Visual Saliency Detection Based on Multiscale Deep CNN Features

Guanbin Li and Yizhou Yu

***Abstract*—Visual saliency is a fundamental problem in both cognitive and computational sciences, including computer vision. In this paper, we discover that a high-quality visual saliency model can be learned from multiscale features extracted using deep convolutional neural networks (CNNs), which have had many successes in visual recognition tasks. For learning such saliency models, we introduce a neural network architecture, which has fully connected layers on top of CNNs responsible for feature extraction at three different scales. The penultimate layer of our neural network has been confirmed to be a discriminative high-level feature vector for saliency detection, which we call deep contrast feature. To generate a more robust feature, we integrate handcrafted low-level features with our deep contrast feature. To promote further research and evaluation of visual saliency models, we also construct a new large database of 4447 challenging images and their pixelwise saliency annotations. Experimental results demonstrate that our proposed method is capable of achieving state-of-the-art performance on all public benchmarks, improving the F-measure by 6.12% and 10.0% respectively on the DUT-OMRON dataset and our new dataset (HKU-IS), and lowering the mean absolute error by 9% and 35.3% respectively on these two datasets.**

***Index Terms*—Convolutional Neural Networks, Saliency Detection, Deep Contrast Feature.**

I. Introduction

Visual saliency attempts to determine the amount of attention steered towards various regions in an image by the human visual and cognitive systems [2]. It is thus a fundamental problem in psychology, neural science, and computer vision. Computer vision researchers focus on developing computational models for either simulating the human visual attention process or identifying visually salient regions. It is originally defined as a task of predicting eye-fixations to investigate the mechanism of human visual system [3]. Recently it has been extended to locating regions of interest, known as salient object detection[4], [5]. Since visual saliency results set relative importance on the visual contents in an image, they are conducive to narrowing the scope of visual processing and saving computing resources. As a result, Visual saliency has been incorporated in a variety of computer vision and image processing tasks to improve their performance. Such tasks include image cropping [6], retargeting [7], summarization [8] and thumbnail generation [9]. Recently, visual saliency has also been increasingly used by visual recognition tasks, such as object tracking [10], image classification [11] and person re-identification [12].

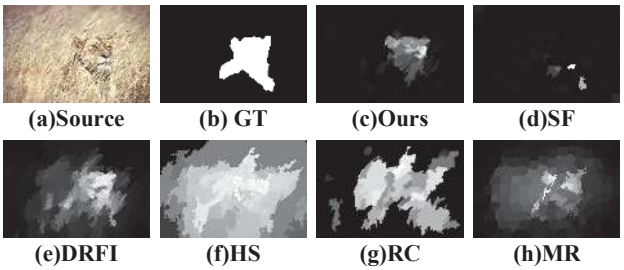


Fig. 1: An example illustrating that saliency models based on handcrafted low-level features are fragile. From top left to bottom right: source image, ground truth, our saliency map, and saliency maps of other five latest methods, including SF[13], DRFI[14], HS[15], RC[16], and MR[17].

Results from perceptual research [18], [19] show that contrast is the most influential factor to visual attention in the human vision system. Local and global contrast has been successfully adopted to derive saliency maps in various saliency detection methods, where the definition of contrast is based on various types of handcrafted image features (e.g., color, intensity and histogram) at the pixel or superpixel level [16], [4], [17]. Though these methods perform well on simple benchmarks, they may fail when the background becomes complex since handcrafted low-level features are not able to effectively capture semantic contexts hidden in an image, and very often the contrast between these low-level features is not strong enough to make salient objects stand out from the background. For example, in Figure 1, a lion is hidden in the bushes and it could not be detected as a salient object using low-level saliency cues alone. However, humans can easily recognize the lion and check it out carefully since it is semantically salient in high-level cognition. Because of this, in our work, we leverage the advantages of high-level semantically meaningful features from deep learning as well as low-level features when inferring saliency maps.

Human visual and cognitive systems involved in the visual attention process are composed of layers of interconnected neurons. For example, the human visual system has layers of simple and complex cells whose activations are determined by the magnitude of input signals falling into their receptive fields. Since deep artificial neural networks were originally inspired by biological neural networks, it is a natural choice to build a computational model of visual saliency using deep artificial neural networks. Specifically, recently popular convolutional neural networks (CNN) are particularly well suited

G. Li is with Sun Yat-sen University, Guangzhou 510006, China, and also with the Department of Computer Science, the University of Hong Kong (e-mail: ligb86@gmail.com).

Y. Yu is with the Department of Computer Science, the University of Hong Kong (e-mail: yizhouy@acm.org).

A preliminary version of this paper appeared in CVPR 2015 [1].

Project website https://sites.google.com/site/ligb86/mdfsaliency/

for this task because convolutional layers in a CNN resemble simple and complex cells in the human visual system [20] while fully connected layers in a CNN act like higher-level inference and decision making.

In this paper, we develop a new computational model for visual saliency using multiscale deep features computed by convolutional neural networks. Deep neural networks, such as CNNs, have recently achieved many successes in visual recognition tasks [21], [22], [23]. Such deep networks are capable of extracting feature hierarchies from raw pixels automatically. Further, features extracted using such networks are highly versatile and often more effective than traditional handcrafted features. Inspired by this, we perform feature extraction using a CNN originally trained over the ImageNet dataset [24]. Since ImageNet contains images of a large number of object categories, our features contain rich semantic information, which is useful for visual saliency because humans pay varying degrees of attention to objects from different semantic categories. For example, viewers of an image likely pay more attention to objects like cars than the sky or grass. In the rest of this paper, we call such features *CNN features*.

By definition, saliency is resulted from visual contrast as it intuitively characterizes certain parts of an image that appear to stand out relative to their neighboring regions or the rest of the image. Thus, to compute the saliency of an image region, our model should be able to evaluate the contrast between the considered region and its surrounding area as well as the rest of the image. Therefore, we extract multiscale CNN features for every image region from three nested and increasingly larger rectangular windows, which respectively encloses the considered region, its immediate neighboring regions, and the entire image.

On top of the multiscale CNN features, our method further trains fully connected neural network layers. Concatenated multiscale CNN features are fed into these layers trained using a collection of labeled saliency maps. Thus, these fully connected layers play the role of a regressor that is capable of inferring the saliency score of every image region from the multiscale CNN features extracted from nested windows surrounding the image region. The penultimate fully connected layer of our neural network is thus becoming a very discriminative high-level feature vector for saliency detection, and we can generate significantly more accurate saliency maps than those from existing saliency models based on low-level features by simply performing logistic regression. We further find out that this high-level discriminative feature vector is complementary to handcrafted low-level features, and train a random forest regressor on concatenated high-level and low-level features. Experimental results show that such hybrid features can further boost the performance of saliency detection.

We have extensively evaluated our CNN-based visual saliency model over existing datasets, and meanwhile noticed a lack of large and challenging datasets for training and testing saliency models. At present, MSRA-B [4] is the most frequently used dataset. However, this dataset has become less challenging over the years because images there typically include a single salient object located away from the image boundary. DUT-OMRON [17] is currently the most challenging dataset with nature images for the research of both salient object detection and eye fixation prediction. To facilitate research and evaluation of advanced saliency models, we have created another large dataset where an image likely contains multiple salient objects, which have a more general spatial distribution in the image. Furthermore, our dataset only includes images that receive consistent saliency annotations from multiple users. Our proposed saliency model has significantly outperformed all existing saliency models over this new dataset as well as all existing datasets.

In summary, this paper has the following contributions:

- A new visual saliency model is proposed to incorporate multiscale CNN features extracted from nested windows with a deep neural network with multiple fully connected layers. The deep neural network for saliency estimation is trained using regions from a set of labeled saliency maps. The penultimate layer of the proposed neural network can be viewed as a discriminative high-level feature vector for saliency detection, and can further boost saliency performance when concatenated with handcrafted low-level features.
- A complete saliency framework is developed by further integrating an aggregated saliency map over multi-level image segmentations with a spatial coherence model based on a fully connected CRF.

The remainder of the paper is organized as follows. Section II reviews related work and differentiates our method from such work. Section III introduces our proposed multiscale deep features. The complete algorithm is presented in Section IV. A new dataset was introduced in the preliminary version of this paper [1], we present it here again in Section V for the completeness of this paper. Extensive experimental results and comparisons are presented in Section VI. And Section VII concludes this paper.

II. Related Work

A. Salient Object Detection

Visual saliency algorithms can be categorized into three groups: bottom-up, top-down, and hybrid algorithms of the previous two.

Bottom-up models are primarily based on the center-surround scheme, computing a master saliency map using a linear or non-linear combination of low-level visual attributes such as color, intensity, texture and orientation [3], [25], [5], [26], [4]. According to the spatial scope of saliency computation, these methods can be further divided into local methods and global methods. Local methods measure saliency by considering the contrast between each pixel or image region and a small neighborhood. One example of this category is the work by Itti *et al.* [3], where color and orientation contrasts across multiple scales are computed to measure local conspicuity. While it is able to identify salient pixels, as pointed out by Cheng *et al.* [16], the results are generally blurry and contain a significant amount of false detection. Ma and Zhang [27] proposed a fuzzy growing process to simulate the process of human perception using local contrast as a

measure of saliency. Harel *et al.* [28] created feature maps using the method from [3] but perform normalization using graph-based random walk. As these methods only consider local contrast, they tend to detect high-frequency features, such as edges or noise, only and suppress homogeneous regions at the interior of salient objects.

Global bottom-up methods estimate saliency by considering contrast over the entire image. Achanta [5] proposed a frequency-tuned method that directly estimates pixel saliency by computing color differences from the average image color. Cheng *et al.* [16], [26] took color histograms as regional features and computed saliency on the basis of histogram dissimilarity. In [15], Yan *et al.* proposed a hierarchical framework to address small-scale high-contrast patterns. Recently, much effort has been made towards designing discriminative features and saliency priors. Most algorithms essentially follow the region contrast framework, aiming to discover features that better characterize the distinctiveness of an image region with respect to its surrounding area. In [4], three novel features are integrated with a conditional random field. A model based on low-rank matrix recovery is presented in [29] to integrate low-level visual features with higher-level priors. Chen *et al.* [30] designs a structure-aware descriptor based on the intrinsic biharmonic distance metric which is able to simultaneously integrate local and global structure information. Though significant improvements have been made, these global features are still weak in capturing image semantic information.

Top-down methods in general require the incorporation of high-level knowledge, such as objectness and object detectors in the computational process [31], [32], [29]. In [33], Judd trained a top-down saliency model using high-level image features including those based on face detection and person detection results. Borji *et al.* [34] integrated bottom-up and top-down features when learning their saliency model, considering person and car detectors as high-level priors. In [31], Jia *et al.* computed a high-level saliency prior using objectness without category information, and applied a Gaussian MRF to enforce the consistency among salient regions. Chang *et al.* [32] proposed a framework which conceptually integrates objectness and saliency via a graphical model accounting for their relationship. Our deep feature extracted from Krizhevsky's CNN [21] implicitly encodes the semantic information of 1.2 million images and has much stronger generalization capability than those based on a relatively small number of object detectors (e.g. face, human and car) or approximate objectness.

Saliency priors, such as the center prior [4], [33] and the boundary prior [14], [35], are widely used to heuristically improve saliency estimation. The center prior is normally formulated as a Gaussian fall-off map assigning higher saliency to the central region of an image while the boundary prior takes a complementary perspective and assigns image boundary regions lower saliency. These saliency priors are either directly integrated with other saliency cues as weights [26], [36], [31] or used as features in learning based algorithms [14], [33]. While these empirical priors can improve saliency results for many images, they can fail when a salient object is off-center or significantly overlaps with the image boundary. Note that object location cues and boundary-based background modeling are not neglected in our framework, but have been implicitly incorporated through multiscale CNN feature extraction and neural network training.

B. Deep Convolutional Neural Networks

Convolutional neural networks have recently achieved many successes in visual recognition tasks, including image classification [21], object detection [23], and scene parsing [22]. Donahue *et al.* [37] pointed out that features extracted from Krizhevsky's CNN trained on the ImageNet dataset [24] can be repurposed to generic tasks. Razavian *et al.* [38] extended their results and concluded that CNN-based deep learning can be a strong candidate for any visual recognition tasks. Nevertheless, saliency detection is generally defined as a low-level computer vision problem and acts quite different from conventional object detection. It is the contrast against the surrounding area rather than the content inside an image region that should be learned for saliency prediction. This paper proposes a simple but very effective neural network architecture for digging out contrast information hidden in multi-scale deep CNN features and inferring the saliency score for each region. Note that in [22], a multiscale convolutional network was trained to extract hierarchical feature vectors well suited for scene labeling. The raw input image was transformed through a Laplacian pyramid into three scales before being fed to a 3-stage convolutional network, and the pixelwise features are similar to hypercolumn features [39], formed by stacking responses corresponding to the same pixel from all convolutional layers of the CNN. Different from region-oriented features used in our method, their pixel-oriented features are not focused on region contrast which is crucial in saliency detection.

There exist other convolutional neural network based saliency detection methods since the publication of our earlier work [1]. Wang *et al.* [40] applied a deep neural network (DNN-L) to learn local patch features for determining the saliency score of the center pixel. Since only local patches were considered, the quality of the generated saliency map may be sensitive to high-frequency background noise, and homogeneous regions inside salient objects may be misclassified. Therefore, a global search stage was added to exploit the complex relationships among global saliency cues which are represented using handcrafted features. Li *et al.* [41] proposes an end-to-end deep contrast network which considers both pixel-level and segment-wise saliency inference. In [42], both global and local contexts were utilized and integrated into a unified deep learning framework for saliency detection. Their model calculates a saliency score for every superpixel. The global context of a superpixel contains the whole image with the superpixel located at the center of the context, while the local context has a fixed size equal to one third of the global context. While our proposed method also extracts CNN-based context features, it is different from [42] in three aspects and is also more robust. First, the size of our local context is spatially varying, relying on the actual size of the surrounding regions. Our local context can better estimate the contrast

between each region and the background. Second, Instead of direct regression, we propose a neural network architecture to mine the contrast information hidden inside the concatenated multiscale deep features. Third, we apply multi-level segmentation and pixel-level CRF-based refinement to compensate the inaccuracy caused by superpixels. Experimental results demonstrate that our proposed method outperforms all existing CNN based saliency models.

This paper provides a more complete understanding of multiscale deep features first presented in the conference version [1], providing additional insights, analysis, and evaluation. Furthermore, we improve the original framework in two aspects. First, we propose the concept of deep contrast features, and analyze their strengths and weaknesses. To complement deep contrast features, we also extract low-level features, which can effectively capture segment properties as well as color and texture contrasts between a region and the rest of the image. Low-level features are concatenated with deep contrast features to yield a hybrid deep and handcrafted feature vector. We show that training a random forest regressor over this hybrid feature vector can further boost the performance. Second, to enhance spatial coherence and better preserve the boundary of salient objects, a fully connected CRF model is integrated into our framework to perform pixelwise saliency refinement.

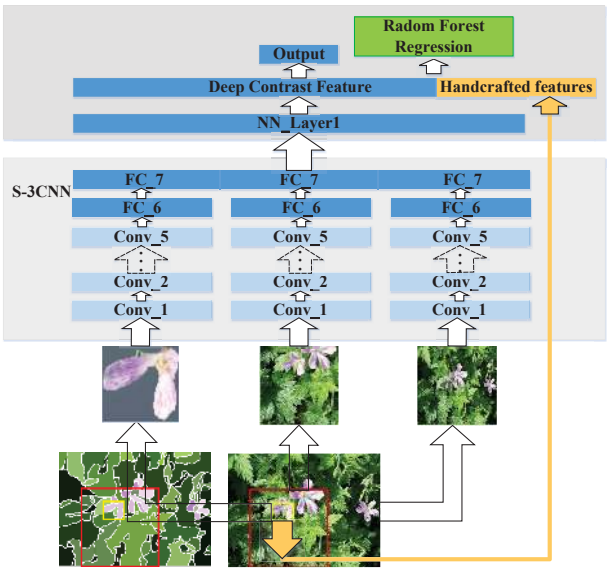


Fig. 2: The architecture of our deep feature based visual saliency model.

III. Saliency Inference with Deep Features

As shown in Fig. 2, the architecture of our deep feature based model for visual saliency consists of one output layer and two fully connected hidden layers on top of three deep convolutional neural networks. Our saliency model requires an input image to be decomposed into a set of nonoverlapping regions, each of which has almost uniform saliency values internally. The three deep CNNs are responsible for multi-scale feature extraction. For each image region, they perform automatic feature extraction from three nested and increasingly larger rectangular windows, which are respectively the bounding box of the considered region, the bounding box of its immediate neighboring regions, and the entire image. The features extracted from the three CNNs are fed into the two fully connected layers, each of which has 300 neurons. The output of the second fully-connected layer is fed into the output layer, which performs logistic regression that infers the probability of a region being salient. When generating a saliency map for an input image, we run our trained saliency model repeatedly over every region of the image to produce a single saliency score for that region. This saliency score is further transferred to all pixels within that region. When the output of the penultimate layer is taken as a deep contrast feature, it can be concatenated with handcrafted low-level features to further boost saliency detection performance.

A. Multiscale Feature Extraction

We extract multiscale features for each image region with a deep convolutional neural network originally trained over the ImageNet dataset [24] and fine-tuned for object detection[23] using Caffe [43], an open source framework for CNN training and testing. The architecture of this CNN has eight layers including five convolutional layers and three fully-connected layers. Features are extracted from the output of the second last fully connected layer, which has 4096 neurons. Although this CNN was originally trained on datasets for visual recognition, automatically extracted CNN features turn out to be highly versatile and can be more effective than traditional handcrafted features on other visual computing tasks.

Since an image region may have an irregular shape while CNN features have to be extracted from a rectangular region, to make the CNN features only relevant to the pixels inside the region, as in [23], we define the rectangular region for CNN feature extraction to be the bounding box of the image region and fill the pixels outside the region but still inside its bounding box with the mean pixel values at the same locations across all ImageNet training images. These pixel values become zero after mean subtraction and do not have any impact on subsequent results. We warp the region in the bounding box to a square with 227x227 pixels to make it compatible with the deep CNN trained for ImageNet. The warped RGB image region is then fed to the deep CNN and a 4096-dimensional feature vector is obtained by forward propagating a mean-subtracted input image region through all the convolutional layers and fully connected layers. We name this vector *feature A*.

Feature A itself does not include any information around the considered image region, thus is not able to tell whether the region is salient or not with respect to its neighborhood as well as the rest of the image. To include features from an area surrounding the considered region for understanding the amount of contrast in its neighborhood, we extract a second feature vector from a rectangular neighborhood, which is the bounding box of the considered region and its immediate

neighboring regions. All the pixel values in this bounding box remain intact. Again, this rectangular neighborhood is fed to the deep CNN after being warped. We call the resulting vector from the CNN *feature B*.

As we know, a very important cue in saliency computation is the degree of (color and content) uniqueness of a region with respect to the rest of the image. The position of an image region in the entire image is another crucial cue. To meet these demands, we use the deep CNN to extract *feature C* from the entire rectangular image, where the considered region is masked with mean pixel values for indicating the position of the region. These three feature vectors obtained at different scales together define the features we adopt for saliency model training and testing. Since our final feature vector is the concatenation of three CNN feature vectors, we call it S-3CNN.

B. Neural Network Training

As discussed above, our proposed S-3CNN is a concatenation of three parts of deep features of 12288 dimmensions. On top of S-3CNN, we train a neural network with one output layer and two fully connected hidden layers. This network plays the role of a regressor that infers the saliency score of every image region from the multiscale CNN features extracted for the image region. It is well known that neural networks with fully connected hidden layers can be trained to reach a very high level of regression accuracy.

Concatenated multiscale CNN features are fed into this network, which is trained using a collection of training images and their labeled saliency maps, that have pixelwise binary saliency label. Before training, every training image is first decomposed into a set of regions. The saliency label of every image region is further estimated using pixelwise saliency labels. During the training stage, only those regions with 70% or more pixels with the same saliency label are chosen as training samples, and their saliency score are set to either 1 or 0 respectively. During training, the output layer and the fully connected hidden layers together minimize the least-squares prediction errors accumulated over all regions from all training images.

C. Deep Contrast Feature

Note that the output of the penultimate layer of our neural network can be viewed as a fine-tuned feature vector for saliency detection. The final layer of our neural network essentially performs logistic regression on this fine-tuned feature, which effectively captures the contrast of a region with respect to its surrounding neighborhood at the semantic level. We name this feature Deep Contrast Feature (DCF) in the rest of this paper. Traditional regression techniques, such as support vector regression and boosted decision trees, can be trained on DCF to generate a saliency score for every image region. Nonetheless, we have found experimentally that this feature vector is highly discriminative and even simple logistic regression performed in the final layer of our neural network is sufficient to produce state-of-the-art performance on all visual saliency datasets. Since DCF reflects image semantics, we have further confirmed that DCF is complementary to handcrafted low-level features. In the following section, we show that training a random forest regressor over hybrid features including both DCF and some low-level regional features can further boost the performance.

IV. The Complete Algorithm

A. Multi-Level Image Decomposition

A variety of methods can be applied to decompose an image into nonoverlapping regions. Example methods include grids, region growing, and pixel clustering. Hierarchical image segmentation can generate regions at multiple scales to support the intuition that a semantic object at a coarser scale may be composed of multiple parts at a finer scale. In this paper, we applied the graph-based image segmentation[44] approach to compute M levels of segmentation based on M groups of segmentation parameters. Specifically, for an image I, M levels of image segmentations, $S = \{S_1, S_2, ..., S_M\}(|S_i| = N_i)$, are constructed from the finest to the coarsest scale. The regions at any level form a nonoverlapping image decomposition. In our earlier version [1], to generate a more accurate segmentation, region merger was prioritized by edge strength at boundary pixels shared by two adjacent regions and the edge strength was determined by an ultrametric contour map (UCM) proposed in [45]. However, calculating UCM is time-consuming but does not clearly improve the accuracy of the final saliency map. In this paper, we simply apply the graph-based segmentation algorithm in [44] to generate 15 levels of segmentations using different parameter settings. The target number of regions at the finest and coarsest levels are controled to be around 300 and 20 respectively, and the number of regions at intermediate levels follows a geometric series. We train a unified model based on all the regions across these 15 levels of segmentations instead of a single model for each level of segmentation.

B. HDHF: Hybrid Deep and Handcrafted Feature

As discussed in Section III-C, the initial saliency map from our trained neural network can be viewed as the result of logistic regression on DCF. As shown in Fig. 3, DCF is especially adept at detecting salient regions in images with low contrast and complex background as long as there exists semantic distinction against their surrounding neighborhoods (Fig. 3 a&b). However, since DCF is derived from multi-scale CNN features that are focused on image semantics, it may not contain sufficient information about the contrast in low-level attributes. For example, as shown in Fig. 3 c, when regions are salient due to contrast in low-level attributes (e.g. color and texture), DCF tends to perform worse than those methods based on handcrafted low-level features. And there are many examples where neither deep features nor handcrafted low-level features alone are good enough to generate accurate saliency maps (e.g. Fig. 3 a,c&d)). To overcome this deficiency, we propose a small set of complementary low-level features to compensate DCF.

Given an image, we first generate an initial saliency map SM^{init} using multiscale deep features. We define a pseudo

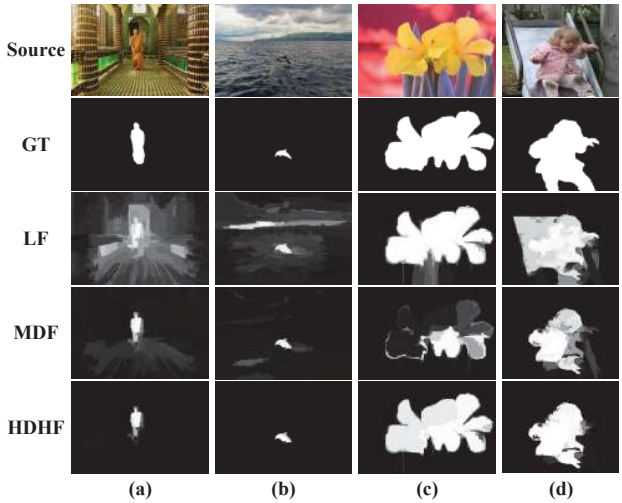


Fig. 3: The integration of handcrafted low-level features with DCF. The ground truth (GT) is shown in the second row. LF denotes saliency maps generated using our defined handcrafted low-level feature. MDF refers to saliency maps generated using our multiscale deep feature. HDHF refers to saliency maps generated using hybrid deep and handcrafted feature. HDHF is consistently better than MDF and LF.

background region B as the set of pixels within 30 pixels from the image borders and having an initial saliency value $SM^{init} < 0.1$. We compute low-level features for the entire image, the pseudo background region, and every region in every level of image segmentation. Such low-level features includes both color and texture features. Color features include RGB, LAB and HSV histograms as well as their corresponding average values. Texture features include the histogram of maximum responses of LM filters as well as the histogram of LBP features.

On the basis of these low-level features, for each region R in each level of segmentation, we extract a 39-dimensional low-level feature descriptor including both contrast features and segment properties. The contrast features include the contrast between the low-level features of R and their corresponding features of the pseudo background B as well as the contrast between the low-level features of R and their counterparts for the entire image. We adopt the χ^2 distance as the contrast between two histograms and the absolute difference as the contrast between two scalar features. Segment properties include the variance of various color and texture features as well as geometric properties including the perimeter and area of the segment. Note that the geometric properties are normalized with respect to the overall image size. The details of all handcrafted low-level features are given in Table I. We normalize the L_2 norm of both our proposed 300-dimensional DCF and this handcrafted low-level feature descriptor before concatenating them into a 339-dimensional hybrid feature vector, called hybrid deep and handcrafted feature (HDHF).

C. Training Saliency Regressor over HDHF

To demonstrate the effectiveness of HDHF, we train a random forest regressor using hybrid deep and handcrafted features. Each training sample corresponds to a region with a 339-dimensional HDHF. As done for neural network training in Section III-B, only those regions with 70% or more pixels with the same saliency label are chosen as training samples, and their saliency scores are set to either 1 or 0 accordingly. Learning a random forest based model can automatically integrate low-level and high-level features, and map every HDHF to a saliency score. We also train another random forest model base only on 39 dimensional low-level features for comparison. As shown in Fig. 3 and the quantitative results in Section VI-F, HDHF based saliency maps are consistently better than those based on DCF or handcrafted features only.

D. Saliency Map Fusion

Given the regions in an image decomposition, we can generate an initial saliency map either with the neural network model or the HDHF-based random forest regressor. Given M levels of segmentations, we obtain M saliency maps, $\{A^{(1)}, A^{(2)}, ..., A^{(M)}\}$, interpreting salient parts of the input image at various granularity. We aim to further fuse them together to obtain an aggregated saliency map. To this end, we take a simple approach by assuming the aggregated saliency map is a linear combination of the maps at individual segmentation levels, and learn the weights in the linear combination by running a least-squares estimator over a validation dataset. Thus, our aggregated saliency map A is formulated as follows,

$$\begin{aligned} A &= \sum_{k=1}^{M} \alpha_k A^{(k)} \\ \text{s.t. } \{\alpha_k\}_{k=1}^{M} &= \underset{\alpha_1,\alpha_2,...,\alpha_M}{\operatorname{argmin}} \sum_{i \in I_v} \left\| A_i - \sum_k \alpha_k A_i^{(k)} \right\|_F^2 . \end{aligned} \tag{1}$$

where I_v stands for the set of indices of the images in the validation dataset.

Note that there are many options for saliency fusion. For example, a conditional random field (CRF) framework has been adopted in [46] to aggregate multiple saliency maps from different methods. Nevertheless, we have found that, in our context, a linear combination of all saliency maps can already serve our purposes well and is capable of producing aggregated maps with a quality comparable to those obtained from more complicated techniques.

E. Spatial Coherence Based on CRF

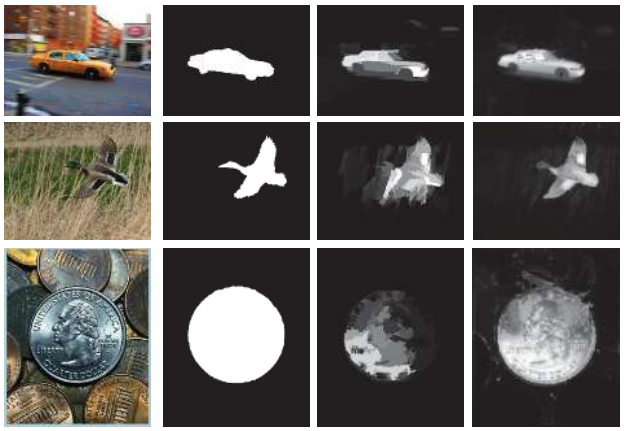
Due to the fact that image segmentation is imperfect and our model assigns saliency scores to individual segments, noisy scores inevitably appear in the above aggregated saliency map. To enhance spatial coherence, we perform pixelwise saliency refinement using the fully connected CRF model in [47]. This model solves a binary pixel labeling problem, and employs the following energy function,

$$E(L) = -\sum_i \log P(l_i) + \sum_{i,j} \theta_{ij}(l_i, l_j), \tag{2}$$

where L represents a binary label (salient or not salient) assignment for all pixels. $P(l_i)$ is the probability of pixel x_i

Contrast Descriptors (Color and Texture)				Segment Properties			
Notation	Features	Definition	Dim	Notation	Features	Definition	Dim
$c_1 \sim c_6$	Difference between Average RGB Values	$\lvert R^{rgb} - B^{rgb}\rvert$, $\lvert R^{rgb} - I^{rgb}\rvert$	6	$s_1 \sim s_3$	Variances of RGB values	var_R^r, var_R^g, var_R^b	3
$c_7 \sim c_8$	χ^2 distance between RGB Histograms	$\chi^2(h_{rgb}^R, h_{rgb}^B)$, $\chi^2(h_{rgb}^R, h_{rgb}^I)$	2	$s_4 \sim s_6$	Variances of LAB values	var_R^l, var_R^a, var_R^b	3
$c_9 \sim c_{14}$	Difference between Average LAB Values	$\lvert R^{lab} - B^{lab}\rvert$, $\lvert R^{lab} - I^{lab}\rvert$	6	$s_7 \sim s_9$	Variance of HSV values	var_R^h, var_R^s, var_R^v	3
$c_{15} \sim c_{16}$	χ^2 distance between LAB Histograms	$\chi^2(h_{lab}^R, h_{lab}^B)$, $\chi^2(h_{lab}^R, h_{lab}^I)$	2	s_{10}	Normalized perimeter	$Perimeter(R)$	1
$c_{17} \sim c_{22}$	Difference between Average HSV Values	$\lvert R^{hsv} - B^{hsv}\rvert$, $\lvert R^{hsv} - I^{hsv}\rvert$	6	s_{11}	Normalized area	$Area(R)$	1
$c_{23} \sim c_{24}$	χ^2 distance between HSV Histograms	$\chi^2(h_{hsv}^R, h_{hsv}^B)$,$\chi^2(h_{hsv}^R, h_{hsv}^I)$	2				
$c_{25} \sim c_{26}$	χ^2 distance b.w. Max response LM Histograms	$\chi^2(h_{LM}^R, h_{LM}^B)$, $\chi^2(h_{LM}^R, h_{LM}^I)$	2				
$c_{27} \sim c_{28}$	χ^2 distance between LBP Histograms	$\chi^2(h_{LBP}^R, h_{LBP}^B)$, $\chi^2(h_{LBP}^R, h_{LBP}^I)$	2				

TABLE I: A detailed description of handcrafted low-level features. R denotes an image segment, B refers to the pseudo background region, and I denotes the entire image.



(a)Source (b)GT (c)without CRF (d)with CRF

Fig. 4: Comparison of saliency detection results with and without CRF.

having label l_i, which indicates the likelihood of pixel x_i being salient. Initially, $P(1) = S_i$ and $P(0) = 1 - S_i$, where S_i is the saliency score at pixel x_i from the above aggregated saliency map A. $\theta_{ij}(l_i, l_j)$ is a pairwise potential and defined as follows,

$$\theta_{ij} = \mu(l_i, l_j)\left[\omega_1 \exp\left(-\frac{\|p_i - p_j\|^2}{2\sigma_\alpha^2} - \frac{\|I_i - I_j\|^2}{2\sigma_\beta^2}\right) + \omega_2 \exp\left(-\frac{\|p_i - p_j\|^2}{2\sigma_\gamma^2}\right)\right], \tag{3}$$

where $\mu(l_i, l_j) = 1$ if $l_i \neq l_j$, and zero otherwise. θ_{ij} involves two kernels. The first bilateral kernel depends on both pixel positions (denoted as p) and colors (denoted as I), suggesting nearby pixels with similar colors to be assigned similar saliency scores. The degrees of color similarity and pixel closeness are controlled by two parameters, σ_α and σ_β, respectively. The second kernel only depends on pixel position and aims at removing small isolated regions. The "scale" of the Gaussian kernel is controlled by σ_γ. The parameters are determined through cross validation using the validation set of MSRA-B dataset in our experiment, as in [47].

Energy minimization is based on a mean field approximation to the CRF distribution and high-dimensional filtering can be utilized to speed up computation. In this paper, we use the publicly available implementation of [47] to minimize the energy, and it takes less than 0.5 second on an image with 300×400 pixels. At the end of energy minimization, we generate a saliency map using the posterior probability of each pixel being salient. Note that features other than color can be used in the first term to boost performance (e.g. contour information was used in an earlier version of this paper [1]). Currently, we only use color for the sake of efficiency and find it sufficient for enhancing spatial coherence and removing noisy saliency scores in the aggregated saliency map due to imperfect segmentation. The result is an enhanced saliency map. As shown in Fig. 4, our initial saliency maps in general look fragmented and the boundaries of salient objects are not well preserved. The application of the CRF model can not only give rise to smoother results with pixelwise accuracy but also better preserve the boundaries of salient objects. A quantitative study of the effectiveness of the CRF model can be found in Section VI-D3.

V. A New Dataset

We have constructed a more challenging dataset to facilitate the research and evaluation of visual saliency models. To build the dataset, we initially collected 7320 images. These images were chosen by following at least one of the following criteria:

1) there are multiple disconnected salient objects;
2) at least one of the salient objects touches the image boundary;
3) the background is complex;
4) the color contrast (the minimum Chi-square distance between the color histograms of any salient object and its surrounding regions) is less than 0.7.

To reduce label inconsistency, we asked three people to annotate salient objects in all 7320 images individually using a custom designed interactive segmentation tool. On average, each person takes 1-2 minutes to annotate one image. The annotation stage spanned over three months.

Let $A^p = \{a_x^{(p)}\}$ be the binary saliency mask labeled by the p-th user. And $a_x^{(p)} = 1$ if pixel x is labeled as salient and $a_x^{(p)} = 0$ otherwise. We define label consistency as the ratio

between the number of pixels labeled as salient by all three people and the number of pixels labeled as salient by at least one of the people. It is formulated as

$$C = \frac{\sum_x \left(\prod_{p=1}^{3} a_x^{(p)}\right)}{\sum_x \mathbf{1}\left(\sum_{p=1}^{3} a_x^{(p)} \neq 0\right)}. \quad (4)$$

We excluded those images with label consistency $C < 0.9$, and 4447 images remained. For each image that passed the label consistency test, we generated a ground truth saliency map from the annotations of three people. The pixelwise saliency label in the ground truth saliency map, $G = \{g_x | g_x \in \{0, 1\}\}$, is determined according to the majority label among the three people as follows,

$$g_x = \mathbf{1}\left(\sum_{p=1}^{3} a_x^{(p)} \geq 2\right). \quad (5)$$

At the end, our new saliency dataset, called HKU-IS, contains 4447 images with high-quality pixelwise annotations. It is more challenging and unbiased compared with the most often used dataset (e.g. MSRA-B [4]).

VI. Experimental Results

A. Dataset

We have evaluated the performance of our method on several public benchmarks for salient object detection as well as on our own dataset.

MSRA-B[4]. This dataset has 5000 images, and is widely used for salient object detection. Most of the images contain only one salient object. Pixelwise annotation was provided by [14].

DUT-OMRON[17]. This large dataset contains 5168 natural images. Both bounding boxes and pixelwise salient object annotations are provided. We have noticed that many saliency annotations in this dataset may be controversial among different human observers. As a result, none of the existing saliency models has achieved a high accuracy on this dataset.

SOD[48]. This dataset has 300 images, and it was originally designed for image segmentation. Pixelwise annotation of salient objects in this dataset was generated by [14]. This dataset is very challenging since many images contain multiple salient objects either with low contrast or overlapping with the image boundary.

PASCAL-S[49]. This dataset was built using the validation set of the PASCAL VOC 2010 segmentation challenge. It contains 850 images with pixelwise salient object annotation. The groundtruth saliency masks were labeled by 12 subjects. We threshold the masks at 0.5 to obtain binary masks as suggested in [49].

ECSSD[15]. This dataset contains 1,000 structurally complex images acquired from the Internet with pixelwise groundtruth masks.

HKU-IS. Our new dataset contains 4447 images with pixelwise annotation of salient objects.

To save space, the performance on the SED [50] and ICOSEG [51] datasets is no longer reported since these datasets are not challenging and not widely used. Readers can refer to an earlier version of our paper [1] for performance comparisons on these two datasets. To facilitate a fair comparison with other methods, we divided the MSRA dataset into three parts as in [14], 2500 for training, 500 for validation and the remaining 2000 images for testing. To test the adaptability of trained saliency models to other different datasets, we use the models trained on the MSRA-B dataset and test them over all other datasets.

As discussed in the previous sections, we generate two sets of saliency results using our proposed saliency models. To evaluate the effectiveness of multiscale deep features, we construct the first set of saliency maps from the output of the neural network model aggregated with multi-level fusion and further enhanced using the CRF model. To demonstrate the complementariness between DCF and handcrafted low-level features, we generate the second set of saliency maps from the random forest regressor using HDHF, also aggregated with multi-level fusion and enhanced using the CRF model. We refer to the first set of saliency maps as MDF, and the second set HDHF in the rest of this paper. When conducting an ablation study, we investigate the contribution of each component to the accuracy of MDF, and we show the overall performance of both MDF and HDHF when comparing them with other state-of-the-art methods.

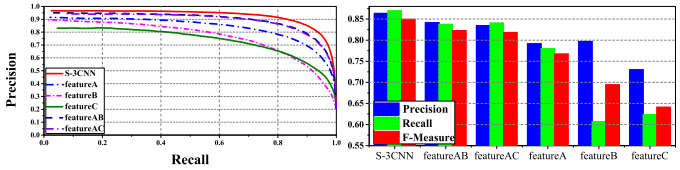


Fig. 5: The effectiveness of our S-3CNN feature. The left figure shows the precision-recall curves of models trained on MSRA-B using different components of S-3CNN, while the right figure shows the corresponding precision, recall and F-measure using an adaptive threshold.

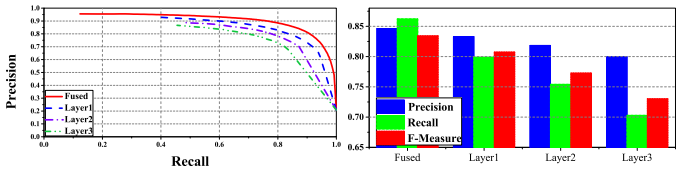
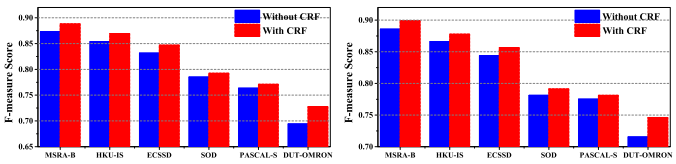


Fig. 6: The effectiveness of multilevel fusion."Layer1", "Layer2" and "Layer3" refer to the three segmentation levels that have the highest single-level saliency detection performance. The left figure shows the precision-recall curves while the right figure shows the corresponding precision, recall and F-measure using an adaptive threshold.

B. Implementation Details

We train our saliency models using the training set of the MSRA-B dataset and test them over all other datasets. The



(a) (b)

Fig. 7: The effectiveness of CRF-based spatial coherence. (a) Maximum F-measure of our MDF-based model achieved with and without CRF on six saliency detection datasets. (b)Maximum F-measure of our HDHF-based model achieved with and without CRF on the same datasets.

training set contains 2500 images. We perform 15 levels of image segmentation and extract around 800 segments across all levels from each image. The S-3CNN feature vector extracted from each segment forms one training sample, and we have 1.9 million training samples in total. Though the dimension of S-3CNN and HDHF are larger than 12 thousand, the number of our training samples is large enough to train an accurate model free from overfitting. We have implemented our proposed framework in Caffe [43]. More specifically, to train our three-layer perceptron network, the learning rate is set to 0.2 while the momentum parameter is set to 0.5. We use the hyperbolic tangent function as the activation function in the hidden layers and the sigmoid function in the output layer. When jointly fine-tune deep CNN model with our proposed three-layer MLP, the learning rate of the initial deep CNN model is set to 0.0001. We cross-validate the parameters in the fully connected CRF according to [47] on the validation set and the final values of w_1, w_2, σ_α, σ_β, and σ_γ are 3.0, 5.0, 3.0, 50.0 and 3.0 respectively.

C. Evaluation Criteria

Following [52], [26], we first use standard precision-recall (PR) and receiver operating characteristic (ROC) curves to evaluate the performance of our method. A continuous saliency map can be converted to a binary mask using a threshold, resulting in a pair of precision and recall values when the binary mask is compared against the ground truth. A PR curve is then obtained by varying the threshold from 0 to 1. The curves are averaged over each dataset. The ROC curve can be conveniently generated according to the true positive rates and false positive rates obtained during the calculation of the PR curve. The AUC (Area Under ROC Curve) score is also reported given the ROC curve.

Second, since high precision and high recall are both desired in many applications, we compute the F-measure[5] as

$$F_\beta = \frac{(1+\beta^2)\cdot Precision \cdot Recall}{\beta^2 \cdot Precision + Recall}, \tag{6}$$

where β^2 is set to 0.3 to weigh precision more than recall as suggested in [5]. We report the maximum F-measure score among all pairs of precision and recall values. We also report the performance once every saliency map is binarized with an image-dependent threshold proposed by [5]. This adaptive threshold is determined to be twice the mean saliency of the image:

$$T_a = \frac{2}{W \times H}\sum_{x=1}^{W}\sum_{y=1}^{H} S(x,y), \tag{7}$$

where W and H are the width and height of the saliency map S, and $S(x,y)$ is the saliency score of the pixel at (x,y). We report the average precision, recall and F-measure over each dataset using this adaptive threshold.

Although commonly used, PR curves have limited value because they fail to consider true negative pixels. For a more balanced comparison, we adopt the mean absolute error (MAE) as another evaluation criterion. It is defined as the average pixelwise absolute difference between the binary ground truth (G) and the saliency map (S) [13],

$$MAE = \frac{1}{W \times H}\sum_{x=1}^{W}\sum_{y=1}^{H} |S(x,y) - G(x,y)|. \tag{8}$$

MAE measures the numerical distance between the ground truth and the estimated saliency map, and is more meaningful in evaluating the applicability of a saliency model in a task such as object segmentation.

D. Ablation Study

1) Effectiveness of S-3CNN: As discussed in Section III-A, our multiscale CNN feature vector, S-3CNN, consists of three components, A, B and C. To show the effectiveness and necessity of these three parts, we have trained five additional models for comparison, which respectively take $feature$ A only, $feature$ B only, $feature$ C only, concatenated A and B, and concatenated A and C. These five models were trained on MSRA-B using the same setting as the one taking S-3CNN. Quantitative results were obtained on the testing images in the MSRA-B dataset. As shown in Fig. 5, the model trained using S-3CNN consistently achieves the highest PR curve and best performance on average precision, recall and F-measure. Models trained using two components perform much better than those trained using a single component.

These results demonstrate that the three components of our multiscale CNN feature vector are complementary to each other, and the training stage of our saliency model is capable of discovering and understanding region contrast information hidden in our multiscale features.

2) Multilevel Decomposition: Our method exploits information from multiple levels of image segmentation. As shown in Fig. 6, the performance of a single segmentation level is not comparable to the performance of the fused model. The aggregated saliency map from 15 levels of image segmentation improves the average precision by 2.15% and at the same time improves the recall rate by 3.47% when it is compared with the result from the best-performing single level.

3) Spatial Coherence: In Section IV-E, a fully connected CRF model is incorporated to improve spatial coherence and refine the saliency scores obtained using MDF and HDHF. To validate its effectiveness, we have evaluated the performance of our saliency models with and without the CRF model across six datasets. As shown in Figure. 7, the CRF can consistently

Deep Model	AUC	MAE	F-Measure
RCNN	0.978	0.066	0.888
RCNN*	**0.979**	**0.065**	**0.901**
AlexNet	0.975	0.070	0.879
AlexNet*	0.975	0.068	0.881
VGG16	0.976	0.070	0.881
VGG16*	0.978	0.068	0.883
VGG19	0.977	0.069	0.882
VGG19*	0.978	0.069	0.883

TABLE II: Comparison of saliency detection performance using different CNN architectures. $*$ refers to deep model with parameters fine-tuned.

improve the results computed using MDF and HDHF across all the six datasets. Especially on the DUT-OMRON dataset which contains the largest number of testing images, the CRF increases the F-measure by 4.7% on the HDHF results and 4.2% on the MDF results.

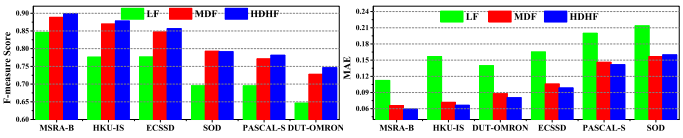
E. Evaluation on Contemporary CNN Architectures

We evaluate the effectiveness of deep features extracted using different CNN architectures. We extract deep features using 4 contemporary deep CNN architectures, and train our saliency model on MSRA-B using such deep features. Evaluated CNN architectures include AlexNet [21], VGG16 [53], VGG19 [53] and the R-CNN model [23]. We obtain quantitative comparison results on the testing images of the MSRA-B dataset. As shown in Table II, the R-CNN model achieved slightly better performance than others. This model can better capture the feature of an image region probably because it was fine-tuned on a dataset of image regions for the purpose of object detection.

We have also tried to jointly fine-tune the deep CNN model with our proposed MLP. As shown in Table II, models with parameters fine-tuned can deliver slightly better results. Though our proposed model can effectively mine the contrast information from different scales of image regions and learn a superior deep contrast feature for saliency detection, it can hardly fine-tune much better description for each scale. In fact, all of these deep models are capable of capturing the feature of an image region but the regional feature performance of all these deep models does not vary much when applied in our contrast learning framework. To sum up, for saliency estimation, discovering the contrast between a region and its surrounding neighborhood is equally important.

F. The Performance of HDHF

We evaluate the effectiveness of HDHF quantitatively by comparing its performance against that of MDF, which is based on deep features (S-3CNN) only, and LF, which is based on the 39-dimensional handcrafted low-level features only. Figure 8 shows the F-measure and MAE of these three methods on six datasets. HDHF performs better than MDF most of the time, and consistently and significantly outperforms LF. Especially on the DUT-OMRON dataset, HDHF



(a) (b)

Fig. 8: Performance of our HDHF-based model. (a)Maximum F-measure of HDHF, MDF and LF on six saliency detection datasets. (b) MAE of HDHF, MDF and LF on the same datasets.

improves the F-measure of MDF by 2.6% and LF by 12.4% while, at the same time, lowers the MAE of MDF by 8.5% and LF by 46.3%.

G. Comparison with the State of the Art

Let us compare our two saliency models (MDF and HDHF) with a number of existing state-of-the-art methods, including multi-context deep learning (MC) [42], local estimation and global search based deep network (LEGS) [40], single-layer cellular automata (BSCA) [54], pixelwise image saliency aggregating (PISA) [55], discriminative regional feature integration (DRFI) [14], optimized weighted contrast (wCtr*) [35], manifold ranking (MR) [17], global cues (GC) [36], region based contrast (RC) [26], hierarchical saliency (HS) [15], saliency filters (SF) [13], frequency-tuned saliency (FT) [5] and the spectral residual approach (SR) [25]. For GC, RC, FT and SR, we use the implementation provided by the authors of [26]; for other methods, we use their original implementation with recommended parameter settings.

A visual comparison is given in Fig. 9. For space consideration, we only choose the top 8 among all the methods we compare with for this visual demonstration. As can be seen, our models (Fig.9j&k) perform well in a variety of challenging cases, e.g., cluttered background (the first two rows), multiple disconnected salient objects (3-rd and 4-th rows), low contrast between salient object and background (5-th and 6-th rows), and objects touching the image boundary (1-st and 4-th rows). In all the complex scenarios shown in Fig. 9, it is obvious that our models are able to successfully highlight entire salient objects, yielding saliency maps closest to the ground truth.

As part of the quantitative evaluation, we first evaluate our method using Precision-Recall and ROC curves. As shown in Figs. 10, our methods (HDHF and MDF) consistently occupy the top two spots and outperform others on all benchmark datasets. The AUC (Area under ROC curve) is reported in Table III. It is necessary to point out that the performance of MC [42] is overrated on the MSRA-B dataset and the performance of LEGS [40] is overrated on both the MSRA-B dataset and the PASCAL-S dataset because most images in the corresponding datasets were actually training samples for the publicly available trained models of MC [42] and LEGS [40] used in our comparison.

Precision, recall and F-measure values using the aforementioned adaptive threshold are shown in Fig. 11. Our method also achieves the highest precision and F-measure on all datasets. On the DUT-OMRON dataset, HDHF achieves 70.9%

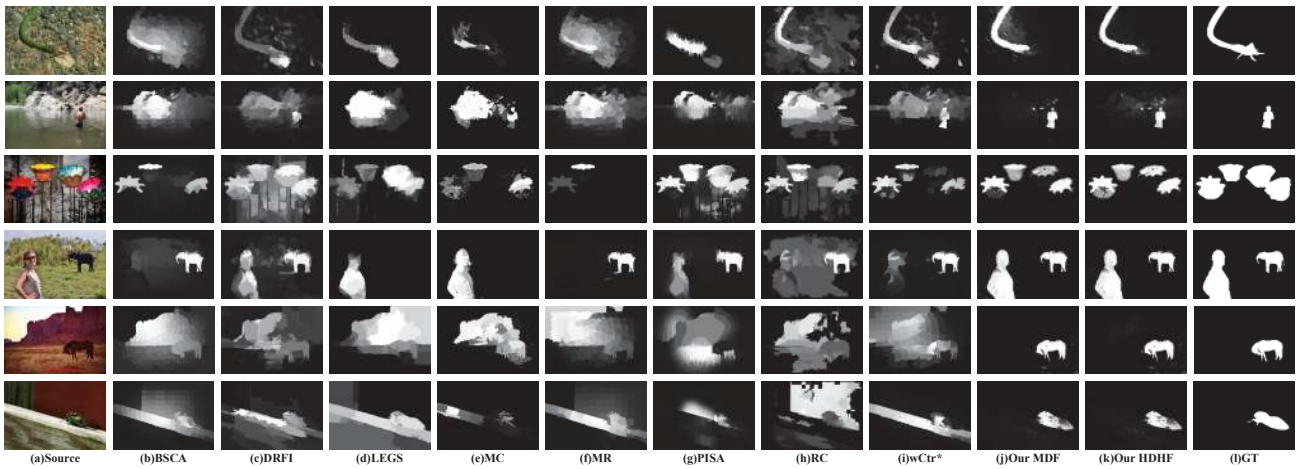


Fig. 9: Visual comparison of saliency maps generated from 10 state-of-the-art methods, including our two models MDF and HDHF. The ground truth (GT) is shown in the last column. MDF and HDHF consistently produce saliency maps closest to the ground truth.

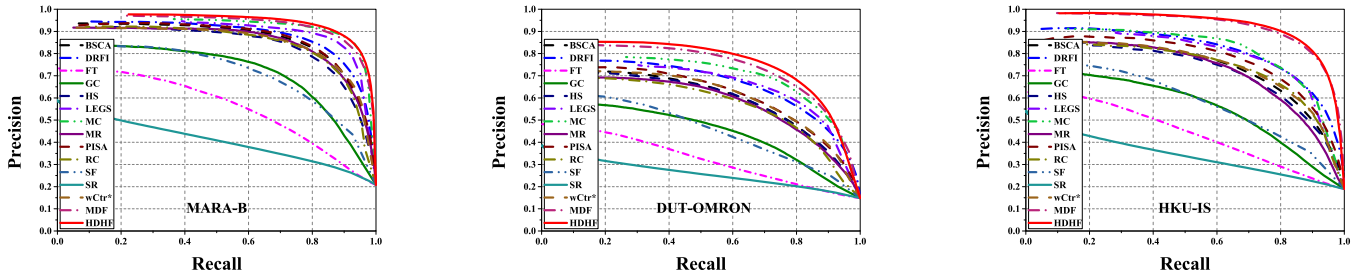


Fig. 10: Comparison of precision-recall curves of 15 saliency detection methods on 3 datasets. Our MDF and HDHF based models consistently outperform other methods across all the testing datasets.

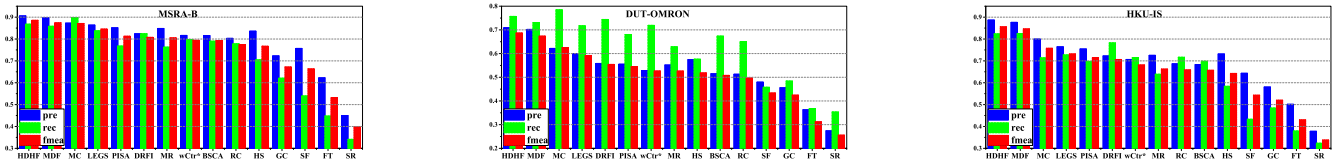


Fig. 11: Comparison of precision, recall and F-measure (computed using a per-image adaptive threshold) among 15 different methods on 3 datasets.

in precision and 75.7% in recall while the second best (MC) achieves 62.2% in precision and 78.5% in recall. Though the recall rate of MC is higher than ours, its precision is much lower. Thus it is much more likely for MC to misclassify unsalient pixels as salient ones. This is also reflected in the lower F-measure and higher MAE achieved with MC. Performance improvement becomes more obvious on HKU-IS. Compared with the second best (MC), our method increases the F-measure from 0.76 to 0.86, and achieves an increase of 10.9% in precision while at the same time improving the recall by 15.3%.

A quantitative comparison is shown in Table III. As can be seen, our HDHF based model improves the F-measure achieved by the best-performing existing algorithm by 6.4%, 2.3%, 10.0%, 6.1%, 5.5% and 8.1% respectively on MSRA-B (skipping MC and LEGS on this dataset), ECSSD, HKU-IS, DUT-OMRON, PASCAL-S (skipping LEGS on this dataset) and SOD. And at the same time, our HDHF based model also outperforms other existing methods in terms of MAE, which provides a better estimation of the visual difference between the predicted saliency map and the ground truth. As shown in Table III, our HDHF based model lowers the MAE by 48.0%, 2.0%, 35.3%, 9.1%, 2.1% and 12.3% respectively on MSRA-B (skipping MC and LEGS on this dataset), ECSSD, HKU-IS, DUT-OMRON, PASCAL-S (skipping LEGS on this dataset) and SOD.

In summary, the improvement our method achieves over the state of the art is substantial if we keep in mind the already good performance of state-of-the-art algorithms. Furthermore, the more challenging the dataset, the more obvious the advantages because our multiscale CNN features are capable of identifying subtle contrast among different parts of an image. More importantly, although our models are learned using the training set of the MSRA-B dataset, they are consistently

Data Set	Metric	BSCA	DRFI	FT	GC	HS	LEGS	MC	MR	PISA	RC	SF	SR	wCtr*	MDF	HDHF
MSRA-B	AUC	0.954	0.966	0.766	0.863	0.930	0.958	**0.975**	0.941	0.954	0.937	0.886	0.710	0.948	**0.978**	**0.982**
	F-measure	0.830	0.845	0.579	0.719	0.813	0.870	**0.894**	0.824	0.837	0.817	0.700	0.430	0.820	**0.888**	**0.899**
	MAE	0.130	0.112	0.241	0.159	0.161	0.081	**0.054**	0.127	0.102	0.138	0.166	0.224	0.110	**0.066**	**0.053**
ECSSD	AUC	0.922	0.943	0.663	0.767	0.885	0.925	**0.948**	0.888	0.921	0.893	0.793	0.632	0.896	**0.957**	**0.960**
	F-measure	0.758	0.782	0.430	0.597	0.727	0.827	**0.837**	0.736	0.764	0.738	0.548	0.376	0.716	**0.847**	**0.856**
	MAE	0.183	0.170	0.289	0.233	0.228	0.118	**0.100**	0.189	0.150	0.186	0.219	0.264	0.171	**0.106**	**0.098**
HKU-IS	AUC	0.911	0.950	0.710	0.777	0.884	0.907	**0.928**	0.870	0.925	0.903	0.828	0.674	0.910	**0.971**	**0.972**
	F-measure	0.723	0.776	0.477	0.588	0.710	0.770	**0.798**	0.714	0.753	0.726	0.590	0.373	0.726	**0.869**	**0.878**
	MAE	0.174	0.167	0.244	0.211	0.213	0.118	**0.102**	0.174	0.127	0.165	0.173	0.220	0.140	**0.072**	**0.066**
DUT-OMRON	AUC	0.882	0.931	0.682	0.757	0.860	0.885	**0.929**	0.853	0.893	0.859	0.810	0.688	0.894	**0.935**	**0.935**
	F-measure	0.617	0.664	0.381	0.495	0.616	0.669	**0.703**	0.610	0.630	0.599	0.495	0.298	0.630	**0.728**	**0.746**
	MAE	0.191	0.150	0.250	0.218	0.227	0.133	**0.088**	0.187	0.141	0.189	0.147	0.181	0.144	**0.088**	**0.080**
PASCAL-S	AUC	0.872	0.899	0.627	0.727	0.838	0.891	**0.907**	0.852	0.866	0.840	0.746	0.671	0.866	**0.921**	**0.922**
	F-measure	0.666	0.690	0.413	0.539	0.641	**0.752**	0.740	0.661	0.660	0.644	0.493	0.392	0.655	**0.771**	**0.781**
	MAE	0.224	0.210	0.309	0.266	0.264	0.157	**0.145**	0.223	0.196	0.227	0.240	0.263	0.201	**0.146**	**0.142**
SOD	AUC	0.843	0.890	0.607	0.692	0.817	0.836	**0.868**	0.812	0.848	0.828	0.714	0.679	0.827	**0.899**	**0.901**
	F-measure	0.654	0.699	0.441	0.526	0.646	**0.732**	0.727	0.636	0.660	0.657	0.516	0.444	0.653	**0.793**	**0.791**
	MAE	0.251	0.223	0.323	0.284	0.283	0.195	**0.179**	0.259	0.223	0.242	0.267	0.291	0.229	**0.157**	**0.160**

TABLE III: Comparison of quantitative results including AUC (larger is better), maximum F-measure (larger is better) and MAE (smaller is better). The best three results are shown in **red**, **blue**, and **green** color, respectively. Note that MC [42] and LEGS [40] are overrated on the MSRA-B dataset and LEGS [40] is overrated on the PASCAL-S dataset.

among the top performers over all other challenging datasets.

H. Efficiency

While it takes around 20 hours to train our deep neural network based prediction model using the training set of the MSRA-B dataset, it only takes around 4 seconds to detect salient objects in a testing image with 400×300 pixels on a PC with two NVIDIA GTX Titan Black GPUs and a 3.4GHz Intel processor using our MATLAB code. Noted that feature extraction efficiency can be improved using multi-GPU techniques provided in the latest Caffe framework [43].

VII. Conclusions and Future Work

In this paper, we have introduced a neural network architecture, which has fully connected layers on top of CNNs responsible for feature extraction at three different scales. The proposed neural network architecture works as a feature learning model which deduces high-level semantic contrast and contextual relationships among the three scales. The penultimate layer of our neural network has been confirmed to be a very discriminative high-level feature vector for saliency detection and is complementary to handcrafted low-level features. To generate a more robust feature, we integrate low-level features with our deep contrast feature and feed the concatenated feature vector into a random forest regressor which maps the feature vector of each region to a saliency score. We aggregate multiple saliency maps computed for different levels of image segmentation to reduce error due to imperfect segmentation, and further incorporate a pixel-level CRF model to enhance spatial coherence. To promote further research and evaluation of visual saliency models, we have also constructed a large dataset of 4447 challenging images and their pixelwise saliency annotations. Experimental results demonstrate that our proposed method significantly outperforms all existing saliency estimation techniques on all public datasets.

As future work, we are considering to improve the efficiency of deep feature extraction. In this paper, we treat each region as an independent unit in feature extraction without any shared computation. We are considering spatial pyramid pooling networks (SPPnets) [56] for speeding up regional feature extraction by computing a single convolutional feature map for an entire image and then extracting all regional features from this shared feature map. We are also considering the application of our deep contrast feature in other pixel labeling problems, e.g. depth prediction from monocular images, eye-fixations and object proposals.

Acknowledgment

The authors would like to thank Sai Bi, Wei Zhang, and Feida Zhu for their help during the construction of our dataset. The first author is supported by Hong Kong Postgraduate Fellowship.

References

[1] G. Li and Y. Yu, "Visual saliency based on multiscale deep features," in *Proc. IEEE Conf. CVPR*, 2015, pp. 5455–5463.
[2] A. Borji and L. Itti, "State-of-the-art in visual attention modeling," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 185–207, 2013.
[3] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 11, pp. 1254–1259, 1998.
[4] T. Liu, Z. Yuan, J. Sun, J. Wang, N. Zheng, X. Tang, and H.-Y. Shum, "Learning to detect a salient object," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 2, pp. 353–367, 2011.
[5] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in *Proc. IEEE Conf. CVPR*, 2009, pp. 1597–1604.
[6] C. Rother, L. Bordeaux, Y. Hamadi, and A. Blake, "Autocollage," *ACM Trans. Graphics*, vol. 25, no. 3, pp. 847–852, 2006.
[7] S. Avidan and A. Shamir, "Seam carving for content-aware image resizing," *ACM Trans. Graphics*, vol. 26, no. 3, 2007.
[8] D. Simakov, Y. Caspi, E. Shechtman, and M. Irani, "Summarizing visual data using bidirectional similarity," in *Proc. IEEE Conf. CVPR*, 2008, pp. 1–8.
[9] L. Marchesotti, C. Cifarelli, and G. Csurka., "A framework for visual saliency detection with applications to image thumbnailing," in *Proc. IEEE Conf. ICCV*, 2009, pp. 2232–2239.
[10] H. Wu, G. Li, and X. Luo, "Weighted attentional blocks for probabilistic object tracking," *The Visual Computer*, vol. 30, no. 2, pp. 229–243, 2014.
[11] R. Wu, Y. Yu, and W. Wang, "Scale: Supervised and cascaded laplacian eigenmaps for visual object recognition based on nearest neighbors," in *Proc. IEEE Conf. CVPR*, 2013, pp. 867–874.
[12] S. Bi, G. Li, and Y. Yu, "Person re-identification using multiple experts with random subspaces," *Journal of Image and Graphics*, vol. 2, no. 2, 2014.
[13] F. Perazzi, P. Krahenbuhl, Y. Pritch, and A. Hornung, "Saliency filters: Contrast based filtering for salient region detection," in *Proc. IEEE Conf. CVPR*, 2012, pp. 733–740.

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

[15] Q. Yan, L. Xu, J. Shi, and J. Jia, "Hierarchical saliency detection," in *Proc. IEEE Conf. CVPR*, 2013, pp. 1155–1162.

[16] M.-M. Cheng, G.-X. Zhang, N. J. Mitra, X. Huang, and S.-M. Hu, "Global contrast based salient region detection," in *Proc. IEEE Conf. CVPR*, 2011, pp. 569–582.

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

[18] L. Itti and C. Koch, "Computational modelling of visual attention," *Nature reviews neuroscience*, vol. 2, no. 3, pp. 194–203, 2001.

[19] P. Reinagel and A. M. Zador, "Natural scene statistics at the centre of gaze," *Network: Computation in Neural Systems*, vol. 10, no. 4, pp. 341–350, 1999.

[20] K. Fukushima, "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," *Biological cybernetics*, vol. 36, no. 4, pp. 193–202, 1980.

[21] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Conf. NIPS*, 2012, pp. 1097–1105.

[22] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, "Learning hierarchical features for scene labeling," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1915–1929, 2013.

[23] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. CVPR*, 2014, pp. 580–587.

[24] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proc. IEEE Conf. CVPR*, 2009, pp. 248–255.

[25] X. Hou and L. Zhang, "Saliency detection: A spectral residual approach," in *Proc. IEEE Conf. CVPR*, 2007, pp. 1–8.

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

[27] Y.-F. Ma and H.-J. Zhang, "Contrast-based image attention analysis by using fuzzy growing," in *Proceedings of the eleventh ACM international conference on Multimedia*. ACM, 2003, pp. 374–381.

[28] J. Harel, C. Koch, and P. Perona, "Graph-based visual saliency," in *Proc. Conf. NIPS*, 2006, pp. 545–552.

[29] X. Shen and Y. Wu, "A unified approach to salient object detection via low rank matrix recovery," in *Proc. IEEE Conf. CVPR*, 2012, pp. 853–860.

[30] C. Chen, S. Li, H. Qin, and A. Hao, "Structure-sensitive saliency detection via multilevel rank analysis in intrinsic feature space," *IEEE Transactions on Image Processing*, vol. 24, no. 8, pp. 2303–2316, 2015.

[31] Y. Jia and M. Han, "Category-independent object-level saliency detection," in *Proc. IEEE Conf. ICCV*, 2013, pp. 1761–1768.

[32] K.-Y. Chang, T.-L. Liu, H.-T. Chen, and S.-H. Lai, "Fusing generic objectness and visual saliency for salient object detection," in *Proc. IEEE Conf. ICCV*, 2011, pp. 914–921.

[33] T. Judd, K. Ehinger, F. Durand, and A. Torralba, "Learning to predict where humans look," in *Proc. IEEE Conf. ICCV*, 2009, pp. 2106–2113.

[34] A. Borji, "Boosting bottom-up and top-down visual features for saliency estimation," in *Proc. IEEE Conf. CVPR*, 2012, pp. 438–445.

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

[36] M.-M. Cheng, J. Warrell, W.-Y. Lin, S. Zheng, V. Vineet, and N. Crook, "Efficient salient region detection with soft image abstraction," in *Proc. IEEE Conf. ICCV*, 2013, pp. 1529–1536.

[37] 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 *Proc. ACM Conf. ICML*, 2014, pp. 647–655.

[38] A. Sharif Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "Cnn features off-the-shelf: an astounding baseline for recognition," in *Proc. IEEE Conf. CVPRW*, 2014, pp. 806–813.

[39] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Hypercolumns for object segmentation and fine-grained localization," in *Proc. IEEE Conf. CVPR*, 2015, pp. 447–456.

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

[41] G. Li and Y. Yu, "Deep contrast learning for salient object detection," in *Proc. IEEE Conf. CVPR*, 2016.

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

[43] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. ACM MM*, 2014, pp. 675–678.

[44] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *IJCV*, vol. 59, no. 2, pp. 167–181, 2004.

[45] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 898–916, 2011.

[46] L. Mai, Y. Niu, and F. Liu, "Saliency aggregation: A data-driven approach," in *Proc. IEEE Conf. CVPR*, 2013, pp. 1131–1138.

[47] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected crfs with gaussian edge potentials," in *Proc. Conf. NIPS*, 2012.

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

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

[50] S. Alpert, M. Galun, R. Basri, and A. Brandt, "Image segmentation by probabilistic bottom-up aggregation and cue integration," in *Proc. IEEE Conf. CVPR*, 2007, pp. 315–327.

[51] D. Batra, A. Kowdle, D. Parikh, J. Luo, and T. Chen, "icoseg: Interactive co-segmentation with intelligent scribble guidance," in *Proc. IEEE Conf. CVPR*, 2010, pp. 3169–3176.

[52] A. Borji, D. N. Sihite, and L. Itti, "Salient object detection: A benchmark," in *Proc. Conf. ECCV*. Springer, 2012, pp. 414–429.

[53] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[54] Y. Qin, H. Lu, Y. Xu, and H. Wang, "Saliency detection via cellular automata," in *Proc. IEEE Conf. CVPR*, 2015, pp. 110–119.

[55] K. Wang, L. Lin, J. Lu, C. Li, and K. Shi, "Pisa: Pixelwise image saliency by aggregating complementary appearance contrast measures with edge-preserving coherence," *IEEE Transactions on Image Processing*, vol. 24, no. 10, pp. 3019–3033, 2015.

[56] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," in *Proc. Conf. ECCV*. Springer, 2014, pp. 346–361.

Guanbin Li received the PhD degree from the University of Hong Kong in 2016. He is currently a research scientist at Sun-Yat Sen University. He is a recipient of Hong Kong Postgraduate Fellowship. His current research interests include computer vision, image processing, and deep machine learning.

Yizhou Yu received the PhD degree from University of California at Berkeley in 2000. He is currently a professor at The University of Hong Kong, and was a faculty member at University of Illinois at Urbana-Champaign for twelve years. He received 2002 US National Science Foundation CAREER Award, and 2007 NNSF China Overseas Distinguished Investigator Award. Prof Yu has served on the editorial board of IEEE Transactions on Visualization and Computer Graphics, The Visual Computer, and International Journal of Software and Informatics. He has also served on the program committee of many leading international conferences, including SIGGRAPH, SIGGRAPH Asia, and International Conference on Computer Vision. His current research interests include deep learning methods for visual computing, digital geometry processing, video analytics and biomedical data analysis.