



End-to-End Saliency Mapping via Probability Distribution Prediction

Saumya Jetley
University of Oxford
Oxford, United Kingdom
sjetley@robots.ox.ac.uk

Naila Murray
XRCE
Meylan, France
naila.murray@xrce.xerox.com

Eleonora Vig
XRCE*
Meylan, France
eleonora.vig@dlr.de

Abstract

Most saliency estimation methods aim to explicitly model low-level conspicuity cues such as edges or blobs and may additionally incorporate top-down cues using face or text detection. Data-driven methods for training saliency models using eye-fixation data are increasingly popular, particularly with the introduction of large-scale datasets and deep architectures. However, current methods in this latter paradigm use loss functions designed for classification or regression tasks whereas saliency estimation is evaluated on topographical maps. In this work, we introduce a new saliency map model which formulates a map as a generalized Bernoulli distribution. We then train a deep architecture to predict such maps using novel loss functions which pair the softmax activation function with measures designed to compute distances between probability distributions. We show in extensive experiments the effectiveness of such loss functions over standard ones on four public benchmark datasets, and demonstrate improved performance over state-of-the-art saliency methods.

1. Introduction

This work is concerned with visual attention prediction, specifically, predicting a topographical visual saliency map when given an input image. Visual attention has been traditionally used in computer vision as a pre-processing step in order to focus subsequent processing on regions of interest in images, an ever-more important step as vision models and datasets increase in size. Saliency map prediction has found useful applications in tasks such as automatic image cropping [39], content aware image resizing [1], image thumb-nailing [29], object recognition [7], and fine-grained scene, and human action classification [36]. Traditional saliency models, such as the seminal work of Itti *et al.* [14], have focused on designing mechanisms to explicitly model biological systems. Another popular attention mod-

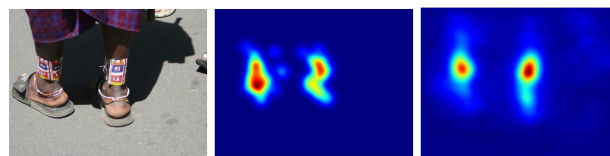


Figure 1. Sample image (left) with ground-truth saliency map (middle) and map predicted by our PDP approach (right).

elling paradigm involves using data-driven approaches to learn patch-level classifiers which give a local image patch a “saliency score” [19, 18], using eye-fixation data to derive training labels. A recent trend has emerged which intersects with both of these paradigms: to use hierarchical models to extract saliency maps, with model weights being learned in a supervised manner. In particular, end-to-end or “deep” architectures, which have been successfully used in semantic labelling tasks such as categorization or object localization, have been re-purposed as attention models [23, 33]. This trend has been facilitated by the introduction of large visual attention datasets created using novel eye movement collection paradigms [16, 43]. However, while these deep methods have focused on designing appropriate architectures for extracting saliency maps, they continue to use loss functions adapted for semantic tasks, such as classification or regression losses.

In this work, we propose a novel formulation of saliency map prediction as a *probability distribution prediction* task. The map is formulated as a generalized Bernoulli distribution, and several novel loss functions are proposed based on probability distance measures. We show that training a deep architecture with such loss functions results in superior performance with respect to standard regression loss functions such as the Euclidean and Huber loss. We also perform a comparison among our proposed loss functions and show that our loss function, based on the Bhattacharyya distance for multinomial distributions, gives top performance.

Our contributions are therefore the following:

- a novel formulation which represents a saliency map as a generalized Bernoulli distribution;

*EV is now at the German Aerospace Center.

- a set of novel loss functions which are paired with the softmax function and which penalize the distance between predicted and target distributions;
- a fully-convolutional architecture which can generate a saliency map for a large image in 200ms using modern GPUs.

Our extensive experimental validation on four datasets demonstrates the effectiveness of our approach when compared to other loss functions and other state-of-the-art approaches to saliency map generation. Figure 1 illustrates its prediction performance.

The remainder of the paper is organized as follows: in section 2 we discuss related work. Section 3 describes our saliency modelling and estimation approach. We report and discuss evaluation results in section 4 and conclude in section 5.

2. Related work

Existing approaches can be organized into one of four broad categories based on whether they involve a *shallow* or *deep* architecture, and an *unsupervised* or *supervised* learning paradigm. We will discuss each of these broad categories in turn. For an excellent survey of saliency estimation methods, please refer to [2].

Unsupervised shallow methods Most early work on saliency builds on psychological and psychophysical models of attention as studied in humans. Koch and Ullman [20] were among the first to use feature integration theory [40] to propose a set of individual topographical maps of elementary cues such as color, contrast, and motion, and combine them to produce a global topographical map of saliency. Their model is implemented using a simple neural circuitry with winner-take-all and inhibition-of-return mechanisms. It is further investigated in [13] by combining features maps over a wider set of modalities (42 such maps) and testing on real-world images. Later approaches largely explore the same idea of complementary feature ensembles [14, 41, 26, 24, 45, 32] and often add to it additional center-surround cues [14, 31, 45].

Complementing the biologically motivated approaches, a number of methods adopt an information-theoretic justification for attentional selection, *e.g.* by self-information [46], information maximization [4], or Bayesian surprise [12]. High computational efficiency is achieved by spectrum-based methods [10, 35]. All these approaches use bottom-up cues, are shallow (one or few layers) and involve no or minimalistic learning of thresholds/heuristics.

Supervised shallow methods This category includes learning based approaches involving models such as markov chains [8], support vector machines [19, 18] and adaboost classifiers [6]. [8] substitutes the idea of centre-

surroundedness and normalization with learnable graph weights. [6], [18], and [48] enrich learning by incorporating top-down semantic cues in the form of detection maps for faces, persons, cars, and the horizon.

Unsupervised hierarchical methods In the context of saliency prediction, the first attempts to employ deeper architectures are mostly unsupervised. [37] learn higher-level concepts from fixated image patches using a 3-layer network of sparse coding units. [42] perform a large-scale search for optimal network architectures of up to three layers, but the network weights are not learned.

DeepGaze [23] employs an existing network architecture, the 5-layer deep AlexNet [21] trained for object classification on ImageNet, to demonstrate that off-the-shelf CNN features can significantly outperform non-deep and “shallower” models, even if not trained explicitly on the task of saliency prediction. Learning, in their case, has meant finding the optimal linear combination of features from the different network layers.

Supervised hierarchical methods The publication of large-scale attention datasets, such as SALICON [16] and TurkerGaze/iSUN [43], has enabled training deep architectures specifically for the task of saliency prediction. Our work lies in this category and involves training an end-to-end deep model with a novel loss function.

SALICON [16] was collected with a new data-collection paradigm, in which observers were shown foveated images and were asked to move the mouse cursor around to simulate the high-resolution fovea. This novel paradigm was used to annotate 20K images from the MSCOCO dataset [25]. Relying on this new large-scale dataset, the authors of [33] trained a network end-to-end for saliency prediction. Their network, titled JuntingNet, consists of five convolutional and two fully-connected layers, and the parameters of the network are learned by minimizing the Euclidean loss function defined on the ground-truth saliency maps. This method reports state-of-the-art results on the LSUN 2015 saliency prediction challenge [47].

Another end-to-end approach that formulates saliency prediction as regression is that of [22]. DeepFix builds upon the very deep VGGNet [38], uses convolutional layers with large and multi-size receptive fields to capture complementary image context, and introduces a location-biased convolutional (LBC) layer to model the center-bias.

Finally, one of the most recent works in this paradigm [11] proposes the use of deep neural networks to bridge the semantic gap in saliency prediction via a two-pronged strategy. The first is the use of the KL-divergence as a loss function motivated by the fact that it is a standard metric for evaluation of saliency methods. The second is the aggregation of response maps from both coarse and fine resolutions.

In this work, we argue for a well-motivated probabilistic modelling of the saliency maps and hence study the use of KL-divergence, among other probability distance measures, as loss functions. As we discuss in section 4, we observe that our Bhattacharyya distance-based loss function consistently outperforms the KL-divergence-based one across 4 standard saliency metrics.

3. Saliency maps as probability distributions

Saliency estimation methods have typically sought to model local saliency based on conspicuity cues such as local edges or blob-like structures, or on the scores of binary saliency classifiers trained on fixated and non-fixated image patches. More recently, methods have sought to directly predict maps using pixel-wise regression.

However, visual attention is a fundamentally stochastic process due to it being a perceptual and therefore subjective phenomenon. In an analysis of 300 images viewed by 39 observers, the authors of [17] find that the fixations for a set of n observers match those from a different set of n observers with an AUC score that increases with the increase in the value of n . The lower bound of human performance is found to be 85% AUC. Therefore there is high consistency across observers. At the limit of $n \rightarrow \infty$ this AUC score is 92%, which can therefore be considered a realistic upper-bound for saliency estimation performance.

Ground-truth saliency maps are constructed from the aggregated fixations of multiple observers, ignoring any temporal fixation information. Areas with a high fixation density are interpreted as receiving more attention. As attention is thought to be given to a localized region rather than an exact pixel, two-dimensional Gaussian filtering is typically applied to a binary fixation map to construct a smooth “attentional landscape” [44] (c.f. Figure 1, middle image for an example). Our goal is to predict this attentional landscape, or saliency map. Given the stochastic nature of the fixations upon which the maps are based, and the fact that the maps are based on aggregated fixations without temporal information, we propose to model a saliency map as a probability distribution over pixels, where each value corresponds to the probability of that pixel being fixated upon. That is, we represent a saliency map as a generalized Bernoulli distribution $\mathbf{p} = (p_1, \dots, p_i, \dots, p_N)$, where \mathbf{p} is the probability distribution over a set of pixels forming an image, p_i is the probability of pixel i being fixated upon and N is the number of image pixels. While this formulation is somewhat simplistic, it will allow for novel loss functions highly amenable to training deep models with back-propagation. In the sequel, we first describe these loss functions and then describe our model implementation.

3.1. Learning to predict the probability of fixation

We adopt an end-to-end learning framework in which a fully-convolutional network is trained on pairs of images and ground-truth saliency maps \mathbf{g} modeled as distributions. The network outputs predicted distributions \mathbf{p} ¹. Both probability distributions, \mathbf{g} and \mathbf{p} , are computed using the softmax activation function:

$$p_i = \frac{e^{x_i^p}}{\sum_j e^{x_j^p}}, \quad g_i = \frac{e^{x_i^g}}{\sum_j e^{x_j^g}}, \quad (1)$$

where $\mathbf{x} = (x_1, \dots, x_i, \dots, x_N)$ is the set of unnormalized saliency response values for either the ground-truth map (\mathbf{x}^g) or the predicted map (\mathbf{x}^p). To compute \mathbf{x}^g , a binary fixation map \mathbf{b} is first generated from ground-truth eye-fixations. The binary map \mathbf{b} is then convolved with a Gaussian kernel as described earlier in this section to produce \mathbf{y} . The smoothed map \mathbf{y} is then normalized as

$$x_i^g = \frac{y_i - \min[\mathbf{y}]}{\max[\mathbf{y}] - \min[\mathbf{y}]}. \quad (2)$$

We generate \mathbf{x}^p directly from the last response map of our deep network, whose architecture is described in the next section.

We propose to combine the softmax function with distance measures appropriate for probability distributions in order to construct objective functions to be used for training the network. This combination is inspired by the popular and effective softmax/cross-entropy loss pairing which is often used to train models for multinomial logistic regression.

In our case, we propose to combine the softmax functions with the χ^2 , total-variation, cosine and Bhattacharyya distance measures, as listed in Table 1. To our knowledge, these pairings have not previously been used to train a network for probability distribution prediction. We also investigate the use of the KL divergence measure, the minimization of which is equivalent to cross-entropy minimization, and which is used extensively to learn regression models in deep networks. The partial derivatives of these loss functions with respect to x_i^p are all of the form $ap_i - b(1 - p_i)$ due to the pairing with the softmax function, whose partial derivative with respect to x_i^p is

$$\frac{\partial p_j}{\partial x_i^p} = \begin{cases} p_i(1 - p_i), & \text{if } j = i \\ -p_i p_j, & \text{otherwise.} \end{cases} \quad (3)$$

We make comparisons with two standard regression losses, the Euclidean and Huber losses, defined as:

$$L_{euc}(\mathbf{p}, \mathbf{g}) = \sum_j a_j^2, \quad (4)$$

¹We slightly abuse notation and from now on use \mathbf{p} to refer specifically to the predicted distribution.

Probability distances	$L(\mathbf{p}, \mathbf{g})$	$\frac{\partial L(\mathbf{p}, \mathbf{g})}{\partial x_i^p}$
χ^2 divergence	$\sum_j \frac{(g_j)^2}{p_j} - 1$	$p_i \sum_{j \neq i} \frac{g_j^2}{p_j} - \frac{g_i^2}{p_i} (1 - p_i)$
Total Variation distance	$\frac{1}{2} \sum_j g_j - p_j $	$\frac{1}{2} \left[p_i \sum_{j \neq i} \frac{g_j - p_j}{ g_j - p_j } p_j - p_i \frac{g_i - p_i}{ g_i - p_i } (1 - p_i) \right]$
Cosine distance	$1 - \frac{\sum_j p_j g_j}{\sqrt{\sum_j p_j^2} \sqrt{\sum_j g_j^2}}$	$\frac{1}{C} \left[p_i \sum_{j \neq i} p_j (g_j - p_i \frac{\sqrt{\sum_i g_i^2}}{\sqrt{\sum_i p_i^2}} R) - p_i (g_i - p_i R) (1 - p_i) \right];$ where $R = \frac{\sum_i p_i g_i}{C}$ and $C = \sqrt{\sum_i p_i^2} \sqrt{\sum_i g_i^2}$.
Bhattacharyya distance	$-\ln \sum_j (p_j g_j)^{0.5}$	$\frac{-1}{2 \sum_j (p_j g_j)^{0.5}} \left[p_i \sum_{j \neq i} (p_j g_j)^{0.5} - (p_i g_i)^{0.5} (1 - p_i) \right]$
KL divergence	$\sum_j g_j \log \frac{g_j}{p_j}$	$p_i \sum_{j \neq i} g_j - g_i (1 - p_i)$

Table 1. Probability distance measures and their derivatives used for stochastic gradient descent with back-propagation. We propose the use of the first 4 measures as loss functions. We also investigate KL-divergence, which is widely used to train recognition models in the form of the closely-related cross-entropy loss.

and

$$L_{hub}(\mathbf{p}, \mathbf{g}) = \sum_j \begin{cases} \frac{1}{2} a_j^2, & \text{for } |a_j| \leq 1 \\ |a_j| - \frac{1}{2}, & \text{otherwise;} \end{cases} \quad (5)$$

where $a_j = |p_j - g_j|$.

3.2. Training the prediction model

The network architecture and saliency map extraction pipeline is shown in Figure 2. We use the convolutional layers of the VGGNet model [38], which were trained on ImageNet images for the task of classification, as the early layers of our model. This convolution sub-network has been shown to provide good local feature maps for a variety of different tasks including object localization [34] and semantic segmentation [27]. As saliency datasets tend to be much too small to train such large networks from random initializations (the largest dataset has 15000 images, compared to 1M for ImageNet), it is essential to initialize with a pre-trained network. We then progressively decrease the number of feature maps using additional convolutional layers, until a final down-sampled saliency map is produced. We add three new layers, rather than just one, to predict the final map in order to improve both discriminability and generalizability [38]. We experimented with different filter sizes besides 7×7 (e.g. 9×9 , 5×5 , 3×3) and found no significant performance difference. We explicitly avoided fully-connected layers in order to obtain a memory and time-efficient model. The three new layers are initialised with a uniform Gaussian distribution of $\sigma = 0.01$. Because the response maps undergo several max-pooling operations, the predicted saliency map \mathbf{p} is lower-resolution than the input image. The ground-truth map \mathbf{g} is therefore downsampled during training to match the dimensions of \mathbf{p} . Conversely, during inference the predicted map is upsampled with a bi-linear filter to match the dimensions of the input image (see Figure 2), and the softmax function is applied for normalization to a probability distribution.

The final fully-convolutional network comprises 16 convolutional layers, each of which is followed by a ReLU layer. Due to the fully-convolutional architecture, the size is quite small for a deep model, with only 15,530,481 weights (60MB of disk space).

Note that while several deep saliency models explicitly include a center bias (see e.g. [22]), we hypothesized that the model could learn the center-bias implicitly, given that it is largely an artifact of a composition bias in which photographers tend to place highly salient objects in the image center [3]. We tested this by adding Gaussian blurring and a center-bias to our maps, with optimized parameters, using the post-processing code of the MIT saliency benchmark [5]. We found no consistent improvement across different metrics using this post-processing which indicates that a great deal of center-bias and Gaussian blurring is already accounted for in the model.

The objective function is optimized using stochastic gradient descent, with a learning rate of 1 times the global learning rate for newly-introduced layers and 0.1 times the global learning rate for those layers which have been pre-trained on ImageNet. To reduce training time, the first 4 convolutional layers were fixed and thus retained their pre-trained values. We used a momentum of 0.9 and a weight decay of 0.0005. The model is implemented in Caffe [15]. We trained the network using an Nvidia K40 GPU. Training on the SALICON training set took 30 hours.

Saliency datasets tend to have semantic biases and other idiosyncrasies related to the complexity of collecting eye-tracking information (such as the viewing distance to the screen and the eye-tracker calibration). For this reason, we perform dataset-specific fine-tuning, which improves performance. Fine-tuning is particularly essential in our case because the SALICON dataset collected mouse clicks in lieu of actual eye-fixations which, while highly correlated in general, are still an approximation to true human eye movements. As shown on a subset of the SALICON images, image-level conformance between SALICON fixations and

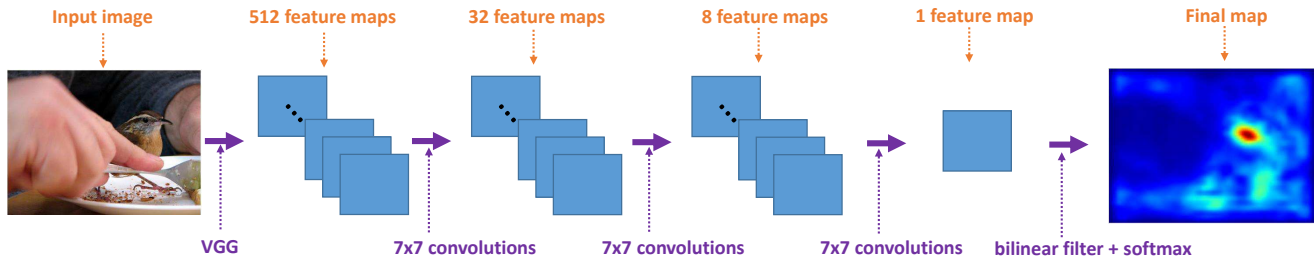


Figure 2. Our proposed saliency map extraction pipeline: the input image is introduced into a convNet with an identical architecture to the convolutional-layer portion of VGGNet. Additional convolutional layers are then applied, resulting in a single response map which is upsampled and softmax-normalized at testing time to produce a final saliency map.

human eye fixations can be as low as shuffled AUC (sAUC) of 0.655 and as high as sAUC of 0.965 [16]. Therefore it is beneficial to fine-tune the network for each dataset of interest. A detailed description of each of these datasets follows.

4. Experimental evaluation

This section describes the experimental datasets used for training and evaluating the saliency prediction models followed by a discussion on the quantitative and qualitative aspects of the results.

4.1. Datasets

SALICON This is one of the largest saliency datasets available in the public domain [16]. It consists of eye-fixation information for 20000 images from the MS COCO dataset [25]. These images contain diverse indoor and outdoor scenes and display a range of scene clutter. 10000 images are marked for training, 5000 for validation and 5000 for testing. The fixation data for the test set is held-out and performance on it must be evaluated on a remote server. The peculiarity of SALICON lies in its mouse-based paradigm for fixation gathering. The attentional focus (foveation) in the human attention mechanism that defines saliency fixations is simulated using mouse-movements over a blurred image. The approximate foveal image region around the mouse position is selectively un-blurred as the user explores the image scene using the mouse cursor. As evaluated on a subset of the dataset, this mouse-click data is in general highly consistent with human eye fixations (at 0.89 sAUC). Therefore, while the mouse fixation data is an approximation to the human baseline, it is useful in adapting the weights of a deep network originally trained for a distinct task to the new task of saliency prediction. We use this dataset for our comparative study of the selected probability distances as loss functions during learning. We have also submitted our best performing model to the SALICON challenge server [47].

MIT-1003 This dataset was introduced as part of the train-

ing and testing paradigm in [18]. The eye tracking data is collected using a head-mounted eye tracking device for 15 different viewers. The 1003 images of this dataset cover natural indoor and outdoor scenes. For our experiments, we use the first 900 images for training and the remaining 103 for validation, similar to the paradigm of [22].

MIT-300 This benchmark consists of held-out eye tracking data for 300 images collected across 39 different viewers [17]. The data collection paradigm for this dataset is very similar to that used in MIT-1003. Hence, as suggested on the online benchmark, we use MIT-1003 as the training data to fine-tune for MIT-300.

OSIE This benchmark contains a set of 700 images. These include natural indoor and outdoor scenes, as well as high aesthetic-quality pictures taken from Flickr and Google. In order to gain from top-down understanding, this dataset provides object and semantic level information (which we do not use) along with the eye-tracking data. Following the work of [28], we randomly divide the set into 500 training and 200 test images and average the results over a 10-fold cross-validation.

VOCA-2012 With the exception of SALICON, the previous datasets are relatively small, with at most 1003 images. Evaluations on large-scale datasets of real fixations would be more informative. However, to our knowledge, there is no truly large-scale dataset of free-viewing fixations. Instead, we evaluate on VOCA-2012, an action recognition dataset which has been augmented with task-dependent eye-fixation data [30]. Predicting such fixations is a different task to predicting free-viewing fixations, the task for which our model is designed. We therefore evaluate on this dataset to determine whether our model generalizes to this task.

Generating ground-truth maps To create ground-truth saliency maps from fixation data, we use the saliency map generation parameters established by the authors of each dataset. For SALICON, this means convolving the binary fixation maps with a Gaussian kernel of width 153 and standard deviation 19. For OSIE, this means applying a Gaus-

Distance	AUC-Judd	sAUC	CC	NSS
Euclidean	0.865	0.761	0.667	2.108
Huber	0.867	0.766	0.684	2.177
KL divergence	0.876	0.780	0.724	2.371
χ^2 divergence	0.872	0.774	0.711	2.337
Total Variation distance	0.869	0.766	0.716	2.385
Cosine distance	0.871	0.778	0.717	2.363
Bhattacharyya distance	0.880	0.783	0.740	2.419

Table 2. SALICON validation set: Performance comparison of models trained using different loss functions.

sian kernel of width of 168 and standard deviation of 24 (all in units of pixels). The authors of MIT-1003 and MIT-300 provide ground-truth saliency maps which, according to their technical report [17], are computed with a Gaussian kernel whose size corresponds to a cutoff frequency of 8 cycles per image.

4.2. Results

We first compare results for different loss functions and then compare to the state-of-the-art methods. For each dataset, we follow the established evaluation protocol and report results on standard saliency metrics, including sAUC, AUC-Judd, AUC-Borji, Correlation Coefficient (CC), Normalized Scanpath Saliency (NSS), Similarity (SIM), and Earth Mover’s Distance (EMD).

Loss functions We compare the performance of models trained using our proposed loss functions to those trained on standard loss functions based on the Euclidean distance, Huber distance, and KL-divergence measure. These models are all trained on the SALICON training set of 10K images, and validated on the SALICON validation set of 5K images. Table 2 presents the best validation-set performance for each loss, as measured by the overall performance with respect to 4 metrics. These results show that: (i) the losses based on distance measures appropriate for probability distributions perform better than standard regression losses; (ii) the KL-divergence compares favorably with other methods; and (iii) the Bhattacharyya distance-based loss outperforms all other losses. These two last losses share the property that they are robust to outliers as they suppress large differences between probabilities (logarithmically in the case of the KL divergence and geometrically in the case of the Bhattacharyya distance). This robustness is particularly important as the ground-truth saliency maps are derived from eye-fixations which have a natural variation due to the subjectivity of visual attention, and which may also contain stray fixations and other noise. Figure 3 shows the evolution of the saliency metrics on the SALICON validation set as the training progresses. The Bhattacharyya distance is consistently the best-performing.

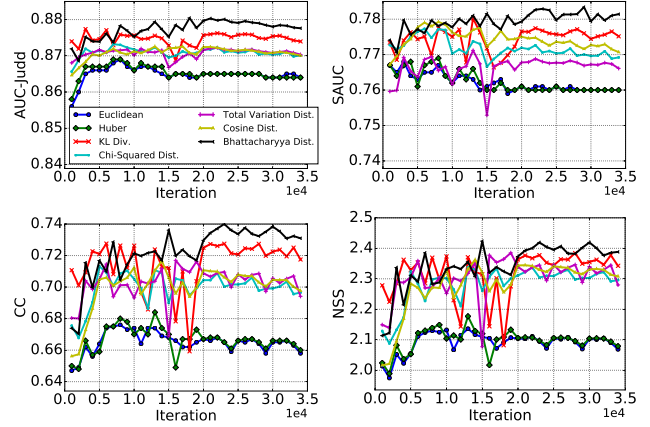


Figure 3. Evolution of different metrics on the evaluation set of SALICON as the number of training iterations increases.

Method	CC	sAUC	AUC-Borji
Itti[14]	0.205	0.610	0.660
GBVS[8]	0.421	0.630	0.782
BMS[45]	0.427	0.694	0.770
WHU_IIP*	0.457	0.606	0.776
Xidian*	0.481	0.681	0.800
Rare12_Improved*	0.511	0.664	0.805
UPC[33]	0.596	0.670	0.829
PDP	0.765	0.781	0.882

Table 3. SALICON Challenge: comparison between different methods. Methods marked by * have no associated publication to-date.

Comparison to the state of the art We compare the performance of our proposed model, using the Bhattacharyya distance, with the state-of-the-art methods for four standard saliency benchmarks as follows.

SALICON challenge: The saliency estimation challenge [47] consists in predicting saliency maps for 5000 images held out from the SALICON dataset. Table 3 shows results for state-of-the-art methods and our approach, which we call PDP for probability distribution prediction. We outperform all published results, to our knowledge, on this dataset across all three metrics.

MIT-300: MIT-1003 images serve as the training set for fine-tuning to this benchmark. The results are compared in Table 4. We perform comparably to the state-of-the-art methods. Note that DeepFix [22] incorporates external cues such as center and horizon biases in its models. We believe that including such cues may also improve our model. In addition, they use a larger architecture, but train with a regression loss. Therefore our approach may complement theirs. Fine-tuning on MIT-1003 could only be performed using a batch size of 1 image due to the large variations in size and aspect ratio of the images. We observed that a much-reduced momentum of 0.70 improved stability and

Method	AUC-Judd	SIM	EMD	AUC-Borji	sAUC	CC	NSS
eDN[42]	0.82	0.41	4.56	0.81	0.62	0.45	1.14
BMS[45]	0.83	0.51	3.35	0.82	0.65	0.55	1.41
SALICON[11]	0.87	0.60	2.62	0.85	0.74	0.74	2.12
DeepFix[22]	0.87	0.67	2.04	0.80	0.71	0.78	2.26
PDP	0.85	0.60	2.58	0.80	0.73	0.70	2.05

Table 4. MIT-300: comparison with the state of the art.

Method	sAUC
Itti[14]	0.658
SUN[46]	0.735
Signature[9]	0.749
GBVS[8]	0.706
LCQS-baseline[28]	0.765
PDP	0.797

Table 5. OSIE: The performance metric of shuffled AUC (sAUC) is averaged over 10-fold cross validation. (Baseline results are taken from [28].)

allowed for an effective learning of the model with this constraint.

OSIE benchmark: The performance comparison on this dataset is done using 10-fold cross validation by randomly dividing the dataset into 500 training and 200 validation images. Table 5 shows that PDP achieves the highest sAUC score. This dataset contains a wide variety of image content and aesthetic properties. Nonetheless, this small set of 500 images was sufficient to successfully adapt our model.

VOCA-2012 (Generalization to task-dependent fixation prediction): We ran experiments on the VOCA-2012 dataset using the same experimental paradigm as in [30]. We used our final SALICON-trained model to predict maps for test images both before and after fine-tuning the model on training images from VOCA-2012. The results summarized in Table 6 show that our method, both with and without finetuning, outperforms the state-of-the-art [30]. This suggests that the task-dependent fixations for this action recognition dataset are highly consistent with free-viewing fixations.

4.3. Discussion

Our probabilistic perspective to saliency estimation is intuitive in two ways. First, attention is competitive as we look at certain regions in the image at the expense of others. Hence, the fixation map normalised over the total visual stimulus can be understood as a spatial probability distribution. Secondly, a probabilistic framework allows the model to account for the noise across subjects and over the data collection paradigm.

To provide qualitative insight, some randomly-chosen predicted maps are shown in Figure 4. Our method consistently gives high fixation probabilities to areas of high

Method	KL	AUC
HOG detector* [30]	8.54	0.736
Judd et al.* [18]	11.00	0.715
Itti & Koch [13]	16.53	0.533
central bias [30]	9.59	0.780
human [30]	6.14	0.922
PDP(without finetuning)	7.92	0.845
PDP*(with finetuning)	8.23	0.875

Table 6. VOCA: Performance comparison on KL-divergence and AUC measures. Note that the best performance is achieved by using the fixations of one human observer to predict those of the remaining observers. The results in bold indicate the best-performing methods that do not require human intervention at testing time. (* denotes the methods that have been trained on this particular dataset.)

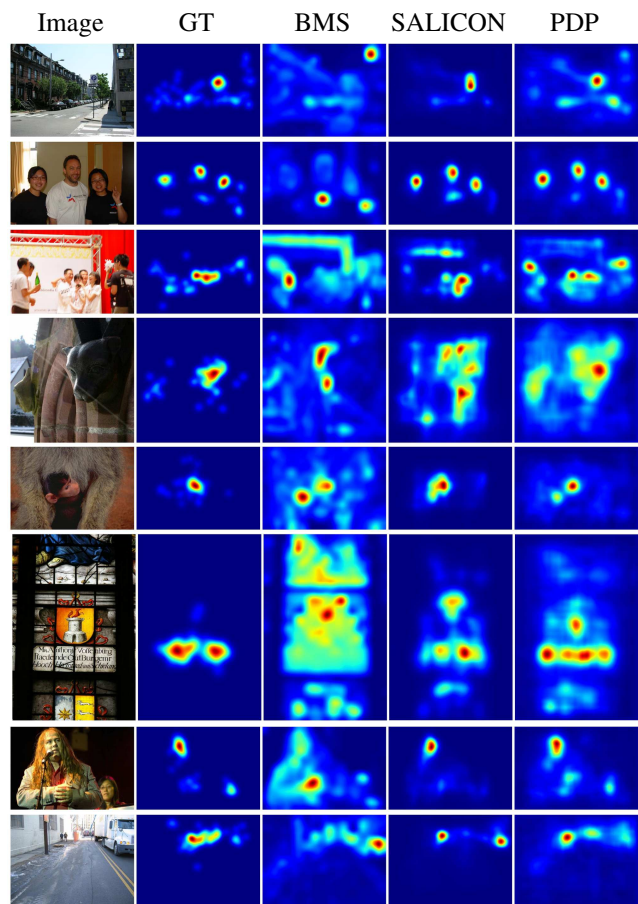


Figure 4. Comparison of BMS, SALICON, and our proposed PDP method for randomly-sampled images from MIT-1003. GT refers to the ground-truth saliency maps. Note that, to ensure a fair comparison, the PDP results shown here were obtained from a network that was trained only on SALICON images, with no fine-tuning to this dataset.

center-surround contrast, and also to high-level cues such as bodies, faces and, to a lesser extent, text. The higher em-

phasis on bodies and faces as compared to text is likely due to the large number of images containing people and faces in the SALICON dataset.

Figure 5 shows saliency map predictions for SALICON training images which were obtained on the forward pass after a given number of training images had been used to train the model. One can see that center-surround contrast cues are learned very quickly, after having seen fewer than 50 images. Faces (both of animate and non-animate objects) are also learned quickly, having seen fewer than 100 images. The saliency of text also emerges fairly rapidly. However, the cue is not as strongly identified, likely due to the relatively smaller amount of training data involving text.

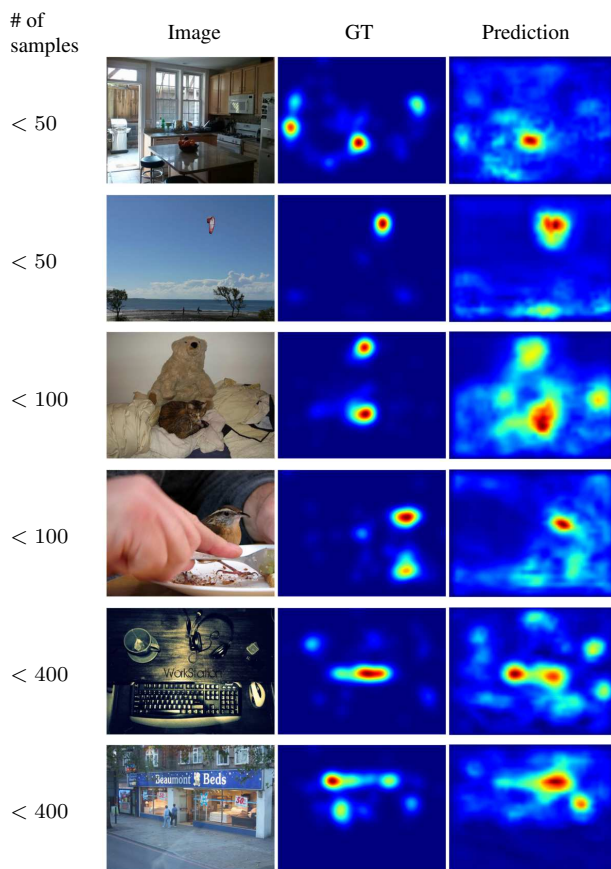


Figure 5. Our method quickly learns that regions of high center-surround contrast, and faces and heads, are salient.

5. Conclusion

We introduce a novel saliency formulation and model for predicting saliency maps given input images. We train a deep network using an objective function which penalizes the distance between target and predicted maps in the form of probability distributions. Experiments on four datasets demonstrate the superior performance of our method with

respect to other loss functions and other state-of-the-art saliency estimation methods. They also illustrate the benefit of using suitable learning criteria adapted to this task.

Acknowledgements

This work was partly funded by the ERC grant ERC-2012-AdG (321162-HELIOS).

References

- [1] R. Achanta and S. Ssstrunk. Saliency detection for content-aware image resizing. In *ICIP*, 2009. 1
- [2] A. Borji and L. Itti. State-of-the-art in visual attention modeling. *TPAMI*, 2013. 2
- [3] A. Borji and J. Tanner. Reconciling saliency and object center-bias hypotheses in explaining free-viewing fixations. *IEEE TNLS*, 2015. 4
- [4] N. Bruce and J. Tsotsos. Saliency based on information maximization. In *NIPS*, 2006. 2
- [5] Z. Bylinskii, T. Judd, A. Borji, L. Itti, F. Durand, A. Oliva, and A. Torralba. Mit saliency benchmark. <http://saliency.mit.edu/>. 4
- [6] M. Cerf, J. Harel, W. Einhuser, and C. Koch. Predicting human gaze using low-level saliency combined with face detection. In *NIPS*, 2008. 2
- [7] S. O. Gilani, R. Subramanian, Y. Yan, D. Melcher, N. Sebe, and S. Winkler. PET: An eye-tracking dataset for animal-centric pascal object classes. In *ICME*, 2015. 1
- [8] J. Harel, C. Koch, and P. Perona. Graph-based visual saliency. In *NIPS*, 2006. 2, 6, 7
- [9] X. Hou, J. Harel, and C. Koch. Image signature: Highlighting sparse salient regions. *TPAMI*, 2012. 7
- [10] X. Hou and L. Zhang. Saliency detection: A spectral residual approach. In *CVPR*, 2007. 2
- [11] X. Huang, C. Shen, X. Boix, and Q. Zhao. Salicon: Reducing the semantic gap in saliency prediction by adapting deep neural networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 262–270, 2015. 2, 7
- [12] L. Itti and P. F. Baldi. Bayesian surprise attracts human attention. In *NIPS*, 2006. 2
- [13] L. Itti and C. Koch. A saliency-based search mechanism for overt and covert shifts of visual attention. *Vision research*, 40(10):1489–1506, 2000. 2, 7
- [14] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *TPAMI*, (11):1254–1259, 1998. 1, 2, 6, 7
- [15] 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 *ACM MM*, 2014. 4
- [16] M. Jiang, S. Huang, J. Duan, and Q. Zhao. SALICON: Saliency in Context. In *CVPR*, 2015. 1, 2, 5
- [17] T. Judd, F. Durand, and A. Torralba. A benchmark of computational models of saliency to predict human fixations. In *MIT Technical Report*, 2012. 3, 5, 6

- [18] T. Judd, K. Ehinger, F. Durand, and A. Torralba. Learning to predict where humans look. In *CVPR*, 2009. 1, 2, 5, 7
- [19] W. Kienzle, F. A. Wichmann, B. Schölkopf, and M. O. Franz. A Nonparametric Approach to Bottom-Up Visual Saliency. In *NIPS*, 2007. 1, 2
- [20] C. Koch and S. Ullman. Shifts in selective visual attention: towards the underlying neural circuitry. In *Matters of intelligence*, pages 115–141. 1987. 2
- [21] A. Krizhevsky, I. Sutskever, and G. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012. 2
- [22] S. S. S. Kruthiventi, K. Ayush, and R. V. Babu. Deepfix: A fully convolutional neural network for predicting human eye fixations. Technical report, 2015. arXiv:1510.02927. 2, 4, 5, 6, 7
- [23] M. Kümmerer, L. Theis, and M. Bethge. Deep Gaze I: Boosting Saliency Prediction with Feature Maps Trained on ImageNet. In *ICLR Workshop*, 2015. 1, 2
- [24] C. Lang, T. Nguyen, H. Katti, K. Yadati, M. Kankanhalli, and S. Yan. Depth matters: Influence of depth cues on visual saliency. In *ECCV*, 2012. 2
- [25] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, 2014. 2, 5
- [26] Z. Liu, O. Le Meur, S. Luo, and L. Shen. Saliency detection using regional histograms. *Optics letters*, 38(5):700–702, 2013. 2
- [27] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, June 2015. 4
- [28] Y. Luo, Y. Wong, and Q. Zhao. Label consistent quadratic surrogate model for visual saliency prediction. In *CVPR*, 2015. 5, 7
- [29] L. Marchesotti, C. Cifarelli, and G. Csurka. A framework for visual saliency detection with applications to image thumbnailing. In *ICCV*, 2009. 1
- [30] S. Mathe and C. Sminchisescu. Action from still image dataset and inverse optimal control to learn task specific visual scanpaths. In *NIPS*, 2013. 5, 7
- [31] N. Murray, M. Vanrell, X. Otazu, and C. A. Parraga. Saliency estimation using a non-parametric low-level vision model. In *CVPR*, 2011. 2
- [32] N. Murray, M. Vanrell, X. Otazu, and C. A. Parraga. Low-level spatiochromatic grouping for saliency estimation. *TPAMI*, 2013. 2
- [33] J. Pan and X. G. i Nieto. End-to-end convolutional network for saliency prediction. Technical report, 2015. arXiv:1507.01422. 1, 2, 6
- [34] S. Ren, K. He, R. B. Girshick, and J. Sun. Faster R-CNN: towards real-time object detection with region proposal networks. *CoRR*, abs/1506.01497, 2015. 4
- [35] B. Schauerte and R. Stiefelhagen. Quaternion-based spectral saliency detection for eye fixation prediction. In *ECCV*, pages 116–129, 2012. 2
- [36] G. Sharma, F. Jurie, and C. Schmid. Discriminative spatial saliency for image classification. In *CVPR*, 2012. 1
- [37] C. Shen, M. Song, and Q. Zhao. Learning high-level concepts by training a deep network on eye fixations. In *NIPS Deep Learning and Unsupervised Feature Learning Workshop*, 2012. 2
- [38] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 2, 4
- [39] F. Stentiford. Attention based auto image cropping. In *The 5th International Conference on Computer Vision Systems, Bielefeld*, 2007. 1
- [40] A. M. Treisman and G. Gelade. A feature-integration theory of attention. *Cognitive psychology*, 12(1):97–136, 1980. 2
- [41] R. Valenti, N. Sebe, and T. Gevers. Image saliency by isocentric curvedness and color. In *ICCV*, 2009. 2
- [42] E. Vig, M. Dorri, and D. Cox. Large-scale optimization of hierarchical features for saliency prediction in natural images. In *CVPR*, 2014. 2, 7
- [43] P. Xu, K. A. Ehinger, Y. Zhang, A. Finkelstein, S. R. Kulkarni, and J. Xiao. Turkergaze: Crowdsourcing saliency with webcam based eye tracking. Technical report, 2015. arXiv:1504.06755v1. 1, 2
- [44] W. H. Zangemeister, H. Stiehl, and C. Freksa. *Visual attention and cognition*. Elsevier, 1996. 3
- [45] J. Zhang and S. Sclaroff. Saliency detection: A boolean map approach. In *ICCV*, 2013. 2, 6, 7
- [46] L. Zhang, M. H. Tong, T. K. Marks, H. Shan, and G. W. Cottrell. SUN: A Bayesian framework for saliency using natural statistics. *JoV*, 8(7):1–20, 12 2008. 2, 7
- [47] Y. Zhang, F. Yu, S. Song, P. Xu, A. Seff, and J. Xiao. Large-scale scene understanding challenge. <http://lsun.cs.princeton.edu/leaderboard/#saliencysalicon>. 2, 5, 6
- [48] Q. Zhao and C. Koch. Learning a saliency map using fixated locations in natural scenes. *JoV*, 11(3), 2011. 2