Salient Object Detection Driven by Fixation Prediction

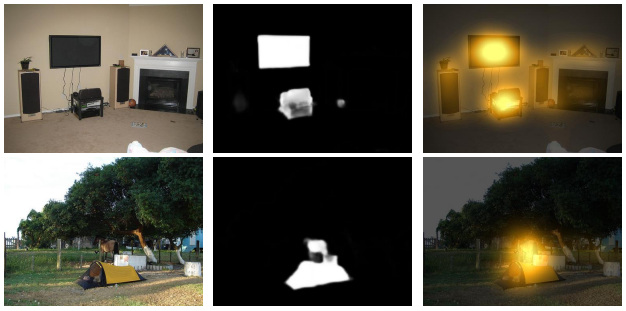
Wenguan Wang [1], Jianbing Shen* [1,2], Xingping Dong [1], Ali Borji[3]
[1]Beijing Lab of Intelligent Information Technology, School of Computer Science, Beijing Institute of Technology, China
[2]Inception Institute of Artificial Intelligence, Abu Dhabi, UAE
[3]Department of Computer Science, University of Central Florida, USA
wenguanwang.ai@gmail.com, {shenjianbing, dongxingping}@bit.edu.cn, aborji@crcv.ucf.edu

Abstract

Research in visual saliency has been focused on two major types of models namely fixation prediction and salient object detection. The relationship between the two, however, has been less explored. In this paper, we propose to employ the former model type to identify and segment salient objects in scenes. We build a novel neural network called Attentive Saliency Network (ASNet)[1] that learns to detect salient objects from fixation maps. The fixation map, derived at the upper network layers, captures a high-level understanding of the scene. Salient object detection is then viewed as fine-grained object-level saliency segmentation and is progressively optimized with the guidance of the fixation map in a top-down manner. ASNet is based on a hierarchy of convolutional LSTMs (convLSTMs) that offers an efficient recurrent mechanism for sequential refinement of the segmentation map. Several loss functions are introduced for boosting the performance of the ASNet. Extensive experimental evaluation shows that our proposed ASNet is capable of generating accurate segmentation maps with the help of the computed fixation map. Our work offers a deeper insight into the mechanisms of attention and narrows the gap between salient object detection and fixation prediction.

1. Introduction

Salient object detection (SOD) has been studied extensively for more than a decade (since [37]). It has several applications such as segmentation [52, 10], object proposal generation [2], and image resizing [57, 50]. Recently, the use of deep neural networks for saliency detection has been trending. Although promising results have been achieved, they occasionally fail to detect the most salient object in complex cluttered scenes containing several objects (such as the ones shown in Fig. 1 (a)). Additionally, for current computational saliency models, their connection with how humans explicitly choose salient objects or watch natural scenes are less clear (as discussed in [3, 6]).



(a) (b) (c)

Figure 1. **Given complex scenes like (a), what are the salient objects?** We propose the Attentive Saliency Network (ASNet) that infers the object saliency (b) from predicted fixation maps (c), which is consistent with human attention mechanisms.

In this paper, we take a further step towards a more biologically plausible SOD model equipped with high-level prior of *fixation map*. The human attention prior, represented by eye movements or from a fixation prediction model, is instinctive and more consistent with visual processing of human visual system. The suggested model not only generates high-quality object saliency maps, but also pushes the boundary of SOD research by building a close connection to human fixation prediction (FP). As shown in Fig. 1, our model infers object saliency using the fixation prior, where this prior acts as a selective mechanism to enhance the saliency representation for the purpose of accurate object saliency inference. Our algorithm has bias on the assumption that aligns with the core views of previous studies [34, 4] that explored the relationship between eye movements (implicit saliency) and explicit object saliency. These studies confirmed a strong correlation between fixations and salient objects. Further beyond treating FP and SOD as two separate tasks that are learned in a branched

*Corresponding author: *Jianbing Shen*. This work was supported in part by the Beijing Natural Science Foundation under Grant 4182056, the Fok Ying Tung Education Foundation under Grant 141067, and the Specialized Fund for Joint Building Program of Beijing Municipal Education Commission.

[1]Available at: https://github.com/wenguanwang/ASNet.

network (as in [26]), these two tasks are more deeply interconnected in our model. The fixation map provides a high-level signal, which is learned from upper layers of our neural network. It is then used for salient object detection in a top-down manner. This process is straightforward and similar to how humans process a scene sequentially (*i.e.*, first paying attention to important areas of a scene quickly, and then taking more efforts for precise segmentation). Further, leveraging the rich information from exiting large-scale eye movement datasets can improve the robustness and generalization ability of SOD models.

The proposed Attentive Saliency Network (ASNet) is based on *convolutional LSTM* (convLSTM) [59], which has convolutional structures in both the input-to-state and state-to-state transitions. Further, beyond the *fully connected LSTM* models, convLSTM encodes the spatial information via convolution operations to facilitate pixel-wise saliency labeling. The key advantage of LSTM is iterative removal of irrelevant information and learning powerful representations through updating the memory cell. This allows the ASNet to progressively optimize features for better saliency estimation in a feed-forward strategy. By stacking multiple convLSTMs, the ASNet is trainable for gradually rendering object saliency from fixation map in a top-down manner.

Our contributions are manifold:

- We aim to infer salient objects (captured in lower network layers) from the fixation map (encoded in higher layers) within a unified neural network. This goes one step beyond previous deep learning based saliency models and offers a deep insight into the confluence between fixation prediction and salient object detection.
- We present the Attentive Saliency Network (ASNet) which is a hierarchy of convLSTMs for step-wise inference of object saliency. ConvLSTM has the advantage of the improved flow of information with recurrent connections, which results in more powerful saliency representation.
- We introduce novel loss functions for SOD, derived from exiting SOD evaluation metrics, for capturing several quality factors. As we will show empirically, these new loss functions lead to higher performance.

2. Related Work

In this section, we first briefly review the fixation prediction (§ 2.1) and salient object detection literature (§ 2.2). Then, in § 2.3, we discuss studies exploring the relationship between the above two tasks.

2.1. Fixation Prediction (FP)

Fixation Prediction (FP) aims to identify the fixation points that human viewers would focus on at first glance. It has a long history from [19] and is still active in vision research. From the view of mechanism to obtain attention, **early attention models** can be classified into different schools [5], such as cognitive model [19, 28, 40, 42, 41], Bayesian model [63], decision theoretic model [12], information theoretic model [7], graphical model [13], spectral analysis model [16], pattern classification model [23], *etc*. We refer the readers to [5] for more detailed overview.

More recently, many **deep learning based attention models** have been proposed. The eDN model [46] represented an early architecture that automatically learns deep representations for FP. After that, DeepFix [25], SALICON net [18], Mr-CNN [36], Shallow and Deep [43], attentive LSTM [11], DVA net [51], were successively proposed, with the use of deeper networks and more complex architectures. Jetley *et al.* [20] tested several loss functions based on probability distance measures and found Bhattacharyya distance could give the best performance. Those deep learning solutions generally achieved better performance, compared with traditional non-deep learning techniques.

2.2. Salient Object Detection (SOD)

Salient object detection (SOD) aims at highlighting salient object regions in images. Different from FP that is originated from cognition and psychology research communities, SOD is more a computer vision task that is driven by object-level applications [56]. The history of SOD is relatively more recent and dates back to the works of Liu *et al.* [37] and Achanta *et al.* [1]. Due to lack of the knowledge of image content, most **non-deep learning SOD models** [10, 58, 60, 21, 67, 53, 55] are based on low-level features and certain heuristic hypothesizes (such as *contrast*, *background prior*). Please see [4] for detailed overviews.

In more recent work, **deep learning based SOD models** have made substantial improvement. These methods mainly utilize multi-scale and multi-level representations [31, 64, 38], integrate both local estimation and global search [47], explore global and local context information [66], combine pixel- and segment-level features [32], develop level set [17], or consider short connections with skip-layer structures [15], based on neural network. Some other methods try to integrate deep learning models with hand-crafted features [30], study saliency prior [48], or exploit various deep learning architectures [27, 35, 65, 49, 54].

2.3. The Relationship between FP and SOD

Although SOD has been extensively studied in computer vision research, only few studies (*e.g.*, [39, 4, 34]) have explored how humans explicitly choose salient objects. They have quantitatively confirmed that object saliency judgments agree with human fixations. According to the analysis in [4], there exists a strong correlation between explicit saliency judgments and free-viewing fixations, which can be viewed as two proxies of visual attention. Li *et al.* [34] have demonstrated that, unlike FP datasets, there exists a heavy bias in many widely used SOD datasets. Most of the

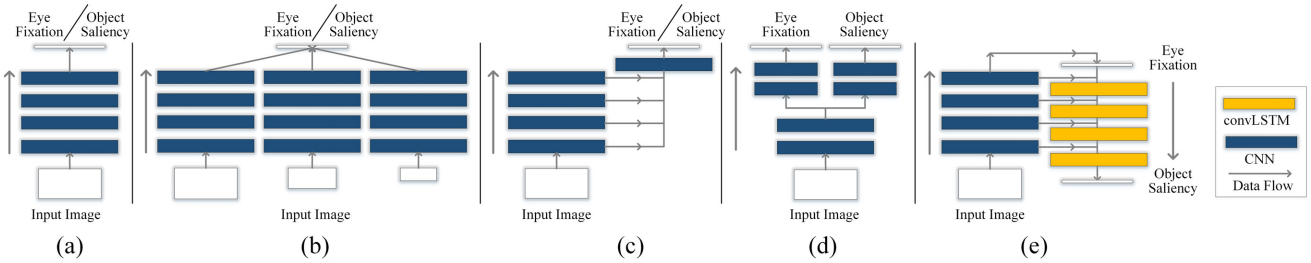


Figure 2. **Typical network architectures used in previous FP or SOD models and our ASNet.** (a) Single-stream network, (b) Multi-stream network, and (c) Skip-layer network. (d) Branched network adopted in [26], where FP and SOD are achieved via two branches sharing several bottom layers. (e) The adopted ASNet captures fixation map from upper layer, which is indicative of the inference of object saliency from lower layers. Stack of convLSTMs are adopted for iteratively optimizing features, while preserving spatial information.

SOD datasets have only a few obvious objects in the scene.

Next, we discuss several representative deep models in SOD or FP from the view of network architecture. This would better situate our work with respect to previous works and help to highlight our contributions. As shown in Fig. 2, most deep learning models for FP or SOD only consider a single task. Typical architectures include: (a) *single-stream network* [25, 20, 43, 65, 17] (standard architecture), (b) *multi-stream network* [18, 66, 31, 36] (training with multi-scale inputs), and (c) *skip-layer network* [15, 32, 64, 38] (concatenate multi-layer responses for final output). As seen, previous deep learning based works often treated FP and SOD as two unrelated tasks.

Instead of performing FP or SOD separately, we exploit the correlation between fixations and salient objects via tightly coupling these two tasks in a unified deep learning architecture. There are only few methods consider FP and SOD tasks together. In [9], fixation map from a *pre-trained* FP model is used as an extra cue for guiding SOD, while it didn't emphasize learning both FP and SOD simultaneously. In [26], FP and SOD are achieved via two separate network branches, which only share weights in several lower layers (Fig. 2 (d)). In our method, as illustrated in Fig. 2 (e), fixation map (high-level knowledge captured in top layers) is used for guiding accurate object saliency detection in lower layers. Thus our approach goes beyond above work by learning FP and SOD within a unified network and in a top-down end-to-end manner.

3. Our Approach

Given an input image, the goal is to produce a pixel-wise saliency map to highlight salient object regions. As demonstrated in Fig. 3, the proposed ASNet first captures a global and high-level understanding of a scene in its higher layers, by learning to predict human fixations (§3.1). Then, it uses a stack of convLSTMs to progressively infer object saliency from the fixation map in a top-down and coarse-to-fine manner (§3.2). The whole network is simultaneously trained to predict fixation locations and to detect salient objects in an end-to-end way (§3.3).

3.1. Fixation Predicting

At the bottom of ASNet resides a stack of convolutional layers where the lower layers respond to primitive image features such as edges, corners and shared common patterns, and the higher layers extract semantic information like objects or faces. The ASNet learns the FP as a high-level task towards modeling human fixation locations with the utilization of features from higher layers, and achieves the SOD by optimizing the fixation prior with the features from the lower layers.

The lower convolutional layers are borrowed from the first five convolutional blocks of VGGNet [45] (13 convolutional layers in total). We omit the last pooling layer (*pool5*) for preserving more spatial information. For a training image, with a typical resolution of $224 \times 224 \times 3$, we compute a convolutional layer by applying a 3×3 kernel with *sigmoid* activation function, to the last convolutional feature map ($14 \times 14 \times 512$). The result is a probability map $P \in [0,1]^{14\times14}$ which is used as a fixation prior from global and high-level image context. The model for the task of FP is trained via minimizing the following *Kullback-Leibler Divergence* (KL-Div) loss function:

$$\mathcal{L}_{Att}(G,P) = \frac{1}{14\times14}\sum_{x}^{14\times14} g_x \log(\frac{g_x}{p_x}), \quad (1)$$

where G denotes the resized ground-truth attention map $G \in [0,1]^{14\times14}$ and $g_x \in G$, $p_x \in P$. The gray-scale fixation map is obtained via filtering the binary fixation map using a Gaussian filter with small variance. The KL-Div measure, the minimization of which is equivalent to cross-entropy minimization, is widely used in visual saliency prediction [18, 51]. In the next section, we will leverage such fixation map as the prior for producing object saliency.

3.2. Detecting Object Saliency with Fixation Prior

The fixation map P gives a coarse but informative prior regarding visually salient regions. A number of previous studies for pixel-labeling tasks such as semantic segmentation [44], and salient object detection [35, 48], have shown that neural networks are capable of producing fine-gained

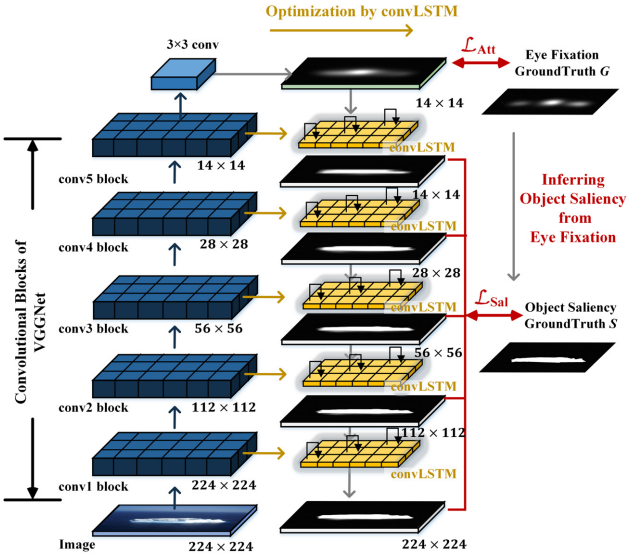


Figure 3. **Architecture of the proposed ASNet.** The fixation map is learned from the upper layers and is used by the ASNet to locate the salient objects. Then, the fine-grained object saliency is gradually inferred from lower layers and is successively optimized via the recurrent architecture of convLSTM. Zoom-in for details.

labeling results via incorporating high-level information encoded in upper network layers. Here, we desire our model to be able to infer precise object saliency from the fixation map predicted in the upper network layers.

The network is trained for detecting and successively refining the salient object via aggregating information from high-level fixation map and the spatially rich information from low-level network features. As shown in Fig. 3, the SOD is computed in a top-down fashion, successively integrating information from earlier layers. Multiple convLSTM networks [59] (the yellow blocks in Fig. 3) are stacked for building more meaningful feature representations with recurrent connections. We leverage the sequential nature of LSTM to process features in an iterative way. For a certain layer, convLSTM discards less informative features while enhances informative features, thus generating gradually improved saliency maps.

ConvLSTM extends traditional fully connected LSTM [14] to consume spatial features. Basically, this is achieved by substituting dot products with convolutional operations in the LSTM equations. ConvLSTM has convolutional structures in both the input-to-state and state-to-state transitions, which can preserve the spatial information of convolutional feature map, thus enabling our network to produce a pixel-wise labeling.

A schematic diagram of convLSTM is presented in Fig. 4. Similar to traditional gated LSTMs, the convLSTM uses the memory cells and gates to control information flow. It works by sequentially updating an internal state $\mathcal{H}$ and memory cell $\mathcal{C}$, according to the values of three sigmoid gates i, f, c. At each step t, as a new input $\mathcal{X}_t$ arrives, its information will be accumulated to the cell if the input gate i_t is activated. Also, the past cell status $\mathcal{C}_{t-1}$ could be "forgotten" in this process if the forget gate f_t is on. Whether the latest cell status $\mathcal{C}_t$ should be propagated to the final state $\mathcal{H}_t$ is further controlled by the output gate o_t. Formally, above memory update process at step t is driven by the following equations:

$$i_t = \sigma(W_i^{\mathcal{X}} * \mathcal{X}_t + W_i^{\mathcal{H}} * \mathcal{H}_{t-1} + b_i), \quad (2)$$

$$f_t = \sigma(W_f^{\mathcal{X}} * \mathcal{X}_t + W_f^{\mathcal{H}} * \mathcal{H}_{t-1} + b_f), \quad (3)$$

$$o_t = \sigma(W_o^{\mathcal{X}} * \mathcal{X}_t + W_o^{\mathcal{H}} * \mathcal{H}_{t-1} + b_o), \quad (4)$$

$$\mathcal{C}_t = f_t \circ \mathcal{C}_{t-1} + i_t \circ \tanh(W_c^{\mathcal{X}} * \mathcal{X}_t + W_c^{\mathcal{H}} * \mathcal{H}_{t-1} + b_c), \quad (5)$$

$$\mathcal{H}_t = o_t \circ \tanh(\mathcal{C}_t), \quad (6)$$

where '$*$' denotes the convolution operator and '$\circ$' represents element-wise product. σ and $\tanh$ are the activation functions of logistic sigmoid and hyperbolic tangent. The inputs $\mathcal{X}_t$, cell memory $\mathcal{C}_t$, hidden states $\mathcal{H}_t$ and gates i_t, f_t, c_t are 3D tensors whose spatial dimensions are the same. Ws and bs are the learned weights and biases.

In our case, the convLSTM takes the features $\mathcal{X}$ extracted from the convolutional neural network (from the last convolutional layers prior to pooling layers) as input, and produces refined saliency features for final saliency estimation. Since it operates on static images, the input features in all steps are the same: $\mathcal{X}_1 = \ldots = \mathcal{X}_t = \mathcal{X}$ (see Fig. 4). Here, we take the advantages of recurrent natures of LSTM for iteratively optimizing the saliency features of static images, instead of using LSTM for modeling the temporal dependency of sequential data.

With the learned fixation prior $P \in [0,1]^{14\times14}$, we first combine P with the convolutional features from *conv5-3* and feed them into a convLSTM. In each time step, the convLSTM is trained for inferring the salient object with the knowledge of fixation information, and sequentially optimizes the features with the updated memory cell and hidden states (see Fig. 4 (b)). Thus, the features are reorganized towards better representation of object saliency. More specially, we first compress the feature responses from *conv5-3* layer via a convolutional layer with 64 filters to lower computational costs and adopt *sigmoid* activation for regularizing the response from features to lie within the same range ($[0,1]$) of P. Then, the attention prior map P is concatenated with the compressed features and fed into the convLSTM. We apply a 1×1 convolution kernel to the final convLSTM output $\mathcal{H}$ for obtaining an object saliency map $Q \in [0,1]^{14\times14}$.

Several different metrics have been proposed for evaluating saliency models and no single metric can fully summarize the performance of a model. This motivates us to combine the classical *weighted cross-entropy* loss function with

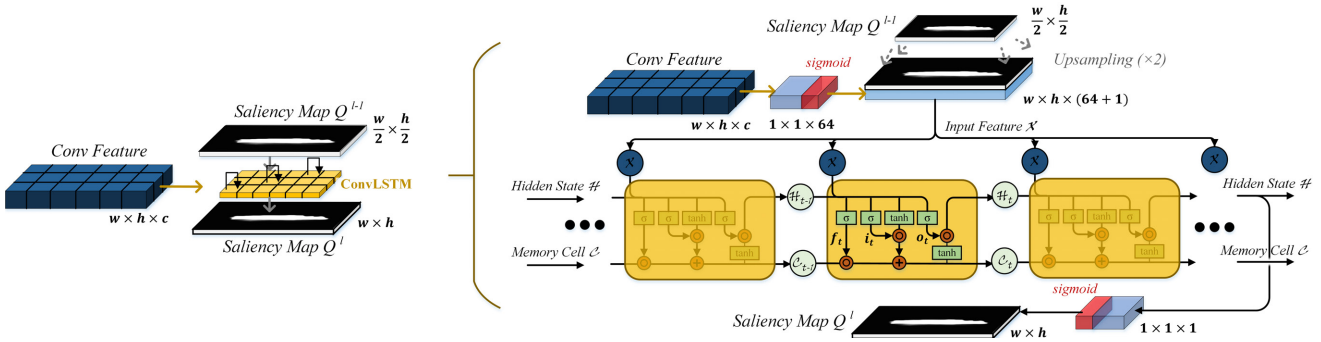


Figure 4. **Illustration of our convLSTM based object saliency optimization**, where (b) shows detailed architecture of our convLSTM optimization module in (a). Zoom-in for details.

precision, *recall*, *F-measure*, and *MAE* metrics for more efficient training. Given the ground-truth salient object annotation S (here $S \in \{0,1\}^{14\times14}$ for *conv5-3* layer), the overall loss function is defined as:

$$\begin{aligned}\mathcal{L}_{Sal}(S,Q) = \mathcal{L}_C(S,Q)+\alpha_1\mathcal{L}_P(S,Q)+\alpha_2\mathcal{L}_R(S,Q)\\ +\alpha_3\mathcal{L}_F(S,Q)+\alpha_4\mathcal{L}_{MAE}(S,Q),\end{aligned} \tag{7}$$

where αs are balance parameters and are empirically set as $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = 0.1$. $\mathcal{L}_C$ is the *weighted cross-entropy* loss function, which is widely adopted for training SOD models and opted as the primary loss in our case:

$$\begin{aligned}\mathcal{L}_C(S,Q) = \frac{1}{N}\sum\nolimits_x \big(\gamma\cdot(1-s_x)\cdot\log(1-q_x)\\ +(1-\gamma)\cdot s_x\cdot\log q_x\big),\end{aligned} \tag{8}$$

where N is the total number of pixels and $s_k \in S$, $q_k \in Q$. γ refers to the ratio of salient pixels in ground truth S. *Weighted cross-entropy* loss handles the imbalance between number of salient and non-salient pixels.

$\mathcal{L}_P$, $\mathcal{L}_R$ and $\mathcal{L}_F$ are computed similar to *precision*, *recall* and *F-measure* scores:

$$\mathcal{L}_P(S,Q) = -\sum\nolimits_x s_x\cdot q_x \Big/ (\sum\nolimits_x q_k + \epsilon), \tag{9}$$

$$\mathcal{L}_R(S,Q) = -\sum\nolimits_x s_x\cdot q_x \Big/ (\sum\nolimits_x s_k + \epsilon), \tag{10}$$

$$\mathcal{L}_F(S,Q) = -\frac{(1+\beta^2)\cdot\mathcal{L}_P(S,Q)\cdot\mathcal{L}_R(S,Q)}{\beta^2\cdot\mathcal{L}_P(S,Q)+\mathcal{L}_R(S,Q)+\epsilon}. \tag{11}$$

where $\beta^2 = 0.3$ as suggested by [1], and ϵ is a regularization constant. Since *precision*, *recall* and *F-measure* are similarity metrics and higher values are better, negative values are used for minimizing.

$\mathcal{L}_{MAE}$ is derived from the *mean absolute error* (*MAE*) measure that computes the discrepancy between the saliency map Q and the ground-truth map S:

$$\mathcal{L}_{MAE}(S,Q) = \frac{1}{N}\sum\nolimits_x |s_x - q_x|. \tag{12}$$

After obtaining the object saliency map $Q \in [0,1]^{14\times14}$ inferred from the fixation map P, we upsample ($\times2$) Q and feed it to the next convLSTM with the compressed features ($28\times28\times64$) from *conv4-3* layer for more detailed refinement. Above process is iteratively applied for *conv4-3*, *conv3-3*, *conv2-2* and *conv1-2* layers, respectively. Finally, the ASNet outputs a high-quality object saliency mask ($224\times224\times1$). In sum, the ASNet is able to effectively infer the object saliency thanks to 1) a learnable fixation prior, 2) iteratively updating saliency features with recurrent architecture and 3) efficiently merging spatially rich information from lower layers in a top-down manner.

3.3. Implementation Details

Overall loss: Let $\mathcal{I} = \{I_k, k = 1,\ldots,K\}$ denote all the training images (resized into 224×224) borrowed from exiting SOD or FP datasets. Since there are only few datasets that offer annotations for both SOD and FP tasks, most of the training images are either labeled with human fixation annotation or object saliency mask. Let $y_k^A \in \{0,1\}$ and $y_k^S \in \{0,1\}$ indicate whether we have the attention annotation G_k and object saliency mask S_k for the k-th training image. Our final loss function can be expressed as:

$$\mathcal{L} = \sum_{k=1}^{K} y_k^A\cdot\mathcal{L}_{Att}(G_k,P_k) + \sum_{k=1}^{K} y_k^S\cdot\sum_{\ell=1}^{5}\mathcal{L}_{Sal}(S_k^\ell, Q_k^\ell), \tag{13}$$

where the loss functions $\mathcal{L}_{Att}$ and $\mathcal{L}_{Sal}$ are defined in Eqn. 1 and Eqn. 7, respectively. The indicators y_k^A and y_k^S are employed to remedy missing ground truth in corresponding tasks. That is the error is not propagated back when the annotations are not offered. The $\ell \in \{1,\ldots,5\}$ refers to the ℓ-th convLSTM with *conv-*ℓ block in ASNet. With the hierarchical loss functions, each layer in ASNet has direct access to the gradients from the loss function leading to an implicit deep supervision [29]. We set the time steps to three in our convLSTM and employ 3×3 kernels for convolution operations.

	Dataset	#Images	Annotation		Resolution
			FP	SOD	
Training	SALICON [22]	15,000	✓		640×480
	THUS10K [10]	10,000		✓	$\max(w,h) = 400$
	DUT-OMRON [61]	5,168	✓	✓	$\max(w,h) = 400$
Testing	PASCAL-S[34]	850	✓	✓	$\max(w,h) = 500$
	MIT1003[23]	1,004	✓		$\max(w,h) = 1024$
	ECCSD[60]	1,000		✓	$\max(w,h) = 400$
	HKU-IS[31]	4,447		✓	$\max(w,h) = 400$

Table 1. **Datasets used for training and testing the ASNet.**

Training datasets: Another advantage of ASNet is that it can use data from both SOD and FP benchmarks. We consider three large-scale saliency datasets: SALICON [22], THUS10K [10], and DUT-OMRON [61]. The SALICON dataset is widely used in the domain of FP, while the THUS10K dataset is commonly used for SOD. These two datasets have annotations for fixations and salient objects, respectively. We further utilize the DUT-OMRON dataset which offers annotations for both FP and SOD. Detailed descriptions of employed datasets can be found in Table 1.

Training settings: In each training iteration, we use a mini-batch of 10 images, which are sampled from above three datasets and ensure data balance between SOD and FP. Data augmentation techniques (*e.g.*, flipping, rotation) are also adopted. Our model is implemented in Python on Keras, and trained with the Adam optimizer [24]. During the training phase, the learning rate is set to 0.0001 and is decreased by a factor of 10 every two epochs. The networks were trained for 12 epochs with early stopping strategy.

Computation load: ASNet is simultaneously trained for FP and SOD tasks in an end-to-end manner. The entire training procedure takes about 10 hours with a NVIDIA TITAN X GPU and a 4.0GHz Intel processor. It takes about 0.08s to process an image of 224×224 size, which is faster than most deep learning based competitors.

4. Experimental results

In this section, we first examine the performance of ASNet for the FP task. The goal of this experiment is to investigate the effectiveness of the learned fixation map prior, instead of comparing it with the state-of-the-art FP models. Then we evaluate the performance of the ASNet for the primary SOD task. Finally, an ablation study is performed to gain a deeper insight into the proposed ASNet.

Testing datasets: Four datasets including PASCAL-S [34], MIT1003 [23], ECCSD [60] and HKU-IS [31] are used for testing our model. PASCAL-S offers both annotations for FP and SOD, MIT1003 is a representative benchmark for FP, and ECCSD and HKU-IS are two typical datasets which are widely used for SOD. We report the evaluation results for both tasks over the PASCAL-S dataset. More details of above datasets can be found in Table 1.

Evaluation metrics: For the FP task, there are several ways to measure the agreement between model predictions and human eye movements. Here, we employ five typical metrics, namely Normalized Scanpath Saliency (NSS), Similarity Metric (SIM), Linear Correlation Coefficient (CC), AUC-Judd, and shuffled AUC. Please refer to [5, 51] for detailed descriptions of these metrics. For the SOD task, three standard metrics, namely PR-curve, F-measure, and MAE, are employed for evaluation. See [55] for details.

Methods	AUC-Judd ↑	SIM ↑	shuffled AUC ↑	CC ↑	NSS ↑
Mr-CNN [36]	0.80	0.35	0.73	0.38	1.36
SALICON [18]	0.85	0.42	0.74	0.53	1.86
Shallow-Net [43]	-	-	0.68	-	1.60
Deep-Net [43]	0.86	0.40	0.73	0.51	1.73
SU [26]	-	-	0.73	-	2.08
eDN [46]	0.85	0.30	0.66	0.41	1.29
BMS [62]	0.79	0.33	0.69	0.36	1.25
AIM [8]	0.79	0.27	0.68	0.26	0.82
GBVS [13]	0.83	0.36	0.66	0.42	1.38
ITTI [19]	0.77	0.32	0.66	0.33	1.10
ASNet-14×14	0.87	0.49	0.73	0.60	2.01
ASNet-28×28	0.88	0.52	0.75	0.65	2.30

- The authors in [26, 43] have not released detailed results.

Table 2. **Quantitative comparison of different FP models on the MIT1003 [23] dataset.**

Methods	AUC-Judd ↑	SIM ↑	shuffled AUC ↑	CC ↑	NSS ↑
Mr-CNN [36]	0.79	0.34	0.71	0.40	1.35
SALICON [18]	0.86	0.46	0.72	0.58	1.88
Shallow-Net [43]	-	-	0.69	-	1.90
Deep-Net [43]	0.87	0.42	0.71	0.55	1.74
SU [26]	-	-	0.73	-	2.22
eDN [46]	-	-	0.65	-	1.42
BMS [62]	-	-	0.67	-	1.28
AIM [8]	0.77	0.30	0.65	0.32	0.97
GBVS [13]	0.84	0.36	0.65	0.45	1.36
ITTI [19]	0.82	0.36	0.64	0.42	1.30
ASNet-14×14	0.90	0.55	0.74	0.70	2.26
ASNet-28×28	0.90	0.59	0.74	0.73	2.43

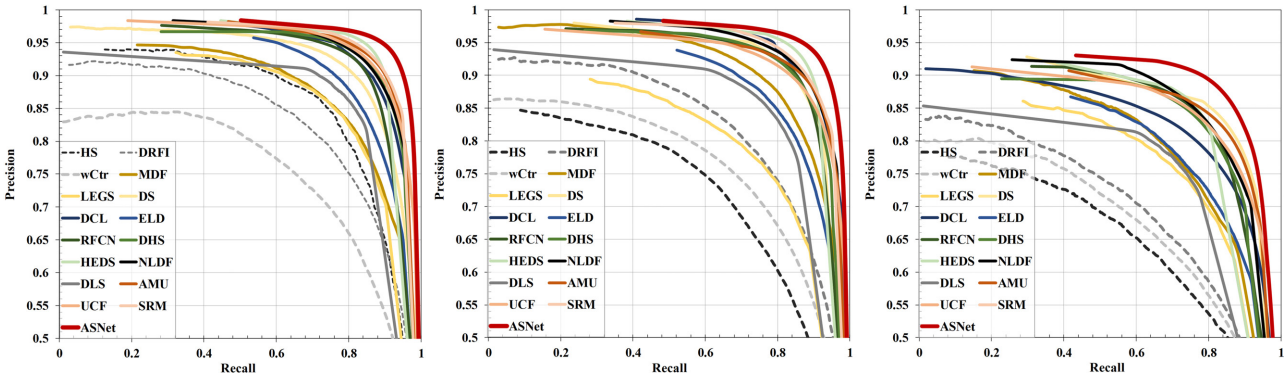
- The authors in [43, 26, 46, 62] have not released detailed results.

Table 3. **Quantitative comparison of different FP models on the PASCAL-S [23] dataset.**

4.1. Performance of ASNet

Performance on FP task: We evaluated the fixation prior map generated by ASNet compared to 10 state-of-the-art fixation models, including 4 classical models: ITTI [19], GBVS [13], AIM [8], BMS [62], and 6 deep learning based models: eDN [46], SALICON [18], SU [26], Mr-CNN [36], Shallow-Net [43] and Deep-Net[43]. Results are reported over PASCAL-S [34] and MIT1003 [23] datasets.

Our ASNet is able to generate a fixation prediction map P from top layer, which is relatively rough, and much smaller (only 14×14) compared to exiting fixation models. For the sake of a fairer comparison and deeper insight into the advantage of our ASNet, we further generate a larger fixation map (28×28) via feeding P into an additional con-



(a) ECCSD (b) HKU-IS (c) PASCAL-S

Figure 5. **SOD results with PR-curve on three widely used benchmarks**: ECCSD [60], HKU-IS [31] and PASCAL-S [34], where the scores from non-deep learning models are indicated by dashed lines. Best viewed in color.

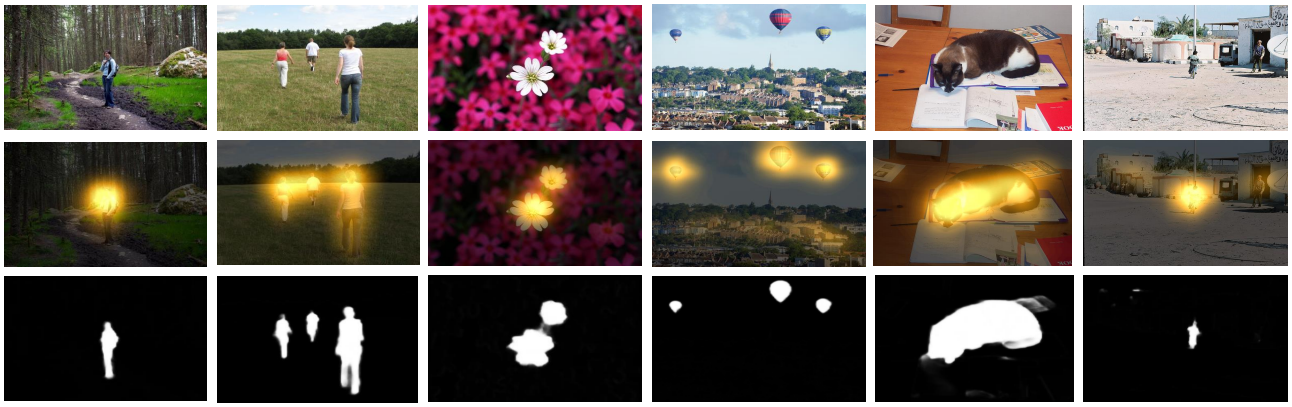


Figure 6. **Qualitative results of ASNet.** From top to bottom: example images, fixation maps, and object saliency results.

vLSTM with the features from *conv4-3* layer. We therefore derive two baselines: ASNet-14×14 and ASNet-28×28, corresponding to the rough prediction and the refined attention map with a more detailed spatial information.

As shown in Table 2 and Table 3, ASNet-14×14 performs better than previous non-deep learning models and is competitive with current top-performing deep learning contenders. Considering our relatively simple network architecture, and smaller output resolution (14×14), the suggested ASNet is much favorable and effective. We attribute this primarily to the extra generality and powerful saliency representations learned from the SOD task. Additionally, ASNet-28×28 produces further better results, which demonstrates the proposed ASNet has potential of obtaining better FP results with considering more detailed spatial information.

Performance on SOD task: Here we evaluate the performance of ASNet on its primary task: *SOD*. We perform quantitative study on 3 widely used datasets, namely ECCSD [60], HKU-IS [31] and PASCAL-S [34]. We compare ASNet against 14 recent deep learning based alternatives: LEGS [47], MDF [31], DS [33], SU [26], DCL [32], ELD [30], RFCN [48], DHS [35], HEDS [15], NLDF [38], DLS [17], AMU [64], UCF [65], and SRM [49]. We also

Methods	ECCSD [60]		HKU-IS [31]		PASCAL-S [34]	
	F-score↑	MAE↓	F-score↑	MAE↓	F-score↑	MAE↓
HS [60]	0.730	0.223	0.710	0.215	0.636	0.259
DRFI [21]	0.787	0.166	0.783	0.143	0.692	0.196
wCtr [67]	0.672	0.178	0.694	0.138	0.611	0.193
MDF [31]	0.831	0.108	0.860*	0.129*	0.764	0.145
LEGS [47]	0.831	0.119	0.812	0.101	0.749	0.155
DS [33]	0.810	0.160	0.848	0.078	0.818	0.170
DCL [32]	0.898	0.071	0.907	0.048	0.822	0.108
ELD [30]	0.865	0.080	0.844	0.071	0.767	0.121
SU [26]	0.88	0.06	-	-	0.77	0.10
RFCN [48]	0.898	0.097	0.895	0.079	0.827	0.118
DHS [35]	0.905	0.061	0.892	0.052	0.820	0.091
HEDS [15]	0.915	0.052	0.913	0.039	0.830	0.080
NLDF [38]	0.905	0.063	0.902	0.048	0.831	0.099
DLS [17]	0.825	0.090	0.806	0.072	0.719	0.136
AMU [64]	0.889	0.058	0.918	0.052	0.834	0.098
UCF [65]	0.868	0.068	0.905	0.062	0.771	0.116
SRM [49]	0.910	0.056	0.892	0.046	0.783	0.127
ASNet	0.928	0.043	0.920	0.035	0.857	0.072

- The authors in [26] have not released detailed results.
* MDF [31] is trained on a subset of HKU-IS, and evaluated on the remaining images.

Table 4. **The F-measure and MAE scores of SOD on three popular datasets**. ASNet gains the best performance with the assistance of visual attention prior. See text for details.

Aspects	Methods	ECCSD [60]		HKU-IS [31]		PASCAL-S [34]	
		F-score↑	MAE↓	F-score↑	MAE↓	F-score↑	MAE↓
	ASNet (conv1-output)	0.928	0.043	0.920	0.035	0.857	0.072
variants	w/o fixation	0.913	0.051	0.915	0.040	0.831	0.083
	w/o convLSTM	0.891	0.068	0.887	0.069	0.797	0.112
architecture	conv5-output	0.853	0.093	0.830	0.079	0.739	0.117
	conv4-output	0.875	0.076	0.844	0.058	0.749	0.092
	conv3-output	0.903	0.061	0.892	0.049	0.794	0.086
	conv2-output	0.919	0.049	0.912	0.040	0.847	0.078
loss	w/o $\mathcal{L}_P$	0.923	0.045	0.917	0.038	0.852	0.075
	w/o $\mathcal{L}_R$	0.924	0.046	0.915	0.039	0.854	0.074
	w/o $\mathcal{L}_F$	0.924	0.047	0.916	0.040	0.854	0.074
	w/o $\mathcal{L}_{MAE}$	0.921	0.044	0.914	0.037	0.850	0.072
	w/o extra loss	0.917	0.048	0.912	0.040	0.847	0.075

Table 5. **Ablation study of ASNet.** We change one component at a time, to assess individual contributions. See § 4.2 for details.

consider 3 classical non-deep learning models: HS [60], DRFI [21] and wCtr [67]. The results are obtained from the authors or by running their public implementations with original settings. The precision-recall curves of all methods are plotted in Fig. 5. As can be seen, the ASNet outperforms other competitors[2]. We report maximum F-measure and MAE scores in Table 4. Overall, the proposed method achieves better performance over three datasets using all evaluation metrics. Qualitative results over example images from above datasets are depicted in Fig. 6; showing that the proposed ASNet is well applicable to various complex scenes.

4.2. Validation of the Proposed Algorithm

We now conduct a more detailed examination of our proposed approach. We assess 1) contribution of the fixation prior for the SOD task, 2) the effects of convLSTM architecture, 3) the influence of stacked convLSTMs structure, and 4) the importance of the introduced loss functions.

1. Does fixation prior contribute to SOD? To answer this question, we directly remove the fixation prediction layer and the corresponding loss function $\mathcal{L}_{Att}$ in Eqn. 13. Then, we retrain ASNet with SOD data and obtain a baseline: *w/o fixation*. From Table 4, we find that fixation map is indeed informative to SOD over all three datasets. The improvement is more pronounced on PASCAL-S [34] dataset, which is collected from PASCAL challenge with more general scenes and less center-bias. These results demonstrate that a strong correlation exists between SOD and FP tasks, and our ASNet achieves better performance with the guidance from the fixation map. This also demonstrate that the leverage of large-scale FP data could improve the generalization ability of ASNet.

[2]Here we do not include the results from SU [26], since the authors have not released code or PR-curve results.

2. What is the effect of convLSTM? Here, we study the contribution of the convLSTM architecture, which constitutes a building block of our ASNet. To this end, we replace the convLSTMs with 5 convolution layers, which have 3×3 kernels and inputs/outputs with original dimensions. Thus we have a baseline: *w/o convLSTM*. Such network has similar architecture with previous bottom-up/top-down deep learning models [44, 35]. From Table 4, we observe a drop in F-score and MAE scores over three datasets which implies the effectiveness of the convLSTM.

3. Is the hierarchical architecture meaningful? We also study the effect of our hierarchical architecture with a stack of several convLSTMs and top-down saliency inference. We test 4 baselines: *conv5-output*, *conv4-output*, *conv3-output*, and *conv2-output*, which correspond to the outputs from the intermediate layers of ASNet. Note that the final prediction of ASNEt can be viewed as the output from $conv1$ layer. We find that the saliency results are gradually optimized by adding more details from lower layers.

4. Are the extra loss functions necessary? ASNet is equipped with 4 extra loss functions: $\mathcal{L}_P$, $\mathcal{L}_R$, $\mathcal{L}_F$, $\mathcal{L}_{MAE}$, which are derived from four widely used SOD metrics. For testing their effects, we retrain ASNet with different loss functions separately and thus we have four baselines: $w/o\ \mathcal{L}_P$, $w/o\ \mathcal{L}_R$, $w/o\ \mathcal{L}_F$, $w/o\ \mathcal{L}_{MAE}$. Another baseline $w/o\ extra\ loss$ indicates the results without considering all the extra loss functions. We show their F-measure and MAE scores in Table 5. We observe that those loss functions would boost the final performance with about 1% improvement in F-score.

5. Conclusions

We proposed a deep learning network, ASNet, towards a better interpretable and efficient SOD model, which leverages fixation prediction for detecting salient objects. The fixation map, as high-level knowledge of a scene, was learned from upper layers of ASNet. Such prior was further utilized for teaching the network where the salient object is and the detailed object saliency was rendered step by step by considering finer and finer features in a top-down manner. ConvLSTM was equipped for iteratively dropping useless features and enhancing the features for better representation. A set of loss functions derived from SOD metrics were introduced for boosting model predictions. Extensive experimental results demonstrate that our approach outperforms state-of-the-art methods and confirm our view that fixation map is valuable and indicative for SOD.

This paper points out two potential directions of following works. The first one regards exploring the rationale behind SOD from the fixation prediction viewpoint. The second one is to seek better loss functions for boosting the performance of deep learning based SOD models.

References

[1] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk. Frequency-tuned salient region detection. In *CVPR*, 2009. 2, 5

[2] B. Alexe, T. Deselaers, and V. Ferrari. Measuring the objectness of image windows. *IEEE TPAMI*, 34(11):2189–2202, 2012. 1

[3] A. Borji. What is a salient object? A dataset and a baseline model for salient object detection. *IEEE TIP*, 24(2):742–756, 2015. 1

[4] A. Borji, M.-M. Cheng, H. Jiang, and J. Li. Salient object detection: A benchmark. *IEEE TIP*, 24(12):5706–5722, 2015. 1, 2

[5] A. Borji and L. Itti. State-of-the-art in visual attention modeling. *IEEE TPAMI*, 35(1):185–207, 2013. 2, 6

[6] A. Borji, D. N. Sihite, and L. Itti. What stands out in a scene? A study of human explicit saliency judgment. *Vision research*, 91:62–77, 2013. 1

[7] N. Bruce and J. Tsotsos. Saliency based on information maximization. In *NIPS*, 2006. 2

[8] N. D. Bruce and J. K. Tsotsos. Saliency, attention, and visual search: An information theoretic approach. *Journal of Vision*, 9(3):5–5, 2009. 6

[9] 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 *ICCV*, 2017. 3

[10] M.-M. Cheng, N. J. Mitra, X. Huang, P. H. Torr, and S.-M. Hu. Global contrast based salient region detection. *IEEE TPAMI*, 37(3):569–582, 2015. 1, 2, 6

[11] M. Cornia, L. Baraldi, G. Serra, and R. Cucchiara. Predicting human eye fixations via an lstm-based saliency attentive model. *arXiv preprint arXiv:1611.09571*, 2016. 2

[12] D. Gao and N. Vasconcelos. Discriminant saliency for visual recognition from cluttered scenes. In *NIPS*, 2005. 2

[13] J. Harel, C. Koch, and P. Perona. Graph-based visual saliency. In *NIPS*, 2007. 2, 6

[14] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997. 4

[15] Q. Hou, M.-M. Cheng, X. Hu, A. Borji, Z. Tu, and P. Torr. Deeply supervised salient object detection with short connections. In *CVPR*, 2017. 2, 3, 7

[16] X. Hou and L. Zhang. Saliency detection: A spectral residual approach. In *CVPR*, 2007. 2

[17] P. Hu, B. Shuai, J. Liu, and G. Wang. Deep level sets for salient object detection. In *CVPR*, 2017. 2, 3, 7

[18] X. Huang, C. Shen, X. Boix, and Q. Zhao. SALICON: Reducing the semantic gap in saliency prediction by adapting deep neural networks. In *ICCV*, 2015. 2, 3, 6

[19] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE TPAMI*, 20(11):1254–1259, 1998. 2, 6

[20] S. Jetley, N. Murray, and E. Vig. End-to-end saliency mapping via probability distribution prediction. In *CVPR*, 2016. 2, 3

[21] H. Jiang, J. Wang, Z. Yuan, Y. Wu, N. Zheng, and S. Li. Salient object detection: A discriminative regional feature integration approach. In *CVPR*, 2013. 2, 7, 8

[22] M. Jiang, S. Huang, J. Duan, and Q. Zhao. SALICON: Saliency in context. In *CVPR*, 2015. 6

[23] T. Judd, K. Ehinger, F. Durand, and A. Torralba. Learning to predict where humans look. In *ICCV*, 2009. 2, 6

[24] D. Kingma and J. Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 6

[25] S. S. Kruthiventi, K. Ayush, and R. V. Babu. Deepfix: A fully convolutional neural network for predicting human eye fixations. *IEEE TIP*, 2017. 2, 3

[26] 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 *CVPR*, 2016. 2, 3, 6, 7, 8

[27] J. Kuen, Z. Wang, and G. Wang. Recurrent attentional networks for saliency detection. In *CVPR*, 2016. 2

[28] O. Le Meur, P. Le Callet, D. Barba, and D. Thoreau. A coherent computational approach to model bottom-up visual attention. *IEEE TPAMI*, 28(5):802–817, 2006. 2

[29] C.-Y. Lee, S. Xie, P. Gallagher, Z. Zhang, and Z. Tu. Deeply-supervised nets. In *AISTATS*, 2015. 5

[30] G. Lee, Y.-W. Tai, and J. Kim. Deep saliency with encoded low level distance map and high level features. In *CVPR*, 2016. 2, 7

[31] G. Li and Y. Yu. Visual saliency based on multiscale deep features. In *CVPR*, 2015. 2, 3, 6, 7, 8

[32] G. Li and Y. Yu. Deep contrast learning for salient object detection. In *CVPR*, 2016. 2, 3, 7

[33] 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 TIP*, 25(8):3919 – 3930, 2016. 7

[34] Y. Li, X. Hou, C. Koch, J. M. Rehg, and A. L. Yuille. The secrets of salient object segmentation. In *CVPR*, 2014. 1, 2, 6, 7, 8

[35] N. Liu and J. Han. DHSNet: Deep hierarchical saliency network for salient object detection. In *CVPR*, 2016. 2, 3, 7, 8

[36] N. Liu, J. Han, T. Liu, and X. Li. Learning to predict eye fixations via multiresolution convolutional neural networks. *IEEE TNNLS*, 2016. 2, 3, 6

[37] T. Liu, J. Sun, N.-N. Zheng, X. Tang, and H.-Y. Shum. Learning to detect a salient object. In *CVPR*, 2007. 1, 2

[38] Z. Luo, A. Mishra, A. Achkar, J. Eichel, S. Li, and P.-M. Jodoin. Non-local deep features for salient object detection. In *CVPR*, 2017. 2, 3, 7

[39] C. M. Masciocchi, S. Mihalas, D. Parkhurst, and E. Niebur. Everyone knows what is interesting: Salient locations which should be fixated. *Journal of vision*, 9(11):25–25, 2009. 2

[40] S. Mathe and C. Sminchisescu. Dynamic eye movement datasets and learnt saliency models for visual action recognition. In *ECCV*. 2012. 2

[41] S. Mathe and C. Sminchisescu. Action from still image dataset and inverse optimal control to learn task specific visual scanpaths. In *NIPS*, 2013. 2

[42] S. Mathe and C. Sminchisescu. Actions in the eye: Dynamic gaze datasets and learnt saliency models for visual recognition. *IEEE TPAMI*, 37(7):1408–1424, 2015. 2

[43] J. Pan, E. Sayrol, X. Giro-i Nieto, K. McGuinness, and N. E. O'Connor. Shallow and deep convolutional networks for saliency prediction. In *CVPR*, 2016. 2, 3, 6

[44] P. O. Pinheiro, T.-Y. Lin, R. Collobert, and P. Dollár. Learning to refine object segments. In *ECCV*, 2016. 3, 8

[45] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 3

[46] E. Vig, M. Dorr, and D. Cox. Large-scale optimization of hierarchical features for saliency prediction in natural images. In *CVPR*, 2014. 2, 6

[47] L. Wang, H. Lu, X. Ruan, and M.-H. Yang. Deep networks for saliency detection via local estimation and global search. In *CVPR*, 2015. 2, 7

[48] L. Wang, L. Wang, H. Lu, P. Zhang, and X. Ruan. Saliency detection with recurrent fully convolutional networks. In *ECCV*, 2016. 2, 3, 7

[49] T. Wang, A. Borji, L. Zhang, P. Zhang, and H. Lu. A stagewise refinement model for detecting salient objects in images. In *ICCV*, 2017. 2, 7

[50] W. Wang and J. Shen. Deep cropping via attention box prediction and aesthetics assessment. In *ICCV*, 2017. 1

[51] W. Wang and J. Shen. Deep visual attention prediction. *IEEE TIP*, 27(5):2368–2378, 2018. 2, 3, 6

[52] W. Wang, J. Shen, and F. Porikli. Saliency-aware geodesic video object segmentation. In *CVPR*, 2015. 1

[53] W. Wang, J. Shen, and L. Shao. Consistent video saliency using local gradient flow optimization and global refinement. *IEEE TIP*, 24(11):4185–4196, 2015. 2

[54] W. Wang, J. Shen, and L. Shao. Video salient object detection via fully convolutional networks. *IEEE TIP*, 27(1):38–49, 2018. 2

[55] W. Wang, J. Shen, L. Shao, and F. Porikli. Correspondence driven saliency transfer. *IEEE TIP*, 25(11):5025–5034, 2016. 2, 6

[56] W. Wang, J. Shen, R. Yang, and F. Porikli. Saliency-aware video object segmentation. *IEEE TPAMI*, 40(1):20–33, 2018. 2

[57] W. Wang, J. Shen, Y. Yu, and K.-L. Ma. Stereoscopic thumbnail creation via efficient stereo saliency detection. *IEEE TVCG*, 23(8):2014–2027, 2017. 1

[58] Y. Wei, F. Wen, W. Zhu, and J. Sun. Geodesic saliency using background priors. *ECCV*, 2012. 2

[59] 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 *NIPS*, 2015. 2, 4

[60] Q. Yan, L. Xu, J. Shi, and J. Jia. Hierarchical saliency detection. In *CVPR*, 2013. 2, 6, 7, 8

[61] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang. Saliency detection via graph-based manifold ranking. In *CVPR*, 2013. 6

[62] J. Zhang and S. Sclaroff. Saliency detection: A boolean map approach. In *ICCV*, 2013. 6

[63] L. Zhang, M. H. Tong, T. K. Marks, H. Shan, and G. W. Cottrell. SUN: A bayesian framework for saliency using natural statistics. *Journal of vision*, 8(7):32–32, 2008. 2

[64] P. Zhang, D. Wang, H. Lu, H. Wang, and X. Ruan. Amulet: Aggregating multi-level convolutional features for salient object detection. In *ICCV*, 2017. 2, 3, 7

[65] P. Zhang, D. Wang, H. Lu, H. Wang, and B. Yin. Learning uncertain convolutional features for accurate saliency detection. In *ICCV*, 2017. 2, 3, 7

[66] R. Zhao, W. Ouyang, H. Li, and X. Wang. Saliency detection by multi-context deep learning. In *CVPR*, 2015. 2, 3

[67] W. Zhu, S. Liang, Y. Wei, and J. Sun. Saliency optimization from robust background detection. In *CVPR*, 2014. 2, 7, 8