Deep Networks for Saliency Detection via Local Estimation and Global Search

Lijun Wang[†], Huchuan Lu[†], Xiang Ruan[‡] and Ming-Hsuan Yang[§]
[†]Dalian University of Technology [‡]OMRON Corporation [§]University of California at Merced

Abstract

*This paper presents a saliency detection algorithm by integrating both local estimation and global search. In the local estimation stage, we detect local saliency by using a deep neural network (**DNN-L**) which learns local patch features to determine the saliency value of each pixel. The estimated local saliency maps are further refined by exploring the high level object concepts. In the global search stage, the local saliency map together with global contrast and geometric information are used as global features to describe a set of object candidate regions. Another deep neural network (**DNN-G**) is trained to predict the saliency score of each object region based on the global features. The final saliency map is generated by a weighted sum of salient object regions. Our method presents two interesting insights. First, local features learned by a supervised scheme can effectively capture local contrast, texture and shape information for saliency detection. Second, the complex relationship between different global saliency cues can be captured by deep networks and exploited principally rather than heuristically. Quantitative and qualitative experiments on several benchmark data sets demonstrate that our algorithm performs favorably against the state-of-the-art methods.*

1. Introduction

Saliency detection, which aims to identify the most important and conspicuous object regions in an image, has received increasingly more interest in recent years. Serving as a preprocessing step, it can efficiently focus on the interesting image regions related to the current task and broadly facilitates computer vision applications such as segmentation, image classification, and compression, to name a few. Although much progress has been made, it remains a challenging problem.

Existing methods mainly formulate saliency detection by a computational model in a bottom-up fashion with either a local or a global view. Local methods [13, 25, 19, 39] compute center-surround differences in a local context for color, texture and edge orientation channels to capture the regions locally standing out from their surroundings. Although being biologically plausible, local models often lack global information and tend to highlight the boundaries of salient objects rather than the interiors (See Figure 1(c)). In contrast, global methods [1, 24, 29] take the entire image into consideration to predict the salient regions which are characterized by holistic rarity and uniqueness, and thus help detect large objects and uniformly assign saliency values to the contained regions. Unlike local methods which are sensitive to high frequency image contents like edges and noise, global methods are less effective when the textured regions of salient objects are similar to the background (See Figure 1(d)). The combination of local and global methods has been explored by a few recent studies, where background prior, center prior, color histograms and other handcrafted features are utilized in a simple and heuristic way to compute saliency maps.

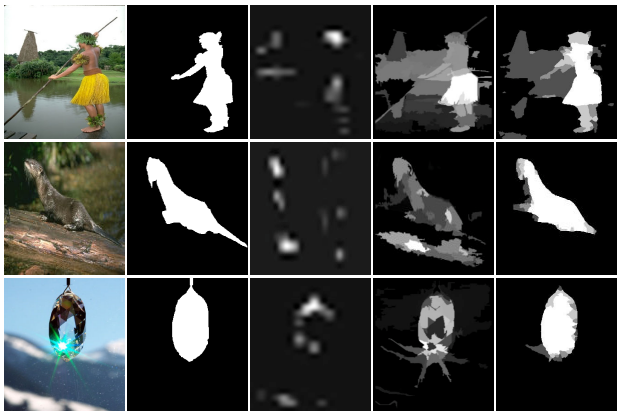


Figure 1. Saliency detection by different methods. (a) Original images. (b) Ground truth saliency maps. (c) Saliency maps by a local method [13]. (d) Saliency maps by a global method [7]. (e) Saliency maps by the proposed method.

While the combination of local and global models [32, 36] is technically sound, these methods have two major drawbacks. First, these methods mainly rely on handcrafted features which may fail to describe complex image scenarios and object structures. Second, the adopted saliency priors and features are mostly combined based on

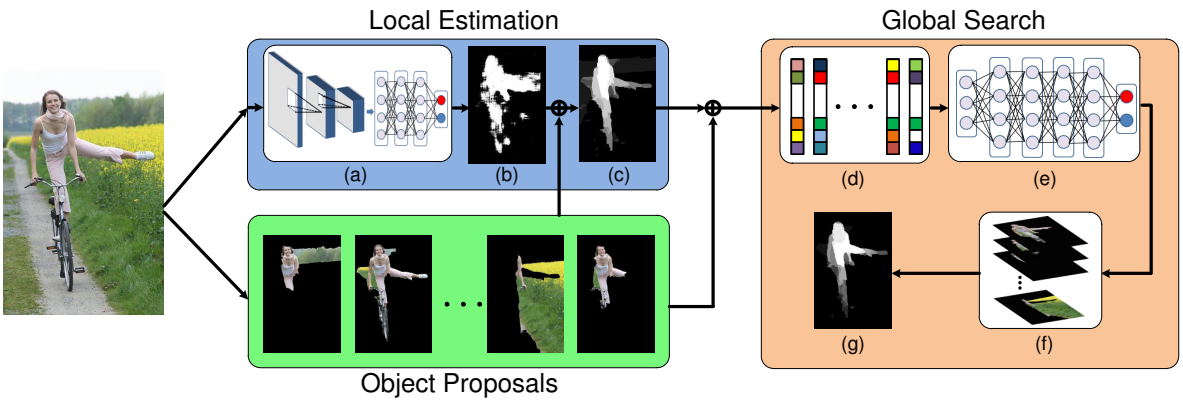


Figure 2. Pipeline of our algorithm. (a) Proposed deep network **DNN-L** (Section 3.1). (b) Local saliency map (Section 3.1). (c) Local saliency map after refinement (Section 3.2). (d) Feature extraction (Section 4.1). (e) Proposed deep network **DNN-G** (Section 4.2). (f) Sorted object candidate regions (Section 4.2). (g) Final saliency map (Section 4.2).

heuristics and it is not clear how these features can be better integrated.

In this paper, we propose a novel saliency detection algorithm by combining local estimation and global search (LEGS) to address the above-mentioned issues. In the local estimation stage, we formulate a deep neural network (DNN) based saliency detection method to assign a local saliency value to each pixel by considering its local context. The trained deep neural network, named as **DNN-L**, takes raw pixels as inputs and learns the contrast, texture and shape information of local image patches. The saliency maps generated by **DNN-L** are further refined by exploring the high level objectness (*i.e*., generic visual information of objects) to ensure label consistency and serve as local saliency measurements. In the global search stage, we search for the most salient object regions. A set of candidate object regions are first generated using a generic object proposal method [20]. A feature vector containing global color contrast, geometric information as well as the local saliency measurements estimated by **DNN-L** is collected to describe each object candidate region. These extracted feature vectors are used to train another deep neural network, **DNN-G**, to predict the saliency value of each object candidate region from a global perspective. The final saliency map is generated by the sum of salient object regions weighted by their saliency values. Figure 2 shows the pipeline of our algorithm.

Much success has been demonstrated by deep networks in image classification, object detection, and scene parsing. However, the use of DNNs in saliency detection is still limited, since DNNs, mainly fed with image patches, fail to capture the global relationship of image regions and maintain label consistency in a local neighborhood. Our main contribution addresses these issues by proposing an approach to apply DNNs to saliency detection from both local and global perspectives. We demonstrate that the proposed **DNN-L** is capable of capturing the local contrast, texture as well as shape information, and predicting the saliency value of each pixel without the need for hand-crafted features. The proposed **DNN-G** can effectively detect global salient regions by using various saliency cues through a supervised learning scheme. Both **DNN-L** and **DNN-G** are trained on the same training data set (See Section 5.1 for details). Without additional training, our method generalizes well to the other data sets and performs well against the state-of-the-art approaches.

2. Related Work

In this section, we discuss the related saliency detection methods and their connection to generic object proposal methods. In addition, we also briefly review deep neural networks that are closely related to this work.

Saliency detection methods can be generally categorized as local and global schemes. Local methods measure saliency by computing local contrast and rarity. In the seminal work [13] by Itti *et al*., center-surround differences across multi-scales of image features are computed to detect local conspicuity. Ma and Zhang [25] utilize color contrast in a local neighborhood as a measure of saliency. In [11], the saliency values are measured by the equilibrium distribution of Markov chains over different feature maps. The methods that consider only local contexts tend to detect high frequency content and suppress the homogeneous regions inside salient objects. On the other hand, global methods detect saliency by using holistic contrast and color statistics of the entire image. Achanta *et al*. [1] estimate visual saliency by computing the color difference between each pixel with respect to its mean. Histograms based global contrast and spatial coherence are used in [7] to detect saliency. Liu *et al*. [24] propose a set of features from both local and global views, which are integrated by a conditional random field to generate a saliency map. In [29], two

Table 1. Architecture details of the proposed deep networks. C: convolutional layer; F: fully connected layer; R: ReLUs; L: local response normalization; D: dropout; S: softmax; Channels: the number of output feature maps; Input size: the spatial size of input feature maps.

	DNN-L						**DNN-G**					
Layer	1	2	3	4	5	6 (Output)	1	2	3	4	5	6 (Output)
Type	C+R+L	C+R	C+R	F+R+D	F+R+D	F+S	F+R+D	F+R+D	F+R+D	F+R+D	F+R	F
Channels	96	256	384	2048	2048	2	1024	2048	2048	1024	1024	2
Filter size	11x11	5x5	3x3	–	–	–	–	–	–	–	–	–
Pooling size	3x3	2x2	3x3	–	–	–	–	–	–	–	–	–
Polling stride	2x2	2x2	3x3	–	–	–	–	–	–	–	–	–
Input size	51x51	20x20	8x8	2x2	1x1	1x1	1x1	1x1	1x1	1x1	1x1	1x1

contrast measures based on the uniqueness and spatial distribution of regions are defined for saliency detection. To identify small high contrast regions, Yan *et al*. [40] propose a multi-layer approach to analyze saliency cues. A random forest based regression model is proposed in [16] to directly map regional feature vectors to saliency scores. Recently, Zhu *et al*. [42] present a background measurement scheme to utilize boundary prior for saliency detection. Although significant advances have been made, most of the above-mentioned methods integrate hand-crafted features heuristically to generate the final saliency map, and do not perform well on challenging images. In contrast, we utilize a deep network (**DNN-L**) to automatically learn features capturing local saliency, and learn the complex dependencies among global cues using another deep network (**DNN-G**).

Generic object detection (also known as object proposal) methods [3, 2, 37] aim at generating the locations of all category independent objects in an image and have attracted growing interest in recent years. Existing techniques propose object candidates by either measuring the objectness [2, 5] of an image window or grouping regions in a bottom-up process [37, 20]. The generated object candidates can significantly reduce the search space of category specific object detectors, which in turn helps other modules for recognition and other tasks. As such, generic object detection are closely related to salient object segmentation. In [2], saliency is utilized as objectness measurement to generate object candidates. Chang *et al*. [4] use a graphical model to exploit the relationship of objectness and saliency cues for salient object detection. In [23], a random forest model is trained to predict the saliency score of an object candidate. In this work, we propose a DNN-based saliency detection method combining both local saliency estimation and global salient object candidate search.

Deep neural networks have achieved state-of-the-art results in image classification [21, 8, 34], object detection [35, 10, 12] and scene parsing [9, 30]. The success stems from the expressibility and capacity of deep architectures that facilitates learning complex features and models to account for interacted relationships directly from training examples. Since DNNs mainly take image patches as inputs, they tend to fail in capturing long range label dependencies for scene parsing as well as saliency detection. To address this issue, Pinheiro and Collobert [30] use a recurrent convolutional neural network to consider large contexts. In [9], a DNN is applied in a multi-scale manner to learn hierarchical feature representations for scene labeling. We propose to utilize DNNs in both local and global perspectives for saliency detection, where the **DNN-L** estimates local saliency of each pixel and the **DNN-G** searches for salient object regions based on global features to enforce label dependencies.

3. Local Estimation

The motivation of local estimation is that local outliers, standing out from their neighbors with different colors or textures, tend to attract human attention. In order to detect these outliers from a local view, we formulate a binary classification problem to determine whether each pixel is salient (1) or non-salient (0) based on its surrounding. We use a deep network, namely **DNN-L**, to conduct classification since DNNs have demonstrated state-of-the-art performance in image classification and do not rely on hand-crafted features. By incorporating object level concepts into local estimation, we present a refinement method to enhance the spatial consistency of local saliency maps.

3.1. DNN based Local Saliency Estimation

Architecture of DNN-L. The proposed **DNN-L** consists of six layers, with three convolutional layers and three fully connected layers. Each layer contains learnable parameters and consists of a linear transformation followed by a nonlinear mapping, which is implemented by Rectified Linear Unites (ReLUs) [28] to accelerate the training process. Local response normalization is applied to the first layer to help generalization. Max pooling is applied to all the three convolutional layers for translational invariance. The dropout procedure is used after the first and the second fully connected layers to avoid overfitting. The network takes a RGB image patch of 51×51 pixels as an input, and exploits a softmax regression model as the output layer to generate the probabilities of the central pixel being salient and non-salient. The architecture details are listed in Table 1.

Training data. For each image in the training set (See also Section 5.1), we collect samples by cropping 51×51 RG-

B image patches in a sliding window fashion with a stride of 10 pixels. To label the training patches, we mainly consider the ground truth saliency values of their central pixels as well as the overlaps between the patches and the ground truth saliency mask. The patch $\mathbf{B}$ is labeled as a positive training example if i). the central pixel is salient, and ii). it sufficiently overlaps with the ground truth salient region $\mathbf{G}$: $|\mathbf{B} \bigcap \mathbf{G}| \geq 0.7 \times \min(|\mathbf{B}|, |\mathbf{G}|)$. Similarly, the patch $\mathbf{B}$ is labeled as a negative training example if i). the central pixel is located within the background, and ii). its overlap with the ground truth salient region is less than a predefined threshold: $|\mathbf{B} \bigcap \mathbf{G}| < 0.3 \times \min(|\mathbf{B}|, |\mathbf{G}|)$. The remaining samples labeled as neither positive nor negative are not used. Following [21], we do not pre-process the training samples, except for subtracting the mean values over the training set from each pixel.

Training DNN-L. Given the training patch set $\{\mathbf{B}_i\}_{N^L}$ and the corresponding label set $\{l_i\}_{N^L}$, we use the softmax loss with weight decay as the cost function,

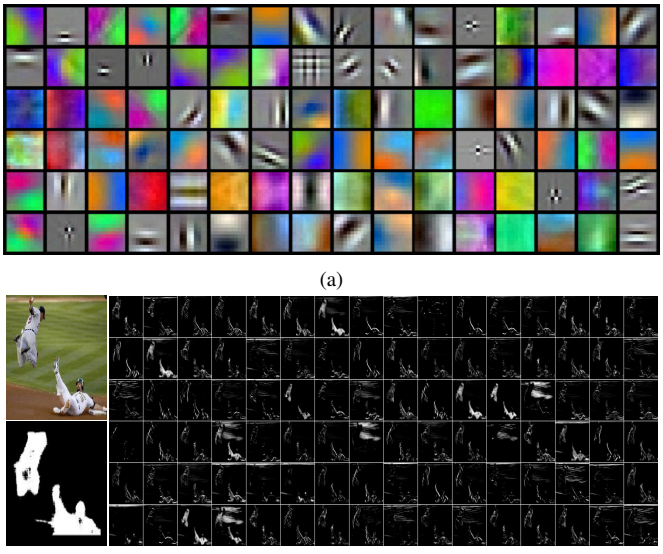
$$L(\boldsymbol{\theta}^L) = -\frac{1}{m}\sum_{i=1}^{m}\sum_{j=0}^{1}\mathbf{1}\{l_i = j\}\log P(l_i = j|\boldsymbol{\theta}^L) + \lambda\sum_{k=1}^{6}\|\mathbf{W}_k^L\|_F^2, \tag{1}$$

where $\boldsymbol{\theta}^L$ is the learnable parameter set of **DNN-L** including the weights and bias of all layers; $\mathbf{1}\{\cdot\}$ is the indicator function; $P(l_i = j|\boldsymbol{\theta}^L)$ is the label probability of the i-th training samples predicted by **DNN-L**; λ is the weight decay parameter; and $\mathbf{W}_k^L$ is the weight of the k-th layer. **DNN-L** is trained using stochastic gradient descent with a batch size of $m = 256$, momentum of 0.9, and weight decay of 0.0005. The learning rate is initially set to 0.01 and is decreased by a factor of 0.1 when the cost is stabilized. The training process is repeated for 80 epochs. Figure 3(a) illustrates the learned convolutional filters in the first layer, which capture color, contrast, edge and pattern information of a local neighborhood. Figure 3(c) shows the output of the first layer, where locally salient pixels with different features are highlighted by different feature maps.

At test stage, we apply **DNN-L** in a sliding window fashion to the entire image and predict the probability $P(l = 1|\boldsymbol{\theta})$ for each pixel as its local saliency value. Figure 4(c) demonstrates the generated local saliency maps. Both Figure 3 and Figure 4 show that the proposed local estimation method can effectively learn, rather than design, useful features characterizing local saliency by training **DNN-L** with local image patches.

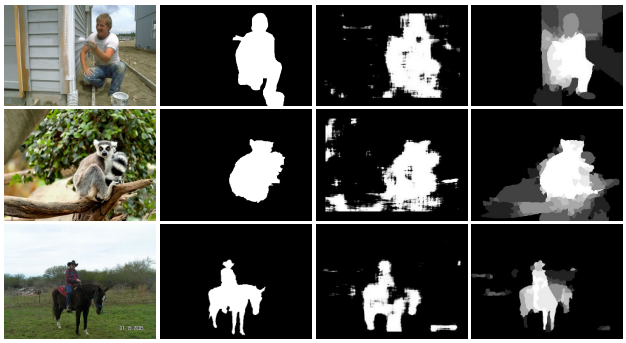
3.2. Refinement

The local estimation method detects saliency by considering the color, contrast and texture information within a neighborhood. Thus it may be sensitive to high frequency background noise and fail to maintain spatial consistency. On the other hand, saliency is closely correlated with the object-level concepts, *i.e.*, interesting objects easily attract human attention. Based on this observation, we propose to refine the local saliency map by combining low level saliency and high level objectness. To this end, we utilize the geodesic object proposal (GOP) [20] method to extract a set of object segments. The generated object candidates encode informative shape and boundary cues and serve as an over-complete coverage of the object in an image. Our method searches for a subset of these candidates with high probabilities to be the potential object according to the local salien-



(a) (b) (c)

Figure 3. Visualization of **DNN-L**. (a) 96 convolutional filters with the size of $11 \times 11 \times 3$ in the first layer. (b) Input image (top) and the local saliency map (bottom) generated by **DNN-L**. (c) Output feature maps of the first layer by applying **DNN-L** to the input image in a sliding window manner. (Better viewed at high resolution.)



(a) (b) (c) (d)

Figure 4. Saliency maps by local estimation. (a) Source images. (b) Ground truth. (c) Local saliency maps predicted by **DNN-L**. (d) Local saliency maps after refinement.

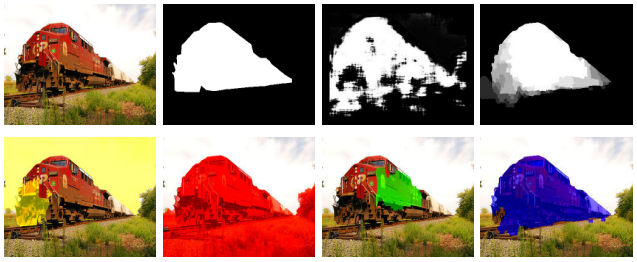


Figure 5. Top row (left to right): source image, ground truth, local saliency map output by **DNN-L**, local saliency map after refinement. Bottom row: different object candidate regions with their corresponding accuracy scores A and coverage scores C.

cy map, and thereby integrates local estimation and generic object proposals as a complementary process.

Given an input image, we first generate a set of object candidate masks $\{\mathbf{O}_i\}_{N_O}$ using the GOP method and a saliency map $\mathbf{S}^L$ using our local estimation method. To determine the confidence of each segment, we mainly consider two measurements based on the local saliency map, accuracy score A and coverage score C, defined by

$$A_i = \frac{\sum_{x,y} \mathbf{O}_i(x,y) \times \mathbf{S}^L(x,y)}{\sum_{x,y} \mathbf{O}_i(x,y)}, \tag{2}$$

$$C_i = \frac{\sum_{x,y} \mathbf{O}_i(x,y) \times \mathbf{S}^L(x,y)}{\sum_{x,y} \mathbf{S}^L(x,y)}, \tag{3}$$

where $\mathbf{O}_i(x,y) = 1$ indicates that the pixel located at (x,y) of the input image belongs to the i-th object candidate, and $\mathbf{O}_i(x,y) = 0$ otherwise; $\mathbf{S}^L(x,y) \in [0,1]$ represents the local saliency value of pixel (x,y).

The accuracy score A_i measures the average local saliency value of the i-th object candidate, whereas the coverage score C_i measures the proportion of salient area covered by the i-th object candidate. Figure 5 presents an intuitive example for interpreting these two measurements. The yellow candidate region having a small overlap with the local salient area is assigned with both a low accuracy score and a low coverage score. The red candidate region covering almost the entire local salient region has a high coverage score but a low accuracy score. The green candidate region located inside the local salient region has a high accuracy score but a low coverage score. Only the optimal blue candidate has a high accuracy score as well as a high coverage score. Based on the above observations, we define the confidence for the i-th candidate by considering both the accuracy score and the coverage score as

$$conf_i^L = \frac{(1+\beta) \times A_i \times C_i}{\beta A_i + C_i}, \tag{4}$$

where we set $\beta = 0.4$ to emphasize the impact of the accuracy score on the final confidence. To find a subset of optimal object candidates, we sort all the candidates by their confidences in a descending order. The refined local saliency map is generated by averaging the top K candidate regions (K is set to 20 in all the experiments). Figure 4 shows the local saliency maps before and after refinement.

4. Global Search

Saliency cues such as center and object bias [31, 22], contrast information [38] and background prior [33, 15] have been shown to be effective in previous work. However, these saliency cues are considered independently, and combined based on heuristics. For example, the background prior is utilized by treating all pixels within the boundary regions of an image as background without considering the color statistics of the entire image or the location of the foreground. Instead, we formulate a DNN-based regression method for saliency detection, where various saliency cues are considered simultaneously and their complex dependencies are learned automatically through a supervised learning scheme. For each input image, we first detect local saliency using the proposed local estimation method. A 72-dimensional feature vector is extracted to describe each object candidate generated by the GOP method from a global view. The proposed deep network **DNN-G** takes the extracted features as inputs and predicts the saliency values of the candidate regions through regression.

4.1. Global Features

The proposed 72-dimensional feature vector covers global contrast features, geometric information, and local saliency measurements of object candidate regions. Global contrast features consist of three components: boundary contrast, image statistic divergence and internal variance, which are computed in the RGB, Lab and HSV color spaces. Given an object candidate region $\mathbf{O}$ and using the RGB color space as an example, we compute its RGB histogram $h_{\mathbf{O}}^{RGB}$, mean RGB values $m_{\mathbf{O}}^{RGB}$, and RGB color variance $var_{\mathbf{O}}^{RGB}$ over all the pixels within the candidate region. We define the border regions of 15 pixels width in four directions of the image as boundary regions. Since the boundary regions in different directions may have different appearance, we compute their RGB histograms and mean RGB values separately. For representation con-

Table 2. Global contrast features of object candidate regions.

Feature	Definition	Feature	Definition
$c_1 - c_4$	$\chi^2(h_{\mathbf{O}}^{RGB}, h_{\mathbf{B}}^{RGB})$	c_{49}	$\chi^2(h_{\mathbf{O}}^{RGB}, h_{\mathbf{I}}^{RGB})$
$c_5 - c_8$	$\chi^2(h_{\mathbf{O}}^{Lab}, h_{\mathbf{B}}^{Lab})$	c_{50}	$\chi^2(h_{\mathbf{O}}^{Lab}, h_{\mathbf{I}}^{Lab})$
$c_9 - c_{12}$	$\chi^2(h_{\mathbf{O}}^{HSV}, h_{\mathbf{B}}^{HSV})$	c_{51}	$\chi^2(h_{\mathbf{O}}^{HSV}, h_{\mathbf{I}}^{HSV})$
$c_{13} - c_{24}$	$d(m_{\mathbf{O}}^{RGB}, m_{\mathbf{B}}^{RGB})$	$c_{52} - c_{54}$	$var_{\mathbf{O}}^{RGB}$
$c_{25} - c_{36}$	$d(m_{\mathbf{O}}^{HSV}, m_{\mathbf{B}}^{HSV})$	$c_{55} - c_{57}$	$var_{\mathbf{O}}^{Lab}$
$c_{37} - c_{48}$	$d(m_{\mathbf{O}}^{Lab}, m_{\mathbf{B}}^{Lab})$	$c_{58} - c_{60}$	$var_{\mathbf{O}}^{HSV}$

Table 3. Geometric information and local saliency measurements of object regions.

Geometric Information				Local Saliency Measurement	
Feature	Definition	Feature	Definition	Feature	Definition
g_1	Bounding box aspect ratio	g_6	Major axis length	s_1	Accuracy score A
g_2	Bounding box height	g_7	Minor axis length	s_2	Coverage score C
g_3	Bounding box width	g_8	Euler number	s_3	$A \times C$
$g_4 - g_5$	Centroid coordinates			s_4	Overlap rate

venience, we uniformly denote the RGB histograms and mean RGB values of the four boundary regions as $h_{\mathbf{B}}^{RGB}$ and $m_{\mathbf{B}}^{RGB}$, respectively. The RGB histogram of the entire image $h_{\mathbf{I}}^{RGB}$ is also used as an image statistic. The boundary contrast is measured by the chi-square distances $\chi^2(h_{\mathbf{O}}^{RGB}, h_{\mathbf{B}}^{RGB})$ between the RGB histograms of the candidate and the four boundary regions, and the Euclidean distances $d(m_{\mathbf{O}}^{RGB}, m_{\mathbf{B}}^{RGB})$ between their mean RGB values. The color divergence of the candidate region from the entire image statistic is measured by the chi-square distance $\chi^2(h_{\mathbf{O}}^{RGB}, h_{\mathbf{I}}^{RGB})$ between the RGB histograms of the candidate region and the entire image. The internal color variance of the candidate region is measured by the RGB color variance $var_{\mathbf{O}}^{RGB}$. The global contrast features in the Lab and HSV color spaces are extracted in a similar way. Table 2 summarizes the components of global contrast features.

Geometric information characterizes the spatial distribution of object candidates. We extract the centroid coordinates, major/minor axis length, Euler number[1] and the shape information of the enclosing bounding box including its width, height and aspect ratio. All the above features except the Euler number are normalized with respect to the input image size. Table 3 shows the details of the geometric information.

Local saliency measurements evaluate the saliency value of each candidate region based on the saliency map produced by the local estimation method. Given the refined local saliency map and the object candidate mask, we compute the accuracy score A and the coverage score C using (2)-(3). The overlap rate between the object mask and the local saliency map is also computed (See Table 3 for details).

4.2. Saliency Prediction via DNN-G Regression

The proposed **DNN-G** consists of 6 fully connected layers. Each layer carries out a linear transformation followed by ReLUs to accelerate the training process and the dropout operation to avoid overfitting (See Table 1). For each image in the training data set (Section 5.1), around 1200 object regions are generated as training samples using the GOP method. The proposed 72-dimensional global feature vector $\mathbf{v}$ is extracted from each candidate region and then preprocessed by subtracting the mean and dividing the standard deviation of the elements. Given the ground truth saliency map $\mathbf{G}$, a label vector of precision p_i and overlap rate o_i, $\mathbf{y}_i = [p_i, o_i]$, is assigned to each object region $\mathbf{O}_i$.

Given the training data set $\{\mathbf{v}_i\}_{N^G}$ and the corresponding label set $\{\mathbf{y}_i\}_{N^G}$, the network parameters of **DNN-G** are learned by solving the following optimization problem

$$\arg\min_{\boldsymbol{\theta}^G} \frac{1}{m} \sum_{i=1}^{m} \|\mathbf{y}_i - \boldsymbol{\phi}(\mathbf{v}_i|\boldsymbol{\theta}^G)\|_2^2 + \eta \sum_{k=1}^{6} \|\mathbf{W}_k^G\|_F^2, \quad (5)$$

where $\boldsymbol{\theta}^G$ is the network parameter set; $\boldsymbol{\phi}(\mathbf{v}_i|\boldsymbol{\theta}^G) = [\phi_i^1, \phi_i^2]$ is the output of **DNN-G** for the i-th training sample; $\mathbf{W}_k^G$ is the weight of the k-th layer; and η is the weight decay parameter which is set to 0.0005. The above optimization problem is solved by using stochastic gradient descent with a batch size m of 1000 and momentum of 0.9. The learning rate is initially set to 0.05 and is decreased by a factor of 0.5 when the cost is stabilized. The training process is repeated for 100 epochs.

At test stage, the network takes the feature vector of the i-th candidate region as an input and predicts its precision and overlap rate by $\boldsymbol{\phi}(\mathbf{v}_i|\boldsymbol{\theta}^G)$. The global confidence score of the candidate region is defined by

$$conf_i^G = \phi_i^1 \times \phi_i^2. \quad (6)$$

Denote $\{\hat{\mathbf{O}}_1, \ldots, \hat{\mathbf{O}}_N\}$ as the mask set of all the candidate regions in the input image sorted by the global confidence scores in a descending order. The corresponding global confidence scores are represented by $\{conf_1^G, \ldots, conf_N^G\}$. The final saliency map is computed by a weighted sum of the top K candidate masks,

$$\mathbf{S}^G = \frac{\sum_{k=1}^{K} conf_k^G \times \hat{\mathbf{O}}_k}{\sum_{k=1}^{K} conf_k^G}. \quad (7)$$

Although similar in spirit, our global search method is significantly different from [10], [16] and [23] in the following aspects: i). Our method utilizes DNNs to learn the complex dependencies among different visual cues and determines the saliency of a candidate region in a global view, whereas [10] applies DNN to a bounding box to extract category-specific features. ii). Both [16] and [23] use random forests to predict region saliency based on regional features, where [23] trains the model for each data set. In contrast, we use DNNs for saliency detection and conduct training in one

[1]The Euler number of an object mask is the total number of objects in the mask minus the total number of holes in those objects.

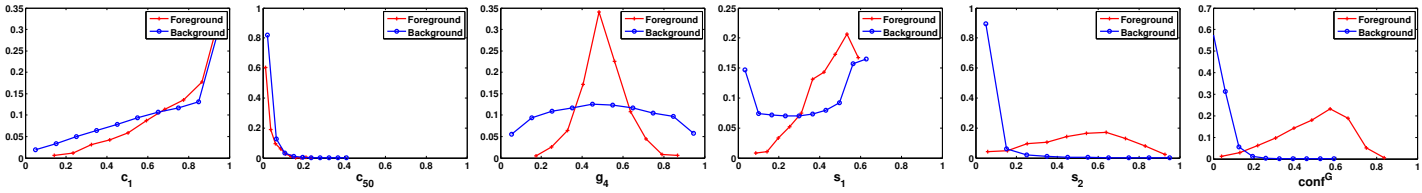


Figure 6. Distribution of foreground and background regions in different feature spaces, including global contrast features (c_1 and c_{50}), geometric information (g_4), local saliency measurements (s_1 and s_2) and the global confidence scores ($conf^G$) generated by **DNN-G**.

Table 4. Quantitative results using F-measure and MAE. The best and second best results are shown in **red** color and **blue** color.

Data Set	Metric	DRFI	GC	HS	MR	PCA	SVO	UFO	wCtr	CPMC-GBVS	HDCT	LEGS
SOD	F-Measure	**0.617**	0.433	0.480	0.542	0.498	0.217	0.521	0.567	–	0.511	**0.630**
	MAE	**0.230**	0.288	0.301	0.274	0.290	0.414	0.272	0.245	–	0.260	**0.205**
ECCSD	F-Measure	**0.726**	0.568	0.631	0.689	0.575	0.237	0.638	0.672	–	0.641	**0.775**
	MAE	**0.172**	0.218	0.232	0.192	0.252	0.406	0.210	0.178	–	0.204	**0.137**
PASCAL-S	F-Measure	0.619	0.496	0.536	0.600	0.531	0.266	0.552	0.611	**0.654**	0.536	**0.669**
	MAE	0.195	0.245	0.249	0.219	0.239	0.373	0.227	0.193	**0.178**	0.226	**0.170**
MSRA-5000	F-Measure	–	0.704	0.765	**0.789**	0.707	0.302	0.774	0.788	–	0.773	**0.803**
	MAE	–	0.149	0.160	0.130	0.189	0.364	0.145	**0.110**	–	0.141	**0.128**

data set (See Section 5.1). iii). Global search is integrated with local estimation in our work, which facilitates more robust saliency detection from both perspectives.

5. Experimental Results

5.1. Setup

We evaluate the proposed algorithm on four benchmark data sets: MSRA-5000 [24], SOD [27], ECCSD [40] and PASCAL-S [23]. The MSRA-5000 data set is widely used for saliency detection and covers a large variety of image contents. Most of the images include only one salient object with high contrast to the background. The SOD data set, containing 300 images, is collected from the Berkeley segmentation data base. Many images in this data set have multiple salient objects of various sizes and locations. The ECCSD data set contains 1000 images with complex scenes from the Internet and is more challenging. The newly developed PASCAL-S data set is constructed on the validation set of the PASCAL VOC 2012 segmentation challenge. This data set contains 850 natural images with multiple complex objects and cluttered backgrounds. The PASCAL-S data set is arguably one of the most challenging saliency data sets without various design biases (*e.g.*, center bias and color contrast bias). All the data sets contain manually annotated ground truth saliency maps.

Since the MSRA-5000 data set covers various scenarios and the PASCAL-S data set contains images with complex structures, we randomly sample 3000 images from the MSRA-5000 data set and 340 images from the PASCAL-S data set to train the proposed two networks. The remaining images are used for tests. Both horizontal reflection and rescaling ($\pm 5\%$) are applied to all the training images to augment the training data set. The DNNs are implemented using the Caffe framework [14]. The trained models and source code are available at our website[2].

We evaluate the performance using precision-recall (PR) curves, F-measure and mean absolute error (MAE). The precision and recall of a saliency map are computed by segmenting a salient region with a threshold, and comparing the binary map with the ground truth. The PR curves demonstrate the mean precision and recall of saliency maps at different thresholds. The F-measure is defined as $F_\gamma = \frac{(1+\gamma^2)Precision \times Recall}{\gamma^2 Precision + Recall}$, where $Precision$ and $Recall$ are obtained using twice the mean saliency value of saliency maps as the threshold, and γ^2 is set to 0.3. The MAE is the average per-pixel difference between saliency maps and the ground truth.

5.2. Feature Analysis

Our global search method exploits various saliency cues to describe each object candidate. We present an empirical analysis on the discriminative ability of all the global features based on the distribution of both foreground and background regions in different feature spaces. We generate 500000 object candidate regions using 510 test images from the PASCAL-S data set. Based on the overlap rate o_i with the ground truth salient region, the i-th candidate region is classified as foreground ($o_i > 0.7$) or background ($o_i < 0.2$). The remaining candidate regions ($0.2 \leq o_i \leq 0.7$) are left unused. Figure 6 illustrates the distribution of both foreground and background regions in three types of feature spaces discussed in Section 4.1 and the global confidence score space generated by **DNN-G**. More results can be found in the supplementary material.

The distribution plots in Figure 6 show strong overlap-

[2] http://ice.dlut.edu.cn/lu/index.html

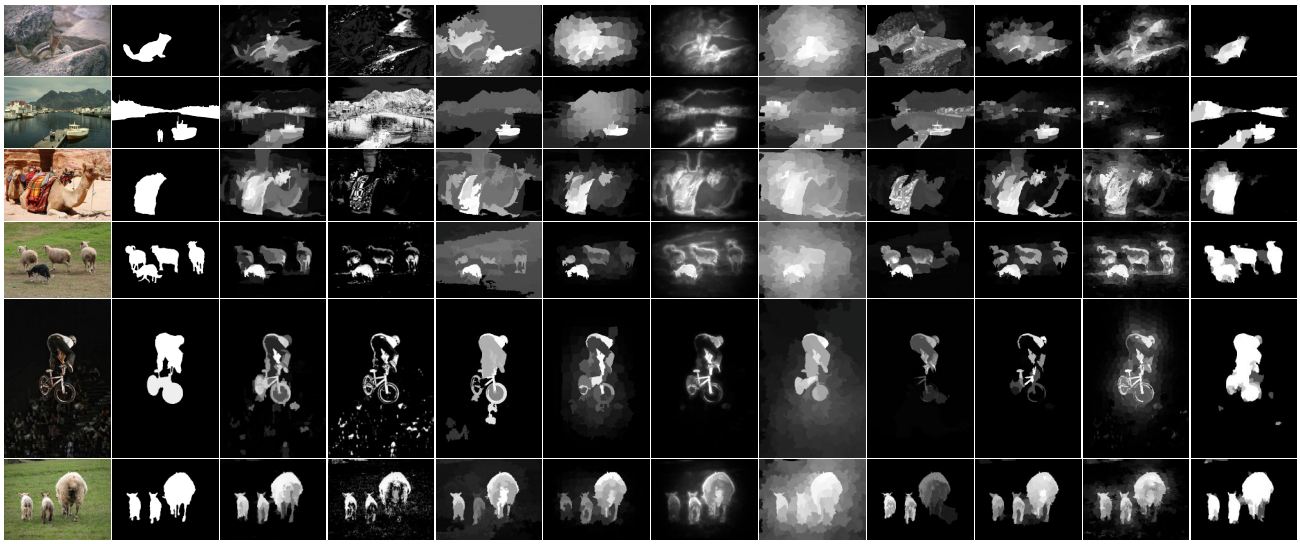
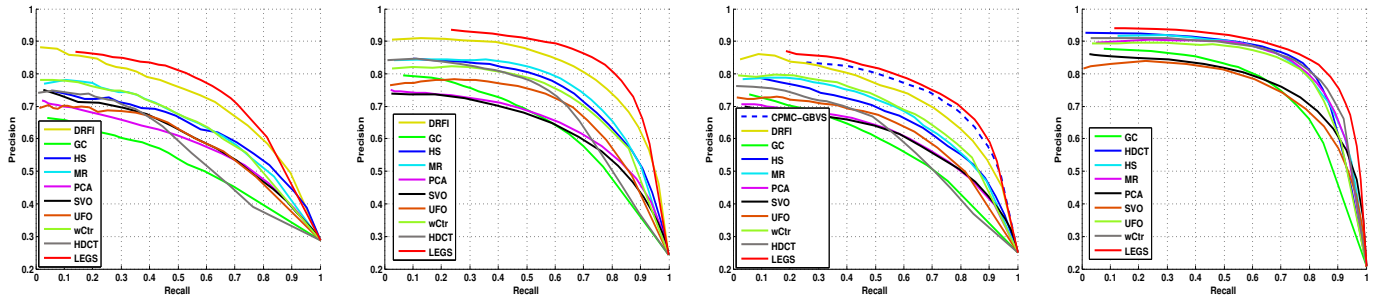


Figure 7. Saliency maps. Top, middle and bottom two rows are images from the SOD, ECCSD and PASCAL-S data sets. GT: ground truth.



(a) SOD (b) ECCSD (c) PASCAL-S (d) MSRA-5000

Figure 8. PR curves of saliency detection methods on four benchmark data sets.

s between foreground and background regions in all three types of feature spaces. Foreground and background regions can be hardly separated based on a heuristic combination of these features. Our global search method trains a deep network to learn complex feature dependencies and achieves accurate confidence scores for saliency detection.

5.3. Performance Comparison

We compare the proposed method (LEGS) with ten state-of-the-art models including SVO [4], PCA[26], DRFI [16], GC [6], HS [40], MR [41], UFO [17], wCtr [42], CPMC-GBVS [23] and HDCT [18]. We use either the implementations or the saliency maps provided by the authors for fair comparison[3]. Our method performs favorably against the state-of-the-art methods in terms of PR curves (Figure 8), F-measure as well as MAE scores (Table 4) in all three data sets. Figure 7 shows that our method generates more accurate saliency maps in various challenging scenarios. The robust performance of our method can be attributed to the use of DNNs for complex feature and model learning, and the integration of local/global saliency estimation.

[3]The result of the DRFI method [16] on the MSRA-5000 data set are not reported, since it is also trained on this data set with different training images from ours. The CPMC-GBVS method [23] only provides the saliency maps of the PASCAL-S data set.

6. Conclusions

In this paper, we propose DNNs for saliency detection by combining local estimation and global search. In the local estimation stage, the proposed **DNN-L** estimates local saliency by learning rich image patch features from local contrast, texture and shape information. In the global search stage, the proposed **DNN-G** effectively exploits the complex relationships among global saliency cues and predicts the saliency value for each object region. Our method integrates low level saliency and high level objectness through a supervised DNN-based learning schemes. Experimental results on benchmark data sets show that the proposed algorithm can achieve state-of-the-art performance.

Acknowledgements. L. Wang and H. Lu are supported by the Natural Science Foundation of China (NSFC) #61472060 and the Fundamental Research Funds for the Central Universities under Grant DUT14YQ101. M.-H. Yang is supported in part by NSF CAREER Grant #1149783 and NSF IIS Grant #1152576.

References

[1] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk. Frequency-tuned salient region detection. In *CVPR*, pages 1597–1604, 2009.

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

[3] J. Carreira and C. Sminchisescu. Constrained parametric min-cuts for automatic object segmentation. In *CVPR*, pages 3241–3248, 2010.

[4] K.-Y. Chang, T.-L. Liu, H.-T. Chen, and S.-H. Lai. Fusing generic objectness and visual saliency for salient object detection. In *ICCV*, pages 914–921, 2011.

[5] M. Cheng, Z. Zhang, W. Lin, and P. H. S. Torr. BING: binarized normed gradients for objectness estimation at 300fps. In *CVPR*, pages 3286–3293, 2014.

[6] M.-M. Cheng, J. Warrell, W.-Y. Lin, S. Zheng, V. Vineet, and N. Crook. Efficient salient region detection with soft image abstraction. In *ICCV*, pages 1529–1536, 2013.

[7] M.-M. Cheng, G.-X. Zhang, N. J. Mitra, X. Huang, and S.-M. Hu. Global contrast based salient region detection. In *CVPR*, pages 409–416, 2011.

[8] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *ICML*, pages 647–655, 2014.

[9] C. Farabet, C. Couprie, L. Najman, and Y. LeCun. Learning hierarchical features for scene labeling. *PAMI*, 35(8):1915–1929, 2013.

[10] R. B. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, pages 580–587, 2014.

[11] J. Harel, C. Koch, and P. Perona. Graph-based visual saliency. In *NIPS*, pages 545–552, 2006.

[12] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. Simultaneous detection and segmentation. In *ECCV*, pages 297–312. 2014.

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

[14] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint*, 2014.

[15] B. Jiang, L. Zhang, H. Lu, C. Yang, and M.-H. Yang. Saliency detection via absorbing markov chain. In *ICCV*, pages 1665–1672, 2013.

[16] H. Jiang, J. Wang, Z. Yuan, Y. Wu, N. Zheng, and S. Li. Salient object detection: A discriminative regional feature integration approach. In *CVPR*, pages 2083–2090, 2013.

[17] P. Jiang, H. Ling, J. Yu, and J. Peng. Salient region detection by ufo: Uniqueness, focusness and objectness. In *ICCV*, pages 1976–1983, 2013.

[18] J. Kim, D. Han, Y. Tai, and J. Kim. Salient region detection via high-dimensional color transform. In *CVPR*, pages 883–890, 2014.

[19] D. A. Klein and S. Frintrop. Center-surround divergence of feature statistics for salient object detection. In *ICCV*, pages 2214–2219, 2011.

[20] P. Krähenbühl and V. Koltun. Geodesic object proposals. In *ECCV*, pages 725–739. 2014.

[21] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1097–1105, 2012.

[22] X. Li, H. Lu, L. Zhang, X. Ruan, and M.-H. Yang. Saliency detection via dense and sparse reconstruction. In *ICCV*, pages 2976–2983, 2013.

[23] Y. Li, X. Hou, C. Koch, J. M. Rehg, and A. L. Yuille. The secrets of salient object segmentation. In *CVPR*, pages 280–287, 2014.

[24] T. Liu, Z. Yuan, J. Sun, J. Wang, N. Zheng, X. Tang, and H.-Y. Shum. Learning to detect a salient object. *PAMI*, 33(2):353–367, 2011.

[25] Y.-F. Ma and H.-J. Zhang. Contrast-based image attention analysis by using fuzzy growing. In *ACM Multimedia*, pages 374–381, 2003.

[26] R. Margolin, A. Tal, and L. Zelnik-Manor. What makes a patch distinct? In *CVPR*, pages 1139–1146, 2013.

[27] V. Movahedi and J. H. Elder. Design and perceptual validation of performance measures for salient object segmentation. In *POCV*, pages 49–56, 2010.

[28] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In *ICML*, pages 807–814, 2010.

[29] F. Perazzi, P. Krahenbuhl, Y. Pritch, and A. Hornung. Saliency filters: Contrast based filtering for salient region detection. In *CVPR*, pages 733–740, 2012.

[30] P. Pinheiro and R. Collobert. Recurrent convolutional neural networks for scene labeling. In *ICML*, pages 82–90, 2014.

[31] X. Shen and Y. Wu. A unified approach to salient object detection via low rank matrix recovery. In *CVPR*, pages 853–860, 2012.

[32] K. Shi, K. Wang, J. Lu, and L. Lin. Pisa: Pixelwise image saliency by aggregating complementary appearance contrast measures with spatial priors. In *CVPR*, pages 2115–2122, 2013.

[33] J. Sun, H. Lu, and S. Li. Saliency detection based on integration of boundary and soft-segmentation. In *ICIP*, pages 1085–1088, 2012.

[34] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. *arXiv preprint*, 2014.

[35] C. Szegedy, A. Toshev, and D. Erhan. Deep neural networks for object detection. In *NIPS*, pages 2553–2561, 2013.

[36] N. Tong, H. Lu, Y. Zhang, and X. Ruan. Salient object detection via global and local cues. *Pattern Recognition*, 2014.

[37] J. R. Uijlings, K. E. van de Sande, T. Gevers, and A. W. Smeulders. Selective search for object recognition. *IJCV*, 104(2):154–171, 2013.

[38] Y. Xie and H. Lu. Visual saliency detection based on bayesian model. In *ICIP*, pages 645–648, 2011.

[39] Y. Xie, H. Lu, and M.-H. Yang. Bayesian saliency via low and mid level cues. *Image Processing, IEEE Transactions on*, 22(5):1689–1698, 2013.

[40] Q. Yan, L. Xu, J. Shi, and J. Jia. Hierarchical saliency detection. In *CVPR*, pages 1155–1162, 2013.

[41] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang. Saliency detection via graph-based manifold ranking. In *CVPR*, pages 3166–3173, 2013.

[42] W. Zhu, S. Liang, Y. Wei, and J. Sun. Saliency optimization from robust background detection. In *CVPR*, pages 2814–2821, 2014.