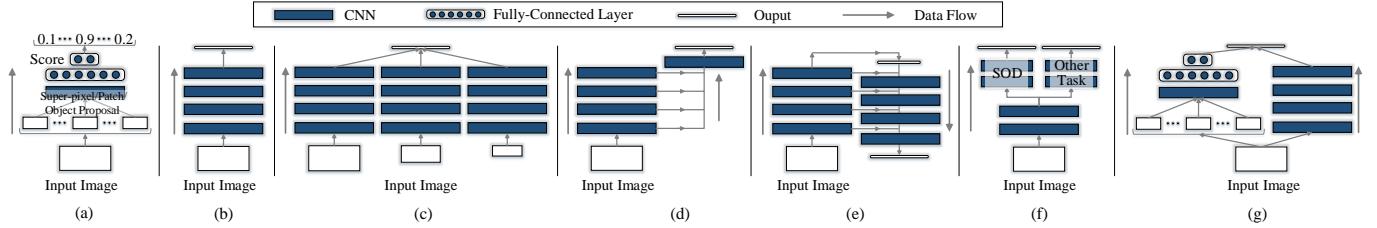


Fig. 2. Category of previous deep SOD models. (a) MLP-based methods; (b)-(f) FCN-based methods, mainly using (b) single-stream network, (c) multi-stream network, (d) side-out fusion network, (e) bottom-up/top-down network, and (f) branch network architectures. (g) Hybrid network-based methods. See §2.1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

trained image classification network. The lateral inhibition enhances the discriminative ability of the attention maps, releasing it from the need of SOD annotations.

**2) Multi-stream network**, as depicted in Fig. 2 (c), typically has multiple network streams, each of which is trained with an input at a particular resolution to explicitly learn multi-scale saliency features. The outputs from different streams are then combined together for the final prediction.

- **MSRNet** [55] consists of three streams of bottom-up/top-down network structure to process three scaled versions of the input image. The three outputs are finally fused through a learnable attention module.

- **SRM** [81] progressively refines saliency features by passing them stage-wisely from a coarser stream to a finer one. The top-most feature of each stream is supervised with the ground-truth saliency mask. The pyramid pooling module further facilitates multi-stage saliency fusion and refinement.

- **FSN** [79], inspired by the observation that salient objects typically gain most of human eye-fixations [59], fuses the outputs of a fixation stream [99] and a semantic stream [95] into an inception-segmentation module to predict saliency.

**3) Side-fusion network** fuses multi-layer responses of a backbone network together for SOD prediction, making use of the inherent multi-scale representations of the CNN hierarchy (Fig. 2 (d)). Side-outputs are typically supervised by the ground-truth, leading to a *deep supervision* strategy [100].

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

- **NLDF** [76] generates a local saliency map by fusing multi-level features and contrast features in a top-down manner, then integrates the local map with a global one yielded by the top layer to produce the final prediction. The contrast features are obtained by subtracting the feature from its average pooling.

- **Amulet** [78] aggregates multi-level features into multiple resolutions. The multiple aggregated features are further refined in a top-down manner. A boundary refinement is introduced at each aggregated feature before final fusion.

- **DSOS** [77] uses two subnets for detecting salient objects and subtitizing the result, respectively. The detection subnet is a U-net structure [101] with side-fusions, whose bottleneck parameters are dynamically determined by the other subnet.

- **RADF** [83] utilizes the integrated side-features to refine themselves, and such process is repeated to gradually yield

finer saliency predictions.

- **RSDNet-R** [90] combines an initial coarse representation with finer features at earlier layers under a gating mechanism to stage-wisely refine the side-outputs. Maps from all the stages are fused to obtain the overall saliency map.

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

- **DHSNet** [37] refines the coarse saliency map by gradually combining shallower features using recurrent layers, where all the intermediate maps are supervised by the ground truth saliency maps [100].

- **SBF** [80] borrows the network architecture of DHSNet [37], but is trained under the weak ground truth provided by several un-supervised heuristic SOD methods.

- **BDMP** [86] refines multi-level features using convolution layers with various reception fields, and enables inter-level exchange through a gated bi-directional pathway. The refined features are fused in a top-down manner.

- **RLN** [88] uses an inception-like module to purify the low-level features. A recurrent mechanism in the top-down pathway further refines the combined features. The saliency output is enhanced by a boundary refinement network.

- **PAGR** [89] enhances the learning ability of the feature extraction pathway by incorporating multi-path recurrent connections to transfer higher-level semantics to lower layers. The top-down pathway is embedded with several channel-spatial attention modules for refining the features.

- **ASNet** [91] learns a coarse fixation map in the feed-forward pass, then utilizes a stack of convLSTMs [102] to iteratively infer pixel-wise salient object segmentation by incorporating multi-level features from successively shallower layers.

- **PiCANet** [39] hierarchically embeds global and local pixel-wise contextual attention modules into the top-down pathway of a U-Net [101] structure.

- **RAS** [93] embeds *reverse attention* (RA) blocks in the top-down pathway to guide residual saliency learning. The RA blocks emphasize the non-object areas using the complement of deeper-level output.

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

- **SU** [65] performs eye-fixation prediction (FP) and SOD in a branched network. The shared layers capture the semantics and global saliency contexts. The FP branch learns to

infer fixations from the top feature, while the SOD branch aggregates side-features to better preserve spatial cues.

- **DS** [72] consists of an SOD branch and a semantic segmentation branch sharing the bottom layers for extracting the semantically rich features. Each branch consists of a sequence of convolution and deconvolution layers to produce pixel-wise prediction.

- **WSS** [73] consists of an image classification branch and an SOD branch. The SOD branch benefits from the features trained under image-level supervision, and produces initial saliency maps in a top-down scheme which are further refined by an iterative *conditional random field* (CRF) and used for fine-tuning the SOD branch.

- **ASMO** [84] performs the same tasks with WSS [73] and is also trained under weak supervision. The main difference is that the shared network in ASMO uses a multi-stream structure to handle different scales of an input image.

- **C2S-Net** [92] is constructed by adding an SOD branch to a pre-trained contour detection model, *i.e.*, CEND [103]. The two branches are trained under an alternating scheme with the supervision signals provided by each other.

### 2.1.3 Hybrid Network-based Methods

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

- **DCL** [61] generates the saliency map by combining a pixel-wise prediction of a side-fusion FCN stream and a segment-level map produced by making binary classification for multi-scale super-pixels based on deep features. The two branches share the same feature extraction network, and are alternatively optimized during training.

- **CRPSD** [69] also combines pixel-level and super-pixel-level saliency. The former is generated by fusing the last and penultimate side-output features of an FCN, while the latter is obtained by applying MCDL [28] to adaptively generated regions. Only the FCN and the fusion layers are trainable.

## 2.2 Level of Supervision

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

### 2.2.1 Fully-Supervised Methods

Most deep SOD models are trained with large-scale pixel-wise human annotations. The success of these fully-supervised methods benefits a lot from a huge number of manually annotated data. However, for an SOD task, obtaining large-scale pixel-wise saliency annotations is time-consuming and requires heavy and intensive human labeling. Moreover, models trained on fine-labeled datasets tend to overfit and usually generalize poorly to real-life images. Thus, how to train SOD with less human annotations becomes an increasingly popular research direction.

### 2.2.2 Un-/Weakly-Supervised Methods

Un-/Weakly supervised learning refers to learning without task-specific ground-truth supervision. To get rid of the laborious manual labeling, some SOD methods make efforts to predict saliency using *image-level* categorical labels [73],

[85], or pseudo *pixel-wise* saliency annotations generated by heuristic un-supervised SOD methods [80], [84], [87] or from other applications [92]. Experiments show comparable performance of these methods with the state-of-the-art.

**1) Category-level supervision.** It has been shown that the hierarchical deep features trained with image-level labels have the ability to locate the regions containing objects [104], [105], which is promising to provide useful cues for detecting salient objects in the scene. Thus, current large-scale image classification datasets can also be used for training deep SOD models to localize salient objects.

- **WSS** [73] first pre-trains a two-branch network to predict image labels at one branch using ImageNet [74], while estimating saliency maps at the other. The estimated maps are refined by CRF and used to fine-tune the SOD branch.

- **LICNN** [85] turns to an ImageNet-pretrained image classification network to generate ‘post-hoc’ saliency maps. It does not need explicit training with any other SOD annotations thanks to the lateral inhibition mechanism.

**2) Pseudo pixel-level supervision.** Though being informative, image-level labels are too sparse to yield precise pixel-wise saliency segmentation. Some researchers propose to utilize traditional un-supervised SOD methods [80], [84], [87] or contour information [92] to automatically generate noisy saliency maps, which are progressively refined and used to provide finer pixel-level supervision for training a more effective deep SOD model.

- **SBF** [80] generates saliency predictions through a fusion process that integrates the weak saliency maps yielded by several classical un-supervised salient object detectors [34], [106], [107] at intra- and inter-image levels.

- **ASMO** [84] trains a multi-task FCN with image categorical labels and noisy maps of heuristic un-supervised SOD methods. The coarse saliency and the average map of the top-3 class activation maps [105] are fed into a CRF model to obtain finer maps for fine-tuning the SOD sub-net.

- **DUS** [87] jointly learns latent saliency and noise patterns from noisy saliency maps generated by several traditional un-supervised SOD methods [34], [35], [108], [109], and produces finer saliency maps for next training iteration.

- **C2S-Net** [92] generates pixel-wise saliency masks from contours [110] using CEDN [103] and trains the SOD branch. The contour and SOD branches alternatively update each other and progressively output finer SOD predictions.

## 2.3 Learning Paradigm

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

### 2.3.1 Single-Task Learning (*STL*) based Methods

In machine learning, the standard methodology is to learn one task at a time, *i.e.*, single-task learning [111]. Most deep SOD methods belong to this realm of learning paradigm. They utilize supervision from a single knowledge domain to train the SOD models, using either the SOD domain, or other related domains such as image classification [85].

### 2.3.2 Multi-Task Learning (MTL) based Methods

Inspired by human learning process where the knowledge learned from related tasks can be used to help learning a new task, Multi-Task Learning (MTL) [111] aims to learn multiple related tasks simultaneously. By incorporating domain-specific information from extra training signals of related tasks, the generalization ability of the model gets improved. The sharing of samples among tasks also alleviates the lack of data for training heavy-parameterized models such as those in deep learning, especially under un-/weakly-supervised learning paradigm where the task-related annotations are limited.

Some MTL-based SOD methods train different tasks on the same architecture in tandem [55], [90]; some learn multi-domain knowledge simultaneously by incorporating different objective terms into the loss function [87], [91], [112]–[114]; while some utilize a branched network structure in which the bottom layers are shared while the top layers are task-specific [65], [72], [73].

Current MTL-based SOD models are typically trained with tasks such as salient object subitizing [36], [77], [90], fixation prediction [65], [91], image classification [73], noise pattern learning [87], semantic segmentation [70], [72], and contour detection [55]. The learning of collaborative feature representations improves the generalization abilities as well as the performances for both tasks.

**1) Salient object subitizing** [67]. The ability of human to rapidly enumerate a small number of items is referred to as subitizing [115]. Some SOD methods learn salient object subitizing and detection simultaneously.

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

- **DSOS** [77] uses an auxiliary network to learn salient object subitizing, which affects the SOD subnet by alternating the parameters of its adaptive weight layer.

- **RSDNet** [90], different from above methods that explicitly model salient object subitizing as a classification problem, applies a stack of saliency-level-aware ground-truth masks to train the network that implicitly learns to figure out the number of salient objects as well as their relative saliency.

**2) Fixation prediction** aims to predict the locations of human eye-fixations that reflect the attention distribution. Due to its close relation with SOD, learning shared knowledge from the two closely related tasks is promising to improve the performances of both.

- **SU** [65] performs eye-fixation prediction and SOD in a branched network. The shared layers learn to capture the semantics and global saliency contexts. The branched layers are distinctively trained to handle task-specific problems.

- **ASNet** [91] learns SOD by jointly training a bottom-up pathway to derive fixation maps. A top-down pathway progressively refines the object-level saliency estimation by incorporating multi-level features under the guidance of the biologically-related, visual-fixation knowledge.

**3) Image classification.** The image category labels can help localize the discriminative regions [104], [105], [116], which

often contain salient object candidates. Some methods thus leverage image-category classification to assist SOD task.

- **WSS** [73] learns a *foreground inference network* (FIN) for predicting image categories as well as estimating foreground maps for all categories. FIN is further fine-tuned to predict saliency map through several deconvolution layers under the supervision of CRF-refined foreground maps.

- **ASMO** [84] learns to predict the saliency map and the image categories simultaneously under the supervision of category labels and pseudo ground-truth saliency maps from traditional un-supervised SOD methods.

**4) Noise pattern modeling** learns the noise pattern out of the noisy saliency maps generated by existing heuristic un-supervised SOD methods, aiming at extracting ‘pure’ saliency maps for supervising SOD training.

- **DUS** [87] proposes to model the noise pattern of the noisy supervision from traditional un-supervised SOD methods instead of denoising. The SOD and noise pattern modeling tasks are jointly optimized under a unified loss.

**5) Semantic segmentation** is to assign each image pixel a label from a set of predefined categories. SOD can be viewed as a class-agnostic semantic segmentation where each pixel is classified as either belongs to a salient object or not. High-level semantics play an important role in distinguishing salient objects from backgrounds in situations where the two have similar visual appearance.

- **RFCN** [70] is first pre-trained on the PASCAL VOC<sub>2010</sub> segmentation dataset [71] to learn semantic information, and then fine-tuned on an SOD dataset to predict foreground and background maps. The saliency map is a softmax combination of the foreground and background scores.

- **DS** [72] carries out SOD and semantic segmentation in a branched network, where the shared layers learn collaborative feature representations. During training, one branch gets updated with the other fixed at each training iteration.

**6) Contour detection** responds to edges belonging to objects without considering background boundaries. Though seem inherently different, contours can provide useful priors for identifying salient regions in the image.

- **C2S-Net** [92] encodes the common features of contour detection and SOD at shared bottom layers, and performs the two tasks at distinct branches. Through alternative training, the contour branch is gradually fine-tuned to detect saliency-aware contours, meanwhile the saliency branch learns to predict the salient object masks from scratch.

## 2.4 Object-/Instance-Level SOD

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

### 2.4.1 Object-Level Methods

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

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

#	Dataset	Year	Publ.	#Img.	#Obj.	Obj. Area(%)	SOD Annotation	Resolution	Fix.
Early	MSRA-A [29]	2007	CVPR	1,000/20,840	1-2	-	bounding-box object-level	$\max(w, h) = 400$	
	MSRA-B [29]	2007	CVPR	5,000	1-2	$20.82 \pm 10.29$	bounding-box object-level, pixel-wise object-level	$\max(w, h) = 400, \min(w, h) = 126$	
	SED1 [117]	2007	CVPR	100	1	$26.70 \pm 14.26$	pixel-wise object-level	$\max(w, h) = 465, \min(w, h) = 125$	
	SED2 [117]	2007	CVPR	100	2	$21.42 \pm 18.41$	pixel-wise object-level	$\max(w, h) = 300, \min(w, h) = 144$	
	ASD [30]	2009	CVPR	1,000	1-2	$19.89 \pm 9.53$	pixel-wise object-level	$\max(w, h) = 400, \min(w, h) = 142$	
Modern&Popular	SOD [118]	2010	CVPR-W	300	1-4+	$27.99 \pm 19.36$	pixel-wise object-level	$\max(w, h) = 481, \min(w, h) = 321$	
	MSRA10K [58]	2011	CVPR	10,000	1-2	$22.21 \pm 10.09$	pixel-wise object-level	$\max(w, h) = 400, \min(w, h) = 144$	
	ECSSD [34]	2013	CVPR	1,000	1-4+	$23.51 \pm 14.02$	pixel-wise object-level	$\max(w, h) = 400, \min(w, h) = 139$	
	DUT-OMRON [52]	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 [59]	2014	CVPR	850	1-4+	$24.23 \pm 16.70$	pixel-wise object-level	$\max(w, h) = 500, \min(w, h) = 139$	✓
	HKU-IS [26]	2015	CVPR	4,447	1-4+	$19.13 \pm 10.90$	pixel-wise object-level	$\max(w, h) = 500, \min(w, h) = 100$	
	DUTS [73]	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 [67]	2015	CVPR	6,900	0-4+	$41.22 \pm 25.35$	number, bounding-box (train set)	$\max(w, h) = 6132, \min(w, h) = 80$	
	MSO [67]	2015	CVPR	1,224	0-4+	$39.51 \pm 24.85$	number, bounding-box instance-level	$\max(w, h) = 3888, \min(w, h) = 120$	
	ILSO [55]	2017	CVPR	1,000	1-4+	$24.89 \pm 12.59$	pixel-wise instance-level	$\max(w, h) = 400, \min(w, h) = 142$	
	XPIE [119]	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 [120]	2018	ECCV	6,000	0-4+	$21.36 \pm 16.88$	pixel-wise instance-level, object category, attribute	$\max(w, h) = 849, \min(w, h) = 161$	✓

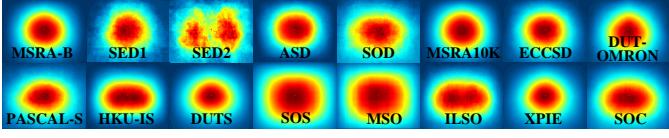


Fig. 3. Ground-truth annotation distributions of representative SOD datasets. See §3 for more detailed descriptions.

#### 2.4.2 Instance-Level Methods

Instance-level SOD methods produce saliency masks with distinct object labels, which perform more detailed parsing of the detected salient regions. The instance-level information is crucial for many practical applications where finer distinctions are needed.

- **MAP** [36] emphasizes instance-level SOD in unconstrained images. It first generates numerous object candidates, and then selects the top-ranking ones as the outputs.
- **MSRNet** [55] decomposes salient instance detection into three sub-tasks, *i.e.*, pixel-level saliency prediction, salient object contour detection and salient instance identification.

### 3 SOD DATASETS

With the rapid development of SOD, numerous datasets have been introduced, which play an important role in both SOD model training and performance benchmarking.

Table 3 summarizes 17 representative datasets. Early SOD datasets collect images with typically one salient object each, and provide bounding box annotations that were thought to be insufficient for reliable evaluations [30], [121]. Later, large-scale datasets with pixel-wise masks were brought out, with images containing very limited number of objects and simple backgrounds. Recently, datasets with multiple salient objects per image in complex or cluttered scenes are collected. In particular, some datasets provide extra annotations like numerical or instance-level information, facilitating other related tasks or applications. Fig. 3 shows the annotation distribution of 16 available datasets.

#### 3.1 Early SOD Datasets

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

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

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

- **SED** [117]<sup>1</sup> comprises of a single-object subset SED1 and a two-object subset SED2, each of which contains 100 images and has pixel-wise annotations. The objects in the images differ from their surroundings by various low-level cues such as intensity, texture, *etc*. Each image was segmented by three subjects. A pixel is considered as foreground if at least two subjects agreed.

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

#### 3.2 Modern Popular SOD Datasets

Recently emerged datasets tend to include more challenging and general scenes with relatively complex backgrounds and contain multiple salient objects. In this section, we review seven most popular and widely-used ones. Their popularity can be roughly attributed to the high difficulty and improved annotation quality.

- **SOD** [118]<sup>3</sup> contains 300 images from the Berkeley segmentation dataset [122]. Each image is labeled by seven subjects. Many images have more than one salient objects that have low color contrast to the background or touch image boundaries. Pixel-wise annotations are available.

- **MSRA10K** [58]<sup>4</sup>, also known as THUS10K, contains 10,000 images selected from MSRA [29] and covers all the 1,000

1. [http://www.wisdom.weizmann.ac.il/~vision/Seg\\_Evaluation\\_DB/dl.html](http://www.wisdom.weizmann.ac.il/~vision/Seg_Evaluation_DB/dl.html)

2. [https://ivrlwww.epfl.ch/supplementary\\_material/RK\\_CVPR09/](https://ivrlwww.epfl.ch/supplementary_material/RK_CVPR09/)

3. <http://elderlab.yorku.ca/SOD/>

4. <https://mmcheng.net/zh/msra10k/>

images in ASD [30]. The images have consistent bounding box labeling, and are further augmented with pixel-level annotations. Due to its large scale and precise annotations, it is widely used to train deep SOD models (see Table 2).

- ECSSD [34]<sup>5</sup> is composed by 1,000 images with semantically meaningful but structurally complex natural contents. The ground-truth masks are annotated by 5 participants.
- DUT-OMRON [52]<sup>6</sup> contains 5,168 images of relatively complex backgrounds and high content variety. Each image is accompanied with pixel-level ground-truth annotation.
- PASCAL-S [59]<sup>7</sup> consists of 850 challenging images selected from the *val* set of PASCAL VOC 2010 [71]. In addition to eye-fixation records, roughly pixel-wise and non-binary salient-object annotations are provided.
- HKU-IS [26]<sup>8</sup> contains 4,447 complex scenes that typically contain multiple disconnected objects with relatively diverse spatial distribution, *i.e.*, at least one salient object touches the image boundary. Besides, the similar fore/back-ground appearance makes this dataset more difficult.
- DUTS [73]<sup>9</sup>, the largest SOD dataset, contains 10,553 training and 5,019 test images. The training images are selected from the ImageNet DET *train/val* set [123], and the test images from the ImageNet *test* set [123] and the SUN dataset [124]. Since 2017, many deep SOD models are trained on the training set of DUTS (see Table 2).

### 3.3 Other Special SOD Datasets

Beyond the “standard” SOD datasets mentioned above, there are some special ones proposed recently, which are useful to capture different aspects in SOD and lead to related research directions. For example, some of them built datasets with instance-level annotations; some include images with *no* salient objects; *etc.*

- SOS [67]<sup>10</sup> is created for SOD subitizing [115], *i.e.*, to predict the number of salient objects without an expensive detection process. It contains 6,900 images selected from [123]–[126]. Each image is labeled as containing 0, 1, 2, 3 or 4+ salient objects. SOS is randomly split into a *training* (5,520 images) and a *test* set (1,380 images).
- MSO [67]<sup>11</sup> is a subset of the *test* set of SOS and contains 1,224 images. It has a more balanced distribution regarding the number of salient objects, and each object is annotated with a bounding box.
- ILSO [55]<sup>12</sup> has 1,000 images with pixel-wise instance-level saliency annotations and coarse contour labeling, where the benchmark results are generated using MSR-Net [55]. Most of the images in ILSO are selected from [26], [34], [52], [67] to reduce ambiguity over the salient object regions.
- XPIE [119]<sup>13</sup> contains 10,000 images with unambiguous salient objects, which are annotated with pixel-wise ground-truths. It covers scenes varied from simple and complex,

5. <http://www.cse.cuhk.edu.hk/leojia/projects/hsaliency>

6. <http://saliencydetection.net/dut-omron/>

7. <http://cbi.gatech.edu/salobj/>

8. [https://i.cs.hku.hk/~gbli/deep\\_saliency.html](https://i.cs.hku.hk/~gbli/deep_saliency.html)

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>

and contains salient objects of different numbers, sizes and positions. It has three subsets: *Set-P* contains 625 images of places-of-interest with geographic information; *Set-I* contains 8,799 images with object tags; and *Set-E* includes 576 images with eye-fixation annotations.

- SOC [120]<sup>14</sup> has 6,000 images with 80 common categories. Half of the images contain salient objects and the others contain none. Each salient-object-contained image is annotated with instance-level SOD ground-truth, object category (*e.g.*, dog, book), and challenging factors (*e.g.*, big/small object). The non-salient object subset has 783 texture images and 2,217 real-scene images (*e.g.*, aurora, sky).

## 4 EVALUATION METRICS

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

- **Precision-Recall (PR)** is calculated based on the binarized saliency mask and the 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. To get the binary mask, a set of thresholds ranging from 0 to 255 is applied, each of which produces a pair of Precision/Recall value to form a PR curve for describing model performance.

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

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

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

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

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

- **Weighted  $F_\beta$  measure (Fbw)** [127] intuitively generalizes F-measure by alternating the way to calculate the Precision and Recall. It extends the four basic quantities TP, TN, FP and FN to real values, and assigns different weights ( $\omega$ ) to different errors at different locations considering the neighborhood information, defined as:

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

14. <http://mmcheng.net/SOCBenchmark/>

TABLE 4

Benchmarking results of 29 state-of-the-art deep SOD models and 3 top-performing classic SOD methods on 6 famous datasets (See §5.1).

Dataset	ECSSD [34]			DUT-OMRON [52]			PASCAL-S [59]			HKU-IS [26]			DUTS-test [73]			SOD [118]			
	Metric	max $\mathcal{F} \uparrow$	$\mathcal{S} \uparrow$	MAE $\downarrow$	max $\mathcal{F} \uparrow$	$\mathcal{S} \uparrow$	MAE $\downarrow$	max $\mathcal{F} \uparrow$	$\mathcal{S} \uparrow$	MAE $\downarrow$	max $\mathcal{F} \uparrow$	$\mathcal{S} \uparrow$	MAE $\downarrow$	max $\mathcal{F} \uparrow$	$\mathcal{S} \uparrow$	MAE $\downarrow$	max $\mathcal{F} \uparrow$	$\mathcal{S} \uparrow$	MAE $\downarrow$
2013-14	*HS [34]	.673	.685	.228	.561	.633	.227	.569	.624	.262	.652	.674	.215	.504	.601	.243	.756	.711	.222
	*DRFI [48]	.751	.732	.170	.623	.696	.150	.639	.658	.207	.745	.740	.145	.600	.676	.155	.658	.619	.228
	*wCtr [35]	.684	.714	.165	.541	.653	.171	.599	.656	.196	.695	.729	.138	.522	.639	.176	.615	.638	.213
2015	MCDL [28]	.816	.803	.101	.670	.752	.089	.706	.721	.143	.787	.786	.092	.634	.713	.105	.689	.651	.182
	LEGS [27]	.805	.786	.118	.631	.714	.133	‡	‡	‡	.736	.742	.119	.612	.696	.137	.685	.658	.197
	MDF [26]	.797	.776	.105	.643	.721	.092	.704	.696	.142	.839	.810	.129	.657	.728	.114	.736	.674	.160
2016	ELD [60]	.849	.841	.078	.677	.751	.091	.782	.799	.111	.868	.868	.063	.697	.754	.092	.717	.705	.155
	DHSNet [37]	.893	.884	.060	‡	‡	‡	.799	.810	.092	.875	.870	.053	.776	.818	.067	.790	.749	.129
	DCL [61]	.882	.868	.075	.699	.771	.086	.787	.796	.113	.885	.877	.055	.742	.796	.149	.786	.747	.195
2017	°MAP [36]	.556	.611	.213	.448	.598	.159	.521	.593	.207	.552	.624	.182	.453	.583	.181	.509	.557	.236
	CRPSD [69]	.915	.895	.048	-	-	-	.864	.852	.064	.906	.885	.043	-	-	-	-	-	-
	RFCN [70]	.875	.852	.107	.707	.764	.111	.800	.798	.132	.881	.859	.089	.755	.859	.090	.769	.794	.170
2018	DS [72]	.868	.821	.122	.708	.750	.120	.718	.740	.175	.848	.853	.078	.747	.793	.090	.757	.712	.190
	MSRNet [55]	.900	.895	.054	.746	.808	.073	.828	.838	.081	‡	‡	‡	.804	.839	.061	.802	.779	.113
	DSS [38]	.906	.882	.052	.737	.790	.063	.805	.798	.093	‡	‡	‡	.796	.824	.057	.805	.751	.122
2018	†WSS [73]	.879	.811	.104	.725	.730	.110	.804	.744	.139	.878	.822	.079	.878	.822	.079	.807	.675	.170
	DLS [75]	.826	.806	.086	.644	.725	.090	.712	.723	.130	.807	.799	.069	-	-	-	-	-	-
	NLDF [76]	.889	.875	.063	.699	.770	.080	.795	.805	.098	.888	.879	.048	.777	.816	.065	.808	.889	.125
2018	Amulet [78]	.905	.894	.059	.715	.780	.098	.805	.818	.100	.887	.886	.051	.750	.804	.085	.773	.757	.142
	FSN [79]	.897	.884	.053	.736	.802	.066	.800	.804	.093	.884	.877	.044	.761	.808	.066	.781	.755	.127
	SBF [80]	.833	.832	.091	.649	.748	.110	.726	.758	.133	.821	.829	.078	.657	.743	.109	.740	.708	.159
2018	SRM [81]	.905	.895	.054	.725	.798	.069	.817	.834	.084	.893	.887	.046	.798	.836	.059	.792	.741	.128
	UCF [82]	.890	.883	.069	.698	.760	.120	.787	.805	.115	.874	.875	.062	.742	.782	.112	.763	.753	.165
	RADF [83]	.911	.894	.049	.761	.817	.055	.800	.802	.097	.902	.888	.039	.792	.826	.061	.804	.757	.126
2018	BDMP [86]	.917	.911	.045	.734	.809	.064	.830	.845	.074	.910	.907	.039	.827	.862	.049	.806	.786	.108
	DGRL [88]	.916	.906	.043	.741	.810	.063	.830	.839	.074	.902	.897	.037	.805	.842	.050	.802	.771	.105
	PAGR [89]	.904	.889	.061	.707	.775	.071	.814	.822	.089	.897	.887	.048	.817	.838	.056	.761	.716	.147
2018	RSDNet [90]	.880	.788	.173	.715	.644	.178	‡	‡	‡	.871	.787	.156	.798	.720	.161	.790	.668	.226
	ASNet [91]	.925	.915	.047	‡	‡	‡	.848	.861	.070	.912	.906	.041	.806	.843	.061	.801	.762	.121
	PiCANet [39]	.929	.916	.035	.767	.825	.054	.838	.846	.064	.913	.905	.031	.840	.863	.040	.814	.776	.096
2018	†C2S-Net [92]	.902	.896	.053	.722	.799	.072	.827	.839	.081	.887	.889	.046	.784	.831	.062	.786	.760	.124
	RAS [93]	.908	.893	.056	.753	.814	.062	.800	.799	.101	.901	.887	.045	.807	.839	.059	.810	.764	.124

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

- **Structural measure (S-measure)** [128], different from the above metrics which only address pixel-wise errors, evaluates structural similarity between the real-valued saliency map and the binary ground-truth. S-measure ( $S$ ) considers two terms,  $S_o$  and  $S_r$ , referring to object-aware and region-aware structure similarities, respectively:

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

where  $\alpha$  is empirically set to 0.5.

- **Enhanced-alignment measure (E-measure)** [129] 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  $\phi_S$  is the enhanced alignment matrix, which reflects the correlation between  $S$  and  $G$  after subtracting their global means, respectively.

- **Salient Object Ranking (SOR)** [90] is designed for salient object subitizing, which is calculated as the normalized Spearman's Rank-Order Correlation between the ground-truth rank order  $rg_G$  and the predicted rank order  $rg_S$  of multiple salient objects in the same image:

$$SOR \triangleq \rho_{rg_G, rg_S} = \frac{\text{cov}(rg_G, rg_S)}{\sigma_{rg_G} \sigma_{rg_S}}, \quad (7)$$

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

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

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 object is partially clipped by the image boundaries.
<i>OCC</i>	Occclusion. Salient object is occluded by other objects.
<i>CSC</i>	Complex Scene. Background region contains confusing objects or rich details.
<i>BC</i>	Background Clutter. Foreground and background regions around the salient object boundaries have similar colors ( $\chi^2$ between RGB histograms less than 0.9).
<i>CSH</i>	Complex Shape. Salient object contains thin parts or holes.
<i>SO</i>	Small Object. Ratio between salient object area and image area is less than 0.1.
<i>LO</i>	Large Object. Ratio between salient object area and image area is larger than 0.5.

## 5 BENCHMARKING AND ANALYSIS

### 5.1 Overall Performance Benchmarking Results

Table 4 shows performances of 29 state-of-the-art deep SOD models and 3 top-performing classic SOD methods on 6 popular datasets widely used and tested in SOD research. Three evaluation metrics, *i.e.* maximal  $F_\beta$  [30], S-measure [128] and MAE [32] are used for assessing pixel-wise saliency prediction accuracy and the structure similarity of salient regions. All the 32 benchmarked models are representative, and have publicly available implementations or saliency prediction results on the 6 selected datasets.

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



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

TABLE 6

Attribute-based study w.r.t. salient object categories, challenges and scene categories. (-) indicates the percentage of the images with a specific attribute. *ND-avg* indicates the average score of three top-performing heuristic models: HS [34], DRFI [48] and wCtr [35]. *D-avg* indicates the average score of three top-performing deep learning models: DGRL [88], PAGR [89] and PiCANet [39]. All the three models are trained on DUTS [73]. (Best in red, worst with underline; See §5.2 for 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>OCC</i> (46.50)	<i>CSC</i> (40.44)	<i>BC</i> (47.22)	<i>CSH</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 $\mathcal{F} \uparrow$	*HS [34]	.587	.650	.636	.704	.663	.637	.631	.645	.558	.647	.629	.493	.737	.594	.627	.650
	*DRFI [48]	.635	.692	.673	.713	.674	.688	.658	.675	.599	.662	.677	.566	.747	.609	.661	.697
	*wCtr [35]	.557	.621	.624	.682	.639	.625	.605	.620	.522	.612	.606	.469	.689	.578	.613	.618
	DGRL [88]	.820	.881	.830	.728	.783	.846	.829	.830	.781	.842	.834	.724	.873	.800	.848	.840
	PAGR [89]	.834	.890	.787	.725	.743	.819	.778	.809	.770	.797	.822	.760	.802	.788	.796	.828
	PiCANet [39]	.840	.897	.846	.669	.791	.861	.843	.845	.797	.848	.850	.763	.889	.806	.862	.859
	*ND-avg	<b>.593</b>	.654	.644	<b>.700</b>	<b>.659</b>	.650	.631	.647	.560	.640	.637	<b>.509</b>	.724	<b>.594</b>	.634	<b>.655</b>
	D-avg	.831	<b>.889</b>	.821	<b>.708</b>	.772	<b>.842</b>	.817	.828	.783	.829	.836	<b>.749</b>	.855	<b>.798</b>	.836	<b>.842</b>

\* Non-deep learning model.

• **Performance evolution of deep SOD.** Since the first deep SOD model in 2015, the performance is gradually improved over time which demonstrates the progress of visual saliency computation models. Among the deep models, MAP [36] proposed in 2016 performs least impressive, since it only outputs the bounding boxes of the salient objects. This demonstrates the need for accurate annotations for more effective training and more reliable evaluations, as discussed in [30], [121].

## 5.2 Attribute-based Evaluation

Applying DNN on SOD has brought significant performance gain, while the challenges associated with foreground and background attributes remain to be conquered. A robust SOD network is expected to deal with various complex cases. In this section, we analyze the performance of three top-performance heuristic and three leading deep approaches on a hybrid benchmark with rich attribute annotations, and conduct a detailed attribute-based analysis on the performance of selected SOD approaches.

### 5.2.1 Models, benchmark and attributes

We choose three top-performing heuristic models, *i.e.* HS [34], DRFI [48] and wCtr [35], and three most recent deep methods, *i.e.* DGRL [88], PAGR [89] and PiCANet [39] to perform attribute-based analysis. All of the deep models are trained on the same dataset, *i.e.*, DUTS [73].

We construct a *hybrid benchmark* consists of 1,800 distinctive images randomly selected from 6 datasets (300 each), namely SOD [118], ECSSD [34], DUT-OMRON [52], PASCAL-S [59], HKU-IS [26] and the test set of DUTS [73]. Please be noted that this benchmark will also be used in §5.3 and §5.4.

Inspired by [59], [120], [130], we annotate each image with a rich set of attributes considering salient object categories, challenges and scene categories. The **salient objects** are categorized into *Human*, *Animal*, *Artifact* and *NatObj*

(Natural Objects), where *NatObj* includes natural objects such as fruit, plant, mountains, icebergs, water (*e.g.* lakes, streaks), *etc*. The **challenges** describe factors that often bring difficulties to SOD, such as occlusion, background cluster, complex shape and object scale, as summarized in Table 5. The **scene** of images includes *Indoor*, *Urban* and *Natural*, where the last two indicate different outdoor environments. Please note that the attributes are not mutually exclusive, *i.e.* an image can be simultaneously assigned with more than one attributes. Some sample images are shown in Fig. 4.

### 5.2.2 Analysis

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

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

• **Most and least difficult scenes.** Deep and heuristic methods perform similarly when facing different scenes (Table 6). For both types of methods, *Natural* is the easiest, which is reasonable since it takes up more than half of the samples. *Indoor* is harder than *Urban* since the former usually contains

TABLE 7

Attribute statistics of top and bottom 100 images based on F-measure. (·) indicates the percentage of the images with a specific attribute. *ND-avg* indicates the average results of three top-performing heuristic models: HS [34], DRFI [48] and wCtr [35]. *D-avg* indicates the average results of three top-performing deep models: DGRL [88], PAGR [89] and PiCANet [39]. (Two largest changes in by red if positive, blue if negative; See §5.2)

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

TABLE 8

Input perturbation study on the **hybrid benchmark** (§5.2).

Perturbations include *Gaussian blur*, *Gaussian noise*, *Rotation* and *Gray*. *ND-avg* indicates the average score of three top-performing heuristic models: HS [34], DRFI [48] and wCtr [35]. *D-avg* indicates the average score of three representative deep learning models: SRM [81], DGRL [88] and PiCANet [39]. See §5.3 for details. (Best in red, worst with underline).

Metric	Method	Original	Gaus. blur		Gaus. noise		Rotation		Gray
			$\sigma =$ 2	4	$\text{var} =$ 0.01	0.08	$15^\circ$	$-15^\circ$	
max $\mathcal{F} \uparrow$	*HS [34]	.600	-.012	-.096	-.022	-.057	+.015	+.009	-.104
	*DRFI [48]	.670	-.040	-.103	-.035	-.120	-.009	-.009	-.086
	*wCtr [35]	.611	+.006	-.000	-.024	-.136	-.004	-.003	-.070
	SRM [81]	.817	-.090	-.229	-.025	-.297	-.028	-.029	-.042
	DGRL [88]	.831	-.088	-.365	-.050	-.402	-.031	-.022	-.026
	PiCANet [39]	.848	-.048	-.175	-.014	-.148	-.005	-.008	-.039
	*ND-avg	.627	-.015	-.066	-.027	<b>-.104</b>	<b>-.000</b>	-.001	-.087
	D-avg	.832	-.075	-.256	-.041	<b>-.282</b>	-.021	<b>-.020</b>	-.037

\* Non-deep learning model.

a plunge of objects within a limited space, and often suffers from highly unevenly distributed illuminations.

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

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

### 5.3 Influences of Input Perturbations

Input perturbations such as noise and blurring often cause troubles in real world applications. In this section, we study the influences of several typical input perturbations through three representative heuristic methods and three deep methods, and show the detailed analysis on the hybrid benchmark (see §5.2).

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

As in §5.2, we choose three top-performing heuristic models, *i.e.* HS [34], DRFI [48] and wCtr [35], and three publicly available deep methods trained on DUTS [73], *i.e.* SRM [81], DGRL [88] and PiCANet [39] for studying the input perturbation influences.

The perturbation results are shown in Table 8. Overall, the heuristic methods are less sensitive towards input perturbations compared with deep methods, largely due to the robustness of hand-craft super-pixel level features. Specifically, heuristic methods are barely affected by *Rotation*, but have larger performance drops for strong *Gaussian blur*, strong *Gaussian noise* and the *Gray* effect. Deep methods suffer the most for *Gaussian blur* and strong *Gaussian noise*, which greatly reduce the richness of local information in the reception fields of shallow layers. Deep methods are relatively robust against *Rotation* due to spatial pooling in feature hierarchy.

### 5.4 Adversarial Attacks Analysis

Deep Neural Networks (DNNs) models have achieved dominant performance in various computer vision tasks, including SOD. However, modern DNNs are shown to be surprisingly susceptible to adversarial attacks, where visually imperceptible perturbations of input images would lead to completely different predictions [131]. Though being intensively studied in classification tasks, adversarial attacks in SOD are significantly under-explored.

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

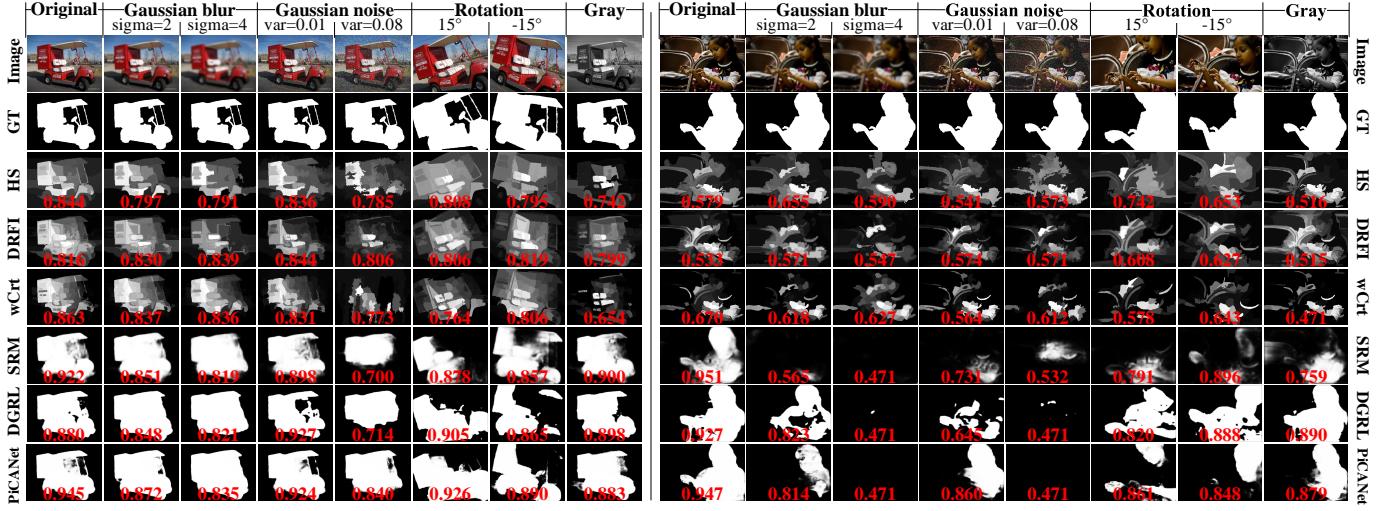


Fig. 5. Examples of saliency prediction under various input perturbations. The max  $\mathcal{F}$  values are denoted using red. See § 5.3 for more details.

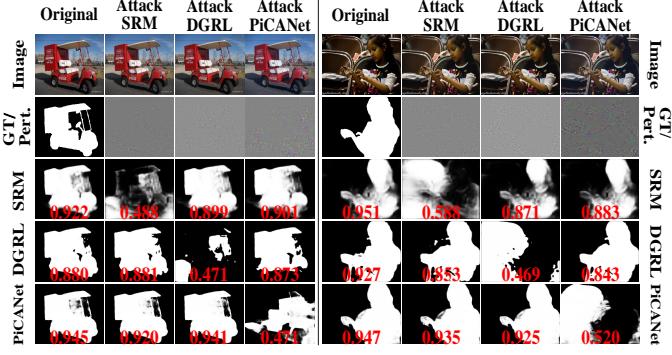


Fig. 6. Adversarial examples for saliency prediction under adversarial perturbations of different target networks. Adversarial perturbations are magnified by 10 for better visualization. The max  $\mathcal{F}$  values are denoted using red. See § 5.4 for more details.

#### 5.4.1 Robustness of SOD against Adversarial Attacks

We choose three representative deep models, *i.e.* SRM [81], DGRL [88] and PiCANet [39], to study the robustness against adversarial attack. All the three models are trained on DUTS [73]. We experiment with the ResNet [96] backbone version of the three models. The experiment is conducted on the hybrid benchmark introduced in § 5.2.

Since SOD can be viewed as a special case of semantic segmentation with two predefined categories, we resort to an adversarial attack algorithm designed for semantic segmentation, *Dense Adversary Generation* (DAG) [132], for measuring the robustness of deep SOD models. The DAG perturbations are visually imperceptible, whose maximal absolute intensity in each channel is less than 20.

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

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

Attack from	SRM [81]	DGRL [88]	PiCANet [39]
None	.817	.831	.848
SRM [81]	<u>.263</u>	.780	.842
DGRL [88]	.778	<u>.248</u>	.844
PiCANet [39]	.772	.799	<u>.253</u>

#### 5.4.2 Transferability across Networks

Transferability refers to the ability of adversarial examples generated against one model to mislead another model without any modification [133], which is widely used for black-box attack against real-world system. Given this property, we analyze the existence of transferability in SOD tasks by attacking one model using the adversarial perturbations generated for another.

The evaluation of transferability among 3 studied models (SRM [81], DGRL [88] and PiCANet [39]) is shown in Table 9. It shows that the DAG attack rarely transfers among different SOD networks. Each of the three models achieves comparable performance under the attacks generated from the other two models. This may be because that the spatial distributions of the attacks are very distinctive among different SOD models.

#### 5.5 Cross-dataset Generalization Evaluation

Datasets play an important role for both training and evaluating different deep models. In this section, we study the generalization and hardness of several main-stream SOD datasets by performing cross-dataset analysis [134], *i.e.*, to train a representative simple SOD model on one dataset, and test it on the other.

The simple SOD model is implemented as a popular bottom-up/top-down encoder-decoder architecture, where the encoder part consists of the convolution layers of VGG16 [95], and the decoder part consists of three convolutional layers for gradually making more precise pixel-wise saliency predictions. To increase the output resolution,

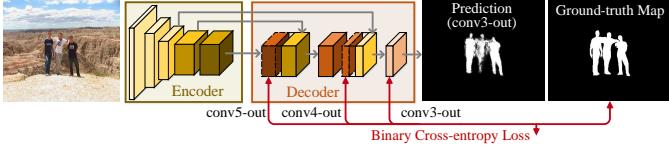


Fig. 7. Network architecture of the SOD model used in cross-dataset generalization evaluation. See §5.5 for more detailed descriptions.

TABLE 10

Results for cross-dataset generalization experiment. max  $\mathcal{F}$  ↑ for saliency prediction when training on one dataset (rows) and testing on another (columns), i.e., each row is: training on one dataset and testing on all the datasets. “Self” refers to training and testing on the same dataset (same as diagonal). “Mean Others” indicates average performance on all except self. See §5.5 for details.

Test on:	MSRA-10K [58]	ECSSD [34]	DUT-OMRON [52]	HKU-IS [26]	DUTS [73]	SOC [120]	Self	Mean others	Percent drop↓	
Train on:	MSRA10K [58]	.875	.818	.660	.849	.671	.617	.875	.723	17%
	ECSSD [34]	.844	<b>.831</b>	.630	.833	.646	.616	.831	.714	14%
	DUT-OMRON [52]	.795	.752	<b>.673</b>	.779	.623	.567	.673	.703	-5%
	HKU-IS [26]	.857	.838	.695	<b>.880</b>	.719	.639	.880	.750	15%
	DUTS [73]	.857	.834	.647	.860	<b>.665</b>	.654	.665	.770	-16%
	SOC [120]	.700	.670	.517	.666	.514	<b>.593</b>	.593	.613	-3%
	Mean others	.821	.791	.637	.811	.640	.614	-	-	-

the strides of the max-pooling layer in the 4-th block is decreased to 1, the dilation rates of the 5-th convolutional block is modified to 2, and the *pool5* layer is excluded. The side output of each attentive feature is obtained through a *Conv(1 × 1, 1)* layer with *Sigmoid* activation, and supervised by the ground-truth saliency segmentation map. The final prediction comes from the 3-nd decoder layer. An illustration of the network architecture is shown in Fig. 7.

For this study we pick six representative datasets: MSRA10K [58], ECSSD [34], DUT-OMRON [52], HKU-IS [26], DUTS [73], and SOC [120]. For each dataset, we train the SOD model with 800 randomly selected training images and test it on 200 other validation images. Please be noted that 1,000 is the maximum possible total number considering the size of the smallest selected dataset, ECSSD [34]. We repeat the training process until convergence.

Table 10 summarizes the results of cross-dataset generalization using max  $\mathcal{F}$ . Each column shows the performance of all the trained models testing on one dataset, indicating the hardness of the tested dataset. Each row shows the performance of one trained model testing on all the datasets, indicating the generalization ability of the training dataset. Please note that the numbers are not comparable with the benchmarking values shown in previous sections due to varied training/testing protocols. What matter are the relative differences. We find that SOC [120] is the most difficult dataset (lowest column *Mean others* 0.619). This may be because that SOC [120] is collected to have distinctive location distributions compared with other datasets, and may contain extremely large or small salient objects. MSRA10K [58] appears to be the easiest dataset (highest column *Mean others* 0.811), and generalizes the worst (highest row *Percent drop* 17%). DUTS [73] is shown to have the best generalization ability (lowest row *Percent drop* -16%).

## 6 DISCUSSIONS

### 6.1 Model Design

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

- **Feature Aggregation.** Efficient aggregation of hierarchical deep features are significant for pixel-wise labeling tasks since it is believed to be beneficial to integrate ‘multi-scale’ abstracted information. Existing SOD methods have brought various strategies for feature aggregation, such as multi-stream/multi-resolution fusion [55], top-down bottom-up fusion [37] or side-output fusion [38], [78], [83]. Fusion with features from other domains, e.g. fixation prediction, may also enhance the feature representation [79]. Besides, it is suggested to learn from the feature aggregation methodology of other closely related research tasks such as semantic segmentation [135]–[137], which learns semantically meaningful features for predicting pixel-level labels.

- **Loss Function.** The elaborate design of loss functions also plays an important role in training more effective models. In [91], loss functions derived from SOD evaluation metrics are used for capturing quality factors and have been empirically shown to improve saliency prediction performance. Another recent work [138] proposes to directly optimize the mean intersection-over-union loss, which brings impact to semantic segmentation as well as its binary case, i.e. foreground-background segmentation. Designing suitable loss functions for SOD is an important consideration for further improving model performance.

- **Network Topology.** Network topology determines the within-network information flow that directly affects training difficulty and parameter usage. For a classic example, in ResNet [96], the block input is directly added to the block output through skip connection, making it possible to train very deep networks. DenseNet [139] further links each layer with all its subsequent layers, greatly alleviating gradient vanishing and encouraging feature reuse. CliqueNet [140] adds bidirectional connections between two arbitrary layers within a block, maximizing the information flow through layers and reusing layer parameters multiple times.

Besides manually determining the network topology all the way up, a promising direction is to resort to automated machine learning (AutoML), which aims to find the best performing algorithms with least possible human intervention. As a promising example, *Neural Architecture Search (NAS)* [141] is able to generate competitive models for image classification and language modeling from scratch. It trains a controller RNN to generate network hyperparameters using Reinforcement Learning (RL) [142]. The computational cost of AutoML can be alleviated by transfer learning [143], [144], which makes it more practical for benefiting a wider range of more complex tasks.

The existing well-designed network topologies and the AutoML technologies all provide insights for constructing novel and effective SOD architectures in future.

- **Dynamic Inference.** The rich redundancy among DNN features facilitates its robustness against perturbed inputs, while inevitably introducing extra computational cost during inference. Besides improving the computational efficiency of DNNs using some *static* methods such as kernel decomposition [145] or parameter pruning [146], some

studies investigate on varying the amount of computation *dynamically* during testing. Bengio *et al.* [147] propose to selectively activate part of the neurons in a multi-perceptron (MLP) network during prediction. BranchyNet [148] stops the computation early once the classification entropy of the added intermediate classification branches falls below a threshold. The recently proposed ConvNet-AIG [149] adaptively updates its inference graph according to the input image, and only runs a subset of layers related to certain classes. Compared with *static* methods, these dynamic ones improve the efficiency without decreasing network parameters, thus is prone to be robust against basic adversarial attacks (*e.g.* ConvNet-AIG [149]).

For SOD model design, incorporating reasonable and effective dynamic network structure is promising for improving both efficiency and performance. For example, the specialized subsets of layers may serve as expert subnets for handling input images with various attributes.

## 6.2 Dataset Collection

Based on previous observations, we would suggest considering *data selection bias*, *annotation inconsistency*, *annotation quality* and *domain knowledge* for constructing SOD datasets in future.

- **Data selection bias.** Most existing SOD datasets collect images that contain salient objects in relatively clean background, while discarding images that do not contain any salient objects, or whose backgrounds are too clustered. However, real-world applications usually face with much more complicated situations, which can cause serious trouble to SOD models trained on these datasets. Thus, creating datasets to faithfully reflect the real world challenges is crucial for improving the generalization ability of SOD [41].

Some recent efforts have been spent to address the selection bias. For example, the SOC dataset [120] collects some non-salient images to better mimic the real-world scenes. More such efforts are encouraged to further boost the saliency prediction performance w.r.t. real-life challenges.

- **Annotation Inconsistency.** Though existing SOD datasets play an important role in training and evaluating modern SOD models, the inconsistencies among different SOD datasets shall not be neglected or overlooked. The intra-dataset inconsistencies are inevitable since the data may not be annotated by identical subjects and under identical rules/conditions.

Fig. 8 show some typical examples. The two cases in the top row represent the instance-level annotation inconsistency where there exists multiple comparable instances but either all or several of them would be annotated as salient objects. The left case in the middle row shows the inconsistency regarding the shadow. The right case in the middle row describe the inconsistency on objects of certain categories, *e.g.* the flowers in the two images are not consistently marked as salient or non-salient. The bottom left case presents the annotation of the bicycle with various degrees of precision. The bottom right case shows the inconsistency when labeling the saliency of the mirror reflection.

- **Coarse v.s. Fine Annotation.** For data-driven learning, the labeling quality is crucial for training reliable SOD models and evaluating faithfully them.

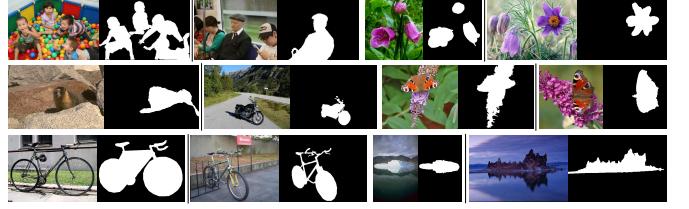


Fig. 8. Examples for annotation inconsistency. Each row shows two exemplar image pairs. See §6.2 for more detailed descriptions.

The first improvement of SOD annotation quality is to replace the bounding-boxes with pixel-wise masks for denoting the salient objects [30], [121], which greatly boost the performance of SOD models. In view of this, almost all the modern SOD datasets have been annotated with pixel-level labels. However, the labeling precisions may be different across different samples. For example, The precision for the bicycle in Fig. 8 are obviously different. There has no comprehensive study about the relation between label quality and model performance for SOD. A similar research regarding pixel-level labeling quality of semantic segmentation [150] has shown that a large number of coarse-labeled data can reach the performance of smaller number of fine-labeled data, and that pre-training with coarse labels then fine-tuning with a small number of fine labels is competitive with training with a large number of fine labels. Though some works have shown the importance of high-quality labels [120], [151], more in-depth study is in demand for SOD model training and dataset construction.

- **Domain-specific SOD datasets.** SOD has wide application scenarios such as autonomous vehicles, video games, medical image processing, *etc.*, as it helps locate objects of interest and situation awareness. Due to different scene settings, the saliency mechanism in these applications can be quite different from the one in conventional natural image setting, considering the visual appearances and semantic components. Thus, it is essential to collect SOD datasets specific for these application domains. The benefits brought by domain-specific datasets have been observed in FP, where the saliency models trained on specially collected datasets outperform other models for predicting fixations on crowds [152], webpages [153]–[155] or during driving [156], [157]. It is promising that collecting domain-specific data can help build saliency models that can better detect and segment the salient objects under specific task settings than generally trained SOD models [47].

## 6.3 Saliency Ranking and Relative Saliency

Traditionally, the salient object generally refers to the most salient object or region in a scene. However, this ‘simple’ definition may be insufficient for images where multiple salient objects exist. Thus, how to assess the saliency of co-existing objects or regions is import for designing SOD models and annotating SOD datasets.

One possible solution is to rank the saliency of objects or regions. Based on the observation that human eye fixations are often guided by the locations of salient objects in the scene, Li *et al.* [59] propose to rank the saliency of image segments using the fixation prediction.

Another solution uses the relative saliency of multiple salient instances based on the votes from several observers. For example, Islam *et al.* [90] use a stack of ground-truth maps that correspond to different levels of saliency defined by observers instead of classical binary ground-truth saliency masks to train an SOD model. The relative saliency among different instances can also serve as an important cue for salient object subitizing.

#### 6.4 Relation with Fixations

Both fixation prediction (FP) and SOD closely relate to the concept of *visual saliency* in the field of computer vision. FP dates back to early 1990s [158] which aims to predict the fixation points that would be the focus of the first glance by human viewers. SOD has a slightly shorter history dating back to [29], [30], and attempts to identify and segment the salient object(s) in the scene. FP is directly derived from the cognition and psychology community, while SOD appears more ‘computer vision’ driven by object-level applications. The generated saliency maps of the two are actually remarkably different due to the distinct purposes in saliency detection.

The strong correlation between FP and SOD has been explored in history. Early in the work of Mishara *et al.* [159], the human fixation is utilized to identify the *object of interest* for segmentation, which is known as the task of ‘active visual segmentation’. Later, a few studies (*e.g.*, [41], [59], [160], [161]) quantitatively explore and demonstrate the existence of a strong correlation between the explicit saliency judgments and human free-viewing fixations. Borji *et al.* [161] also show that both definitions for ‘the most salient object’ in a scene, *i.e.*, the one that attracts the majority of eye fixations or the first glance, would lead to similar conclusions.

Though being closely related, only a handful of models considering FP and SOD tasks at the same time. Li *et al.* [59] propose an effective combinational SOD algorithm consisting of a segmentation process followed by a saliency region ranking using FP. FSN [79] fuses the outputs of a fixation stream [99] and a semantic stream [95] to predict saliency, while it does not learn the two tasks simultaneously. SU [65] utilizes multi-task learning and performs FP and SOD in a branched network. ASNet [91] utilizes fixation map from top layers to guide saliency segmentation in lower layers. How to effectively benefit SOD from FP is still an open and unsolved problem.

There are a few SOD datasets accompanied with fixation data, such as PASCAL-S [59], DUT-OMRON [52] and a subset of XPIE [119]. However, the SOD annotations are typically not guided by the fixation data. For example, the saliency masks of PASCAL-S are constructed based on pre-segmented regions from which the ‘salient’ ones are selected using mouse-click. DUT-OMRON [52] labels the bounding boxes of salient objects without considering the fixations in the preliminary stage. On the contrary, the filtering process of the fixation data is affected by the annotated bounding box. The images in the fixation subset of XPIE [119] are collected from dataset in [162] and [163]. However, the annotation process of the binary masks is independent of the fixation data, which is just the same as images in other subsets without fixations. Considering the strong relation

between SOD and FP, it is suggested to make use of the fixation information when annotating the saliency masks during the construction of SOD datasets in future, as done in Judd-A [161] (image SOD) and VOS [164] (video SOD).

More research on models and datasets regarding the rationale behind the relation of SOD and FP is encouraged toward producing models that are more consistent with the visual selective mechanism of human.

#### 6.5 Improve SOD with Semantics

Semantic information is of key importance in high-level vision tasks such as semantic segmentation, object detection, object class discovery, *etc.* By contrast, its role in SOD is largely under-explored, partly because that SOD seemingly relies more on low-level visual cues than on high-level semantic meanings. In fact, high-level semantic information can provide very helpful guidance for detecting salient objects, especially in difficult scenes such as with highly cluttered background.

A few efforts have been devoted to facilitate SOD with semantic information [70], [72]. Besides pre-training SOD models with segmentation dataset [70], or utilizing multi-task learning to concurrently train SOD with semantic segmentation [72], a feasible direction is to enhance saliency features by incorporating segmentation features as done in some object detection methods, either through concatenation [165] or using activation [166]. Such feature enhancement utilizes semantics embedded in pixel categories to help estimate the class-agnostic saliency value for each pixel, especially in scenarios where the visual pattern is insufficient to distinguish the objects from their surroundings.

#### 6.6 SOD for Real-World Applications

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

To facilitate the application of SOD in real-world scenarios, it is considerable to utilize model compression [167] techniques to learn compact and fast SOD models with competitive prediction accuracy. Hinton *et al.* [168] extended the idea in [167] and propose *knowledge distillation* (KD), which is able to train a shallow or compressed student model under the supervision of the soft outputs from the large teacher model with minor accuracy drop for image classification. Romero *et al.* [169] further extend KD by utilizing intermediate-level feature from the teacher as ‘hints’ for the training of the student network. Such compression techniques have shown the effectiveness in improving the generalization capability and alleviating under-fitting when training faster models for object detection [170], a more challenging task compared with image classification. It is worthy of exploring compressing SOD models with these techniques for fast and accurate saliency prediction.

There are also applications where the inputs of SOD are images from other modalities (*e.g.* depth), and the labeled data is limited compared with RGB datasets. To fully exploit the existing RGB SOD datasets, besides initializing with generic RGB SOD feature representations then finetuning on data of other modalities, one can use *cross modal distillation* [171], which transfers the supervision from labeled RGB images to the paired unlabeled data with new modalities and effectively learns feature hierarchies. In this way, the existing DNN architecture for general SOD can be extended to other modalities without having to collect additional large-scale labeled datasets.

## 7 CONCLUSION

In this paper we present, to the best of our knowledge, the first comprehensive review of SOD with focus on deep learning techniques. We first carefully review and organize deep learning-based SOD models from several different perspectives, including network architecture, level of supervision, *etc.* We then summarize popular SOD datasets and evaluation criteria, and compile a thorough performance benchmarking of major SOD methods.

Next, we investigate several previously under-explored issues with novel efforts on benchmarking and baselines. In particular, we perform attribute-based performance analysis by compiling and annotating a new dataset and testing several representative SOD algorithms. We also study the robustness of SOD methods w.r.t. various input perturbations. Moreover, for the first time in SOD, we investigate the robustness and transferability of deep SOD models w.r.t. adversarial attacks. In addition, we assess the generalization and hardness of existing SOD datasets through cross-dataset generalization experiment. We finally look through several open issues and challenges of SOD in deep learning era, and provide insightful discussions on possible research directions in future.

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

## REFERENCES

- [1] J.-Y. Zhu, J. Wu, Y. Xu, E. Chang, and Z. Tu, "Unsupervised object class discovery via saliency-guided multiple class learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 4, pp. 862–875, 2015.
- [2] F. Zhang, B. Du, and L. Zhang, "Saliency-guided unsupervised feature learning for scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 4, pp. 2175–2184, 2015.
- [3] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *Proc. ACM Int. Conf. Mach. Learn.*, 2015, pp. 2048–2057.
- [4] H. Fang, S. Gupta, F. Iandola, R. K. Srivastava, L. Deng, P. Dollár, J. Gao, X. He, M. Mitchell, J. C. Platt *et al.*, "From captions to visual concepts and back," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1473–1482.
- [5] A. Das, H. Agrawal, L. Zitnick, D. Parikh, and D. Batra, "Human attention in visual question answering: Do humans and deep networks look at the same regions?" *Computer Vision and Image Understanding*, vol. 163, pp. 90–100, 2017.
- [6] Z. Ren, S. Gao, L.-T. Chia, and I. W.-H. Tsang, "Region-based saliency detection and its application in object recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 5, pp. 769–779, 2014.
- [7] D. Zhang, D. Meng, L. Zhao, and J. Han, "Bridging saliency detection to weakly supervised object detection based on self-paced curriculum learning," in *International Joint Conferences on Artificial Intelligence*, 2016.
- [8] W. Wang, J. Shen, R. Yang, and F. Porikli, "Saliency-aware video object segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 1, pp. 20–33, 2018.
- [9] H. Song, W. Wang, S. Zhao, J. Shen, and K.-M. Lam, "Pyramid dilated deeper convlstm for video salient object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2018.
- [10] Y. Wei, J. Feng, X. Liang, M.-M. Cheng, Y. Zhao, and S. Yan, "Object region mining with adversarial erasing: A simple classification to semantic segmentation approach," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017.
- [11] Y. Wei, X. Liang, Y. Chen, X. Shen, M.-M. Cheng, J. Feng, Y. Zhao, and S. Yan, "STC: A simple to complex framework for weakly-supervised semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 11, pp. 2314–2320, 2017.
- [12] X. Wang, S. You, X. Li, and H. Ma, "Weakly-supervised semantic segmentation by iteratively mining common object features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018.
- [13] R. Zhao, W. Ouyang, and X. Wang, "Unsupervised salience learning for person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 3586–3593.
- [14] S. Bi, G. Li, and Y. Yu, "Person re-identification using multiple experts with random subspaces," *Journal of Image and Graphics*, vol. 2, no. 2, 2014.
- [15] J. Han, E. J. Pauwels, and P. De Zeeuw, "Fast saliency-aware multi-modality image fusion," *Neurocomputing*, vol. 111, pp. 70–80, 2013.
- [16] P. L. Rosin and Y.-K. Lai, "Artistic minimal rendering with lines and blocks," *Graphical Models*, vol. 75, no. 4, pp. 208–229, 2013.
- [17] 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.
- [18] S. Avidan and A. Shamir, "Seam carving for content-aware image resizing," in *ACM Trans. Graph.*, vol. 26, no. 3, 2007, p. 10.
- [19] J. Sun and H. Ling, "Scale and object aware image retargeting for thumbnail browsing," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011.
- [20] 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.
- [21] 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.
- [22] Y. Sugano, Y. Matsushita, and Y. Sato, "Calibration-free gaze sensing using saliency maps," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 2667–2674.
- [23] A. Borji and L. Itti, "Defending yarbus: Eye movements reveal observers' task," *Journal of Vision*, vol. 14, no. 3, pp. 29–29, 2014.
- [24] A. Karpathy, S. Miller, and L. Fei-Fei, "Object discovery in 3d scenes via shape analysis," in *Proc. IEEE Conf. Robot. Autom.*, 2013, pp. 2088–2095.
- [25] S. Frintrop, G. M. García, and A. B. Cremers, "A cognitive approach for object discovery," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 2329–2334.
- [26] G. Li and Y. Yu, "Visual saliency based on multiscale deep features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 5455–5463.
- [27] L. Wang, H. Lu, X. Ruan, and M.-H. Yang, "Deep networks for saliency detection via local estimation and global search," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3183–3192.
- [28] R. Zhao, W. Ouyang, H. Li, and X. Wang, "Saliency detection by multi-context deep learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1265–1274.
- [29] T. Liu, J. Sun, N.-N. Zheng, X. Tang, and H.-Y. Shum, "Learning to detect a salient object," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2007, pp. 1–8.

- [30] R. Achanta, S. Hemami, F. Estrada, and S. Sussstrunk, "Frequency-tuned salient region detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 1597–1604.
- [31] M.-M. Cheng, G.-X. Zhang, N. J. Mitra, X. Huang, and S.-M. Hu, "Global contrast based salient region detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 409–416.
- [32] F. Perazzi, P. Krähenbühl, Y. Pritch, and A. Hornung, "Saliency filters: Contrast based filtering for salient region detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*. IEEE, 2012, pp. 733–740.
- [33] J. Wang, H. Jiang, Z. Yuan, M.-M. Cheng, X. Hu, and N. Zheng, "Salient object detection: A discriminative regional feature integration approach," *Int. J. Comput. Vis.*, vol. 123, no. 2, pp. 251–268, 2017.
- [34] Q. Yan, L. Xu, J. Shi, and J. Jia, "Hierarchical saliency detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 1155–1162.
- [35] W. Zhu, S. Liang, Y. Wei, and J. Sun, "Saliency optimization from robust background detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 2814–2821.
- [36] J. Zhang, S. Sclaroff, Z. Lin, X. Shen, B. Price, and R. Mech, "Unconstrained salient object detection via proposal subset optimization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 5733–5742.
- [37] N. Liu and J. Han, "DHSNet: Deep hierarchical saliency network for salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 678–686.
- [38] Q. Hou, M.-M. Cheng, X. Hu, A. Borji, Z. Tu, and P. Torr, "Deeply supervised salient object detection with short connections," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 3203–3212.
- [39] N. Liu, J. Han, and M.-H. Yang, "Picanet: Learning pixel-wise contextual attention for saliency detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3089–3098.
- [40] 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.
- [41] 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.
- [42] T. V. Nguyen, Q. Zhao, and S. Yan, "Attentive systems: A survey," *Int. J. Comput. Vis.*, vol. 126, no. 1, pp. 86–110, 2018.
- [43] A. Borji, M.-M. Cheng, Q. Hou, H. Jiang, and J. Li, "Salient object detection: A survey," *arXiv preprint arXiv:1411.5878*, 2014.
- [44] 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.
- [45] 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.
- [46] 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.
- [47] A. Borji, "Saliency prediction in the deep learning era: An empirical investigation," *arXiv preprint arXiv:1810.03716*, 2018.
- [48] 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.
- [49] H. Peng, B. Li, H. Ling, W. Hu, W. Xiong, and S. J. Maybank, "Salient object detection via structured matrix decomposition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 818–832, 2017.
- [50] R. Cong, J. Lei, H. Fu, Q. Huang, X. Cao, and C. Hou, "Co-saliency detection for RGBD images based on multi-constraint feature matching and cross label propagation," *IEEE Trans. Image Process.*, vol. 27, no. 2, pp. 568–579, 2018.
- [51] Y. Wei, F. Wen, W. Zhu, and J. Sun, "Geodesic saliency using background priors," *Proc. Eur. Conf. Comput. Vis.*, pp. 29–42, 2012.
- [52] 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.
- [53] 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.
- [54] 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.
- [55] 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.
- [56] 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.
- [57] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1–9.
- [58] M.-M. Cheng, G.-X. Zhang, N. J. Mitra, X. Huang, and S.-M. Hu, "Global contrast based salient region detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011.
- [59] 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.
- [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] G. Li and Y. Yu, "Deep contrast learning for salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 478–487.
- [62] 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.
- [63] 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.
- [64] 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.
- [65] 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.
- [66] 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.
- [67] 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.
- [68] 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.
- [69] 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.
- [70] 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.
- [71] 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.
- [72] X. Li, L. Zhao, L. Wei, M.-H. Yang, F. Wu, Y. Zhuang, H. Ling, and J. Wang, "Deepsaliency: Multi-task deep neural network model for salient object detection," *IEEE Trans. Image Process.*, vol. 25, no. 8, pp. 3919 – 3930, 2016.
- [73] 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.
- [74] 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.
- [75] 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.
- [76] 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.
- [77] S. He, J. Jiao, X. Zhang, G. Han, and R. W. Lau, "Delving into salient object subitizing and detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 1059–1067.

- [78] 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.
- [79] 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.
- [80] 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.
- [81] 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.
- [82] 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.
- [83] 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.
- [84] G. Li, Y. Xie, and L. Lin, "Weakly supervised salient object detection using image labels," in *AAAI Conference on Artificial Intelligence*, 2018.
- [85] 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.
- [86] 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.
- [87] 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.
- [88] 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.
- [89] 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.
- [90] M. Amirul Islam, M. Kalash, and N. D. B. Bruce, "Revisiting salient object detection: Simultaneous detection, ranking, and subtitizing of multiple salient objects," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018.
- [91] 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. 1171–1172.
- [92] 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.
- [93] 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.
- [94] 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.
- [95] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Representations*, 2015.
- [96] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
- [97] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu, "Spatial transformer networks," in *Proc. Advances Neural Inf. Process. Syst.*, 2015, pp. 2017–2025.
- [98] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, 2018.
- [99] S. S. Kruthiventi, K. Ayush, and R. V. Babu, "Deepfix: A fully convolutional neural network for predicting human eye fixations," *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4446–4456, 2017.
- [100] S. Xie and Z. Tu, "Holistically-nested edge detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1395–1403.
- [101] 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.
- [102] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. Advances Neural Inf. Process. Syst.*, 2015, pp. 802–810.
- [103] 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.
- [104] J. L. Long, N. Zhang, and T. Darrell, "Do convnets learn correspondence?" in *Proc. Advances Neural Inf. Process. Syst.*, 2014, pp. 1601–1609.
- [105] 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.
- [106] 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.
- [107] 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.
- [108] B. Jiang, L. Zhang, H. Lu, C. Yang, and M.-H. Yang, "Saliency detection via absorbing markov chain," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 1665–1672.
- [109] X. Li, H. Lu, L. Zhang, X. Ruan, and M.-H. Yang, "Saliency detection via dense and sparse reconstruction," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 2976–2983.
- [110] 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.
- [111] R. Caruana, "Multitask learning," *Machine learning*, vol. 28, no. 1, pp. 41–75, 1997.
- [112] P. O. Pinheiro, R. Collobert, and P. Dollár, "Learning to segment object candidates," in *Proc. Advances Neural Inf. Process. Syst.*, 2015, pp. 1990–1998.
- [113] R. Girshick, "Fast r-cnn," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1440–1448.
- [114] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Proc. Advances Neural Inf. Process. Syst.*, 2015, pp. 91–99.
- [115] 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.
- [116] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Object detectors emerge in deep scene cnns," in *Proc. Int. Conf. Learn. Representations*, 2015.
- [117] 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.
- [118] 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.
- [119] 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.
- [120] 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.
- [121] 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.
- [122] 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.
- [123] J. Deng, "A large-scale hierarchical image database," *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009.
- [124] 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.

- [125] 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.
- [126] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge results," 2007.
- [127] 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.
- [128] 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.
- [129] 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.
- [130] 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.
- [131] 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.
- [132] 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.
- [133] 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.
- [134] A. Torralba and A. A. Efros, "Unbiased look at dataset bias," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 1521–1528.
- [135] H. Ding, X. Jiang, B. Shuai, A. Qun Liu, and G. Wang, "Context contrasted feature and gated multi-scale aggregation for scene segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 2393–2402.
- [136] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "Learning a discriminative feature network for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018.
- [137] F. Yu, D. Wang, E. Shelhamer, and T. Darrell, "Deep layer aggregation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018.
- [138] 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.
- [139] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2261–2269.
- [140] Y. Yang, Z. Zhong, T. Shen, and Z. Lin, "Convolutional neural networks with alternately updated clique," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018.
- [141] B. Zoph and Q. V. Le, "Neural architecture search with reinforcement learning," in *Proc. Int. Conf. Learn. Representations*, 2017.
- [142] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Machine learning*, vol. 8, no. 3-4, pp. 229–256, 1992.
- [143] C. Wong, N. Houlsby, Y. Lu, and A. Gesmundo, "Transfer learning with neural automl," in *Proc. Advances Neural Inf. Process. Syst.*, 2018, pp. 8366–8375.
- [144] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, "Learning transferable architectures for scalable image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 8697–8710.
- [145] M. Jaderberg, A. Vedaldi, and A. Zisserman, "Speeding up convolutional neural networks with low rank expansions," in *Proceedings of the British Machine Vision Conference*, 2014.
- [146] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding," in *Proc. Int. Conf. Learn. Representations*, 2016.
- [147] 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.
- [148] S. Teerapittayanon, B. McDaniel, and H. Kung, "Branchynet: Fast inference via early exiting from deep neural networks," in *Proc. Int. Conf. Pattern Recognit.*, 2016, pp. 2464–2469.
- [149] A. Veit and S. Belongie, "Convolutional networks with adaptive inference graphs," in *Proc. Eur. Conf. Comput. Vis.*, 2018.
- [150] 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.
- [151] Q. Hou, M.-M. Cheng, X. Hu, A. Borji, Z. Tu, and P. H. Torr, "Deeply supervised salient object detection with short connections," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2018.
- [152] M. Jiang, J. Xu, and Q. Zhao, "Saliency in crowd," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 17–32.
- [153] C. Shen and Q. Zhao, "Webpage saliency," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 33–46.
- [154] C. Shen, X. Huang, and Q. Zhao, "Predicting eye fixations on webpage with an ensemble of early features and high-level representations from deep network," *IEEE Trans. Multimedia*, vol. 17, no. 11, pp. 2084–2093, 2015.
- [155] Q. Zheng, J. Jiao, Y. Cao, and R. W. Lau, "Task-driven webpage saliency," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 287–302.
- [156] A. Borji, D. N. Sihite, and L. Itti, "Probabilistic learning of task-specific visual attention," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 470–477.
- [157] A. Palazzi, F. Solera, S. Calderara, S. Alletto, and R. Cucchiara, "Where should you attend while driving?" *arXiv preprint arXiv:1611.08215*, 2016.
- [158] 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.
- [159] A. K. Mishra, Y. Aloimonos, L. F. Cheong, and A. Kassim, "Active visual segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 639–653, 2012.
- [160] 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.
- [161] 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.
- [162] N. Bruce and J. Tsotsos, "Saliency based on information maximization," in *Proc. Advances Neural Inf. Process. Syst.*, 2006, pp. 155–162.
- [163] T. Judd, K. Ehinger, F. Durand, and A. Torralba, "Learning to predict where humans look," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2009, pp. 2106–2113.
- [164] J. Li, C. Xia, and X. Chen, "A benchmark dataset and saliency-guided stacked autoencoders for video-based salient object detection," *IEEE Trans. Image Process.*, vol. 27, no. 1, pp. 349–364, 2018.
- [165] S. Gidaris and N. Komodakis, "Object detection via a multi-region and semantic segmentation-aware cnn model," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1134–1142.
- [166] Z. Zhang, S. Qiao, C. Xie, W. Shen, B. Wang, and A. L. Yuille, "Single-shot object detection with enriched semantics," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 5813–5821.
- [167] 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.
- [168] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," in *Proc. Advances Neural Inf. Process. Syst. - workshops*, 2014.
- [169] 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.
- [170] 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.
- [171] S. Gupta, J. Hoffman, and J. Malik, "Cross modal distillation for supervision transfer," in *Proc. Int. Conf. Pattern Recognit.*, 2016, pp. 2827–2836.