Exemplar-Driven Top-Down Saliency Detection via Deep Association

Shengfeng He and Rynson W.H. Lau

Department of Computer Science
City University of Hong Kong
http://www.shengfenghe.com/exemplarsaliency.html

Abstract

Top-down saliency detection is a knowledge-driven search task. While some previous methods aim to learn this "knowledge" from category-specific data, others transfer existing annotations in a large dataset through appearance matching. In contrast, we propose in this paper a locate-by-exemplar strategy. This approach is challenging, as we only use a few exemplars (up to 4) and the appearances among the query object and the exemplars can be very different. To address it, we design a two-stage deep model to learn the intra-class association between the exemplars and query objects. The first stage is for learning object-to-object association, and the second stage is to learn background discrimination. Extensive experimental evaluations show that the proposed method outperforms different baselines and the category-specific models. In addition, we explore the influence of exemplar properties, in terms of exemplar number and quality. Furthermore, we show that the learned model is a universal model and offers great generalization to unseen objects.

1. Introduction

The human visual system has an outstanding ability to rapidly locate salient regions in complex scenes [20]. Our attention is mainly drawn by factors relevant to either bottom-up or top-down saliency detection. Bottom-up visual saliency is stimulus-driven, and thus sensitive to the most interesting and conspicuous regions in the scene. Top-down visual saliency, on the other hand, is knowledge-driven and involves high-level visual tasks, such as intentionally looking for a specific object.

In computer vision, bottom-up saliency detection [19, 37, 36, 16, 17, 41, 40] receives much research attention, due to its task-free nature. For the same reason, it can only capture the most salient object(s) in the scene. On the other hand, top-down saliency [21, 22, 12, 38, 23, 7] aims to locate all the intended objects in the scene, which can help reduce the search space for object detection. Existing methods typically learn the "knowledge" that guides top-down saliency detection, from a set of categorized training data. Thus, they are confined to the pre-defined categories and restricted from training a universal model.

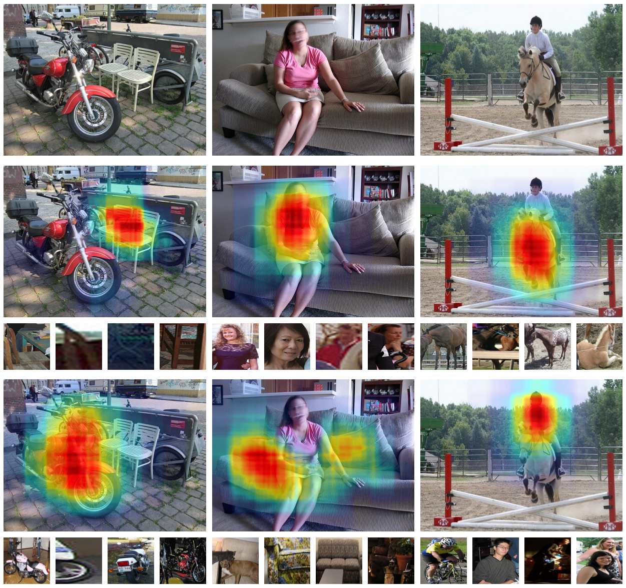

(a) Chair & motorbike (b) Person & sofa (c) Horse & person

Figure 1: The proposed method can locate top-down saliency using a few exemplars (shown under each saliency map), even though there are significant differences among them.

However, human knowledge does not only come from memory (i.e., locating salient objects in the scene using knowledge from training data), but also from object association (i.e., locating objects in the scene using known or unknown exemplars) [2]. For example, we can easily identify and locate a similar object in an image when given an unseen example object in another. As suggested in cognitive studies [27, 2], instead of recognizing an object according to an explicit category representation, human brain categorizes objects by associating an input unseen object to a set of instances. This motivates us to learn the intra-class association between an input query object and some exemplars. This is a challenging task as such association should be universal and is built from a few exemplars (only 2 - 4 exemplars in our experiments) rather than the entire dataset. In addition, objects from the same category may appear in different colors, scales, and viewpoints, which makes the task even more challenging.

In this paper, we propose a multi-exemplar convolutional neural network (CNN) [25] for detecting top-down saliency by establishing visual association between a query image and multiple object exemplars, as shown in Figure 1. The intra-class association is learned in a unified manner by jointly extracting features from the exemplars and query object in a two-stage scheme. These two stages correspond to association and background discrimination learning. The main contributions of our work are summarized as follows:

1. We design a two-stage deep model (Figure 2 left) to detect top-down saliency by associating multiple exemplars with the query object, and explore the performance of different network structures.
2. We delve into the relationship between exemplars and the learned associations. In particular, we explore how different numbers of exemplars as well as the exemplar quality affect saliency detection performance.
3. We explore the proposed deep model in different tasks, including same-class identification, object location predication, and top-down saliency detection (Figure 2 right). Experiments on the Pascal VOC 2012 dataset show that the proposed model outperforms different baselines and the state-of-the-art category-specific methods.
4. We investigate the generalization capability of the learned intra-class association by applying it to unseen objects. The proposed networks offer surprisingly good generalization.

To the best of our knowledge, our work is the first to design and explore a multi-exemplar deep model.

2. Related Work

In this section, we first discuss relevant top-down saliency detection methods. We then describe object localization methods, as they share a similar objective to top-down saliency detection.

Top-down saliency includes two main processes, dictionary learning for each category (i.e., learning category "knowledge") and saliency computation (i.e., knowledge-driven searching). An early work by Torralba *et al.* [33] propose to use contextual information in a Bayesian framework to detect top-down saliency. Gao *et al.* [12] propose to characterize discriminant features using the statistical differences of presense/absense of the target class. Judd *et al.* [21] and Borji [3] combine bottom-up and top-down models by introducing high-level information to detect saliency, as objects like human persons, faces, and cars typically attract human attention. Ehing *et al.* [10] incorporate target appearance, location and scene context to model saliency. Kanan *et al.* [22] use independent component analysis (ICA) to learn target appearance, and then a trained SVM to detect top-down saliency. In [38, 23], top-down saliency is modeled by jointly learning category-specific dictionaries and CRF parameters. Similar to top-down saliency, Oquab *et al.* [28, 29] propose to generate a confidence map for each category location using CNN. In [28], large-scale knowledge in ImageNet is transferred to locate objects, while in [29], a weakly-supervised CNN is used to predict object locations. However, adapting limited amount of training data to unlimited test data is always desirable. Existing methods requires category-specific learning, and are thus restricted from training a universal model.

Object Localization aims to produce bounding boxes on the target objects, and can be roughly categorized into two classes, generic object localization and accurate object localization. Generic object localization (or object proposal detection) [24, 34, 6, 1, 42, 18, 30] aims to cover all objects in the scene with fewer and better bounding boxes than sliding windows, and to reduce the computational overhead of object detection. However, they are too general to obtain high accuracy with few proposals. Accurate object localization is mainly to produce one bounding box for each target object category in the image. It is much challenging and typically requires additional information. Dai *et al.* [8] assume that a given detected bounding box is not accurate, and propose to re-localize the object by propagating the target appearance information. In [14, 31], annotations in the database are used to label target object class. Song *et al.* [32] combine a discriminative submodular cover problem with a smoothed latent SVM formulation to locate objects with minimal supervision. While these works are promising, they attempt to find the best bounding box with visually consistent appearance to the training data. On the contrary, the proposed method is able to locate objects using just a few exemplars, which may contain large appearance variations.

3. Intra-class Association Network

Given a few exemplar objects from the same category and a query image, our goal is to determine whether the query image belongs to the same category of the exemplars. This process should not rely on any semantic information, and it should be as universal as possible and able to apply to unseen objects. Our approach is to train the proposed multi-exemplar deep model in two stages. As objects from the same class shares similar properties and features, the first stage is to learn to extract intra-class features, which determines objects being in the same category. To remove background distraction, the second stage is to learn how to discriminate the background. The trained network can then be applied to top-down saliency detection, same-class identification, and object location prediction. The pipeline of the proposed method is shown in Figure 2.

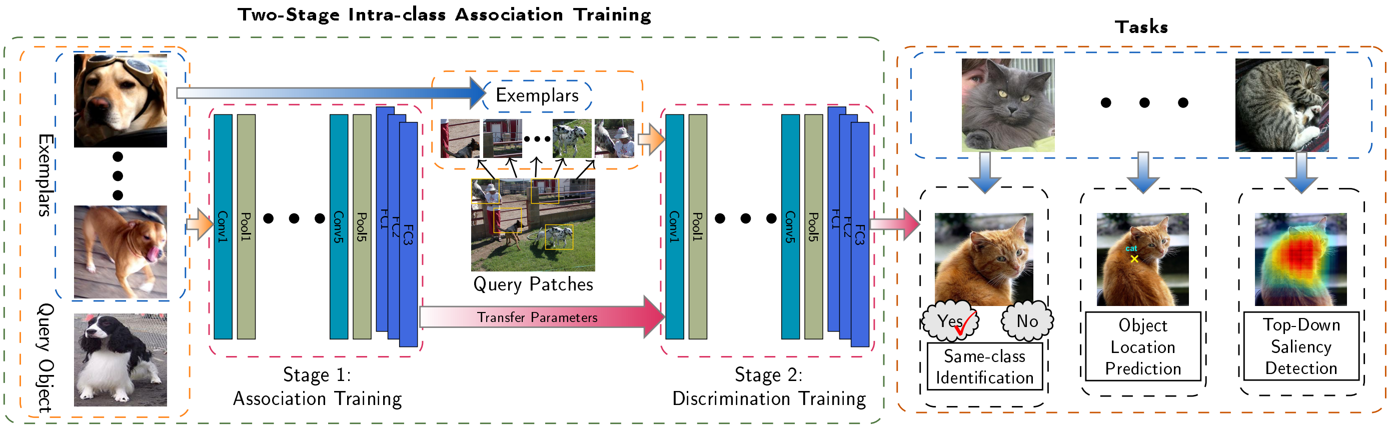


Figure 2: The pipeline of the proposed method. We treat the exemplars and query images as a unified multi-channel input and learn the intra-class association in a two-stage process. The first stage is fed with object patches to learn the intra-class association. The second stage is fed with sliding patches to learn background discrimination. The trained network is powerful and can be applied to different tasks.

3.1. Initialization and Network Architecture

As demonstrated in [13, 41], a network pre-trained on a large scale dataset shows great generalization for different types of tasks and input data. Starting with a pre-trained network can significant improve the task performance, even if the task is very different from the one in pre-training. Similarly, we initialize our network using the fast VGG network (VGG-f) [5] pre-trained on ImageNet [9], which comprises 5 convolutional layers and 3 fully connected layers. (Other networks can also be used here.) As the number of inputs is different from the pre-training task, we need to alter the network architecture to adapt to our problem. There are two possible models for our purpose.

Siamese network is a multi-tower architecture [4]. Each tower has the same layer structure and shares the same set of parameters. Fully connected layers are placed on top of these towers. Each input image is assigned to one tower to extract features of the image. The output feature maps are concatenated and passed to the fully connected layers. To adapt our problem to this network, we initialize each tower as a VGG-f network without the fully connected layers, which are added back to cover all the towers after the initialization. The sizes of the fully connected layers are expanded accordingly to measure the similarity among the images. The number of outputs for the last fully connected layer is set to 2, as we are solving a binary classification problem.

This type of network is mainly used to compare the similarity of two images [39, 15], and it shares a similar idea as the traditional descriptor-based approach. Each tower takes one image as input. This process can be viewed as descriptor computation. The fully connected layers at the top measure all these feature maps and thus can be viewed as a similarity function. However, extracting features from individual inputs is not a proper way to learn object association, especially with multiple exemplars. This is because the network only learns to describe image locally, i.e., the learned features are independent of the other inputs. Based on these mutually independent features, learning a similarity function is not enough to identify the large intra-class appearance differences.

Unified network learns to describe all input images jointly. In contrast to the Siamese network, all input images here are treated as a single multi-channel input. For example, four exemplars and one query image are combined to form a 5×3 channels image volume, with 3 being the 3 color channels. Due to the change in input dimension, the first convolutional layer needs to expand its channels accordingly. In our implementation, we have tried setting the parameters of the first convolutional layer in two ways: with random values and making multiple copies (equal to the number of exemplars) of the parameter values from the pre-trained model. As expected, the latter approach performs much better than randomly initialized parameters. Other layers of the VGG-f network remain the same, except that the number of outputs for the last fully connected layer is set to 2 for binary classification.

Compared with the Siamese network, our unified network has a greater flexibility to learn features of multiple objects, as all of them are considered jointly and the extracted features are intra-class features. In addition, the unified network is faster in both training and testing, especially with a large number of exemplars, as the computational complexity of the Siamese network is proportional to the number of inputs. The performance of the Siamese network and our unified network will be examined in Section 4.

Once the network architecture is determined and the parameters are initialized, we may then train the network for object association.

3.2. Stage 1: Object Association Training

As the initial network is pre-trained for image classification with centered objects, we further train the network on the Pascal VOC 2012 dataset [11] with object-level supervision. In order to learn the association among objects, the training process should be supervised in an object-to-object manner. As a result, we crop all the objects in the training data into patches for training. These patches are resized to the same size as the input size of the first convolutional layer (224 × 224 for the VGG-f network). The object-based training data is augmented by introducing jitter for robust association learning and combating overfitting. All training patches are randomly flipped, shifted, rotated, and enlarged/shrinked by a small amount.

There are two types of inputs for our network, exemplars and query images. Different construction methods for the input image volume lead to intrinsically different supervision approaches. If we train the network by randomly sampling objects from the training set, it is equivalent to identifying if a set of images belong to the same category, which is not our purpose. The proposed model is exemplar-driven, which means that all given exemplars should come from the same category. This construction method reduces the learning ambiguity, allowing the network to focus on delving into the relationship between the known-positives and unknown query (i.e., multiple-to-one connection), and the exemplars are able to provide guidance for both training and testing. For each training query object, its label is randomly defined by selecting exemplars from the same class of the query object or from other classes. (At least 30% of the data is positive to balance data distribution.) Note that the selected exemplars belong to the same category for both positive and negative training samples.

The network is trained using stochastic gradient descent (SGD) with a batch size of 50. Cross-entropy error is minimized in our network. The learning rate for this stage is set to 0.001.

3.3. Stage 2: Background Discrimination Training

In stage 1, we learn the association between the exemplars and query object. However, in order to effectively detect top-down saliency, we also needs to differentiate cluttered background to prevent background distraction. We fine-tune our network using the sliding window strategy to obtain diverse patches for training. The patches are extracted following the sliding window setting in [28]. All the patches are square with width $s = min(w,h)/\lambda$, where w and h are the image width and height, and $\lambda \in \{1, 1.3, 1.6, 2, 2.4, 2.8, 3.2, 3.6, 4\}$. They are sampled on a regular grid with at least 50% overlap with its neighbors. In total, there are around 500 patches for a 500×400 image. Similar to stage 1, these patches are resized to 224×224 before feeding to the network. Compared to the training with object proposals [13], the bounding boxes obtained by sliding windows are more diverse and thus can train the network with less patches.

For each target category in the image, we randomly select exemplars from the same category, and the positive query patches are defined loosely with certain extent of background. The label of patch P is positive if all the following conditions are satisfied: (i) the intersection ratio between P and the ground truth bounding box G_c of class c is larger than $0.2|P|$; (ii) large portion of object G_c is included in P such that $|P \cap G_c| \geq 0.6|G_c|$; (iii) P includes only one object. The training setting is the same as in stage 1 except for the learning rate, which is set to a smaller value of 0.0001 for fine-tuning the parameters of stage 1.

The training process in stage 2 has a different objective to the one in stage 1. Stage 1 trains on objects only, and its goal is to learn to identify what makes objects being in the same category. Stage 2 fine-tunes the network with arbitrary data using a smaller learning rate, and its goal is to learn to differentiate the background as well as the negative patches that partially overlap with the object. While stage 2 may be more important to top-down saliency performance (reducing background errors), stage 1 is the key for learning association and a universal model. As such, we intentionally bias the training process to stage 1. We will show results of different stages in Section 4.

3.4. Prediction

Once the network is properly trained, we are ready to apply it to different tasks. For all these tasks, we only use the final network trained by the two-stage approach.

Same-class identification: The most intuitive task of the propose network is to classify if a query object belongs to the same class of the exemplars. This task is the most straightforward way to show the learned association. It is also fundamental to the other tasks.

Top-down saliency detection: To detect saliency, we apply same-class identification to the entire image. Like the training process in stage 2, we first extract a set of patches from the image, but unlike it, the patch sampling strategy is not limited to sliding windows. In practice, we have found that the learned network has great generalization capability and can process patches with arbitrary sizes and scales. We have tested two bounding box sampling strategies: sliding windows in Section 3.3 and EdgeBoxes [42]. The first one produces diverse bounding boxes, while the latter locates objects tightly. However, object proposal detection requires a larger number of windows (around 1000) in order to cover most of the objects. As such, we use sliding windows in all our experiments, as a trade-off between accuracy and

#	Method	plane	bike	bird	boat	btl	bus	car	cat	chair	cow	table	dog	horse	moto	pers	plant	sheep	sofa	train	tv	mStd	mAP
1	Siamese	83.7	80.4	70.4	77.8	75.1	85.6	83.6	77.8	72.5	77.7	73.1	75.0	75.8	82.0	82.1	83.8	80.6	70.2	81.8	85.1	1.6	78.7
2	Ours - stage 1	88.3	85.9	75.0	82.3	79.6	90.9	87.9	81.5	77.2	82.3	78.1	79.0	80.1	85.9	86.7	88.7	84.6	75.0	86.6	89.4	1.38	83.3
3	Ours - stage 2	81.6	77.2	67.5	73.3	70.1	83.4	82.5	72.2	67.5	73.5	70.5	71.5	71.3	77.1	76.9	81.5	77.4	62.7	77.8	83.7	1.9	75.0
4	Ours (1 expl)	84.8	83.4	76.0	77.0	76.8	83.9	80.0	82.8	72.0	78.2	76.8	79.3	75.0	80.2	80.5	80.3	80.6	70.4	82.0	82.4	1.46	79.1
5	Ours (2 expls)	80.6	88.1	77.0	75.7	76.9	86.3	86.0	83.6	76.8	83.7	79.3	80.6	80.6	86.3	86.6	85.0	84.5	75.3	84.9	89.3	1.38	82.4
6	Ours (3 expls)	86.3	**89.5**	76.8	**84.7**	79.9	92.3	87.2	86.6	**82.0**	83.1	**83.6**	**83.8**	**82.8**	84.5	89.1	**91.1**	85.5	73.3	87.6	90.9	1.36	85.0
7	Ours (4 expls)	**90.0**	87.7	**77.1**	84.3	81.6	**92.6**	89.9	83.6	79.3	**84.1**	80.3	81.1	82.4	**88.0**	88.5	90.4	**86.8**	77.0	**88.1**	**91.4**	**1.31**	**85.2**
8	Category-Specific	87.6	80.2	75.3	81.4	**82.1**	87.6	**90.7**	**87.8**	81.9	72.6	73.7	82.1	75.2	81.9	**89.2**	80.8	78.6	**78.8**	84.8	80.2	–	81.6

Table 1: Same-class identification results on the Pascal VOC 2012 validation set. The proposed method is evaluated with different network architectures, stages, and numbers of exemplars. The category-specific network is trained using the same training strategy and architecture as ours. The proposed method consistently outperforms the category-specific model.

efficiency. Each patch is then fed to the network, and the saliency value of pixel Q for class c is computed as:

$$sal(Q,c) = \sum_{i=1}^{M} y(P_i,1) \cdot \delta(P_i,Q,y), \tag{1}$$

where M is the total number of patches. $y(P_i,1)$ is the confidence score of P_i belonging to the same class as the exemplars. δ is an indicator function:

$$\delta(P_i,Q,y) = \begin{cases} 1, & \text{if } Q \in P_i \text{ and } y(P_i,1) > y(P_i,0) \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

The final saliency map is normalized to $[0,1]$.

Object location prediction: The proposed method is able to output a precise location of a specific object. Based on the top-down saliency map of a target category, the object location can be easily obtained by applying a global max-pooling operation on the entire map. An example of the saliency map and its corresponding predicted location are shown in Figure 2 right. Note that this approach is able to obtain one location per category, but it is sufficient to evaluate the accuracy of the proposed method.

4. Experiments

In this section, we evaluate the performance of the proposed method, explore the use of different numbers of exemplars, and investigate the generalization capability of it to unseen objects. The proposed method is implemented using MatConvnet [35] and tested on a PC with an i7 3.4GHz CPU, a Nvidia 980M GPU, and 32GB RAM. It takes 2-3s to process a 500 × 400 image. In our experiments, all the compared networks are trained with the same amount of data (i.e., the same number of epoches). Stage 1 takes 4 - 5 days for the training to converge, and stage 2 takes 3 - 4 days.

4.1. Same-class Identification

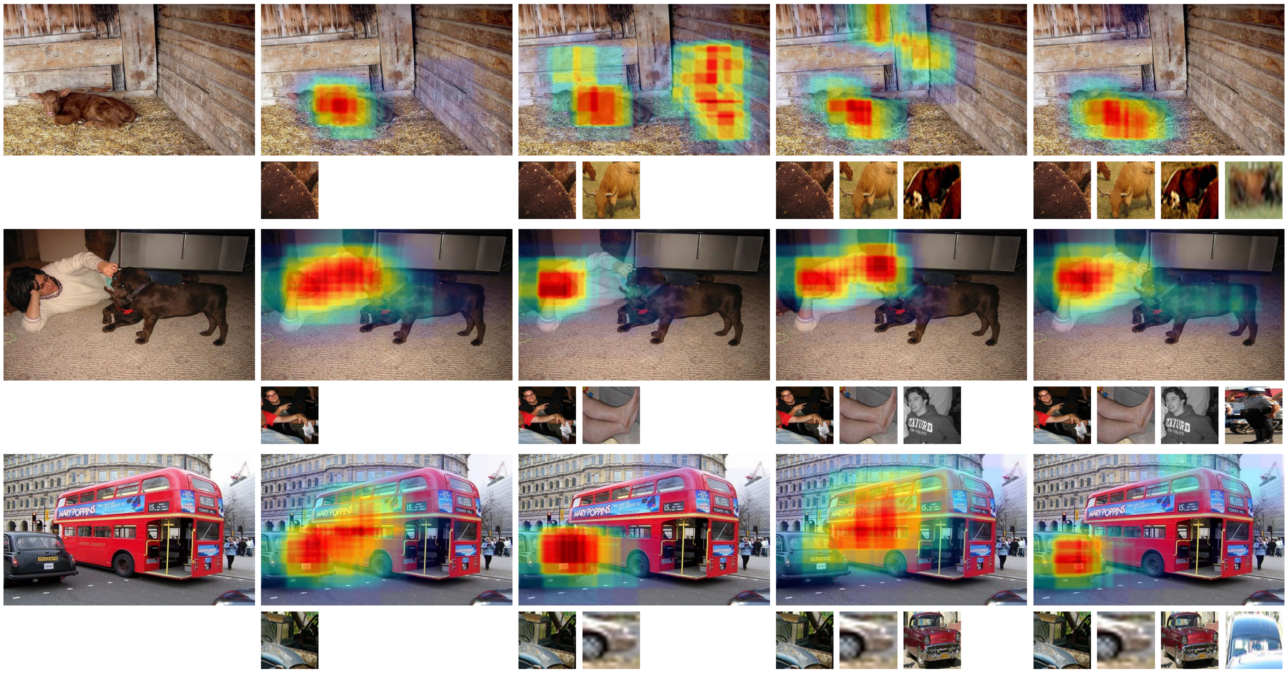
We first evaluate the learned association on same-class identification. The network is trained on the Pascal VOC 2012 training set and tested on the validation set. The input exemplars and query images are cropped according to the ground truth bounding boxes. In total, there are 13,609 object images in the training set, and 13,841 in the validation set. For both training and validation sets, there are at least 280 object images per category. During testing, we use the objects from the training set as exemplars and those from the validation set as query objects. As the use of different numbers of exemplars may affect identification performance, we randomly generate 5 exemplar sets for each number of input exemplars (1 - 4 are tested in our evaluations). All the other evaluations use the same 5 sets of inputs. Table 1 shows the average performance for the 5 sets of exemplars on each class and the average per-class standard deviation for evaluating the influence of different numbers of exemplars.

Compared with the Siamese network. We first compare the proposed network with the Siamese network, which has a multi-tower structure. The Siamese network is trained using 4 exemplars and follows the same training strategy as ours. As shown in Table 1, the Siamese network (row 1) performs much worse than ours (row 7) using 4 exemplars. This is because the Siamese network extracts features from the input exemplars individually, while the proposed network jointly considers all the inputs and thus has higher flexibility to learn the association.

Results of different stages. We then evaluate if training the network with only one of the two stages can achieve good results. Two networks are trained with the same amount of data individually with 4 exemplar inputs. The learning rate is set to 0.001 for both stages. To make the comparison fairer, we randomly skipped 20% background training samples while training stage 2 to make the positive and negative samples balance. Due to the intrinsically different objectives and training processes of the two stages, the performances are different on identification. As shown in rows 2 and 3 of Table 1, stage 1 has better identification performance than stage 2. This is because stage 1 focuses on object-to-object association, while stage 2 biases to object-to-background learning. The trained network of stage 2 is difficult to classify objects across classes.

#	Method	plane	bike	bird	boat	btl	bus	car	cat	chair	cow	table	dog	horse	moto	pers	plant	sheep	sofa	train	tv	mStd	Mean
1	Yang [38]	14.7	28.1	9.8	6.1	2.2	24.1	30.2	17.3	6.2	7.6	10.3	11.5	12.5	24.1	36.7	2.2	20.4	12.3	26.1	10.2	–	15.6
2	Kocak [23]	46.5	**45.0**	33.1	**60.2**	25.8	48.4	31.4	64.4	19.8	32.2	**44.7**	30.1	41.8	**72.1**	33.0	40.5	38.6	12.2	64.6	23.6	–	40.4
3	Oquab [29]	48.9	42.9	37.9	47.1	31.4	68.4	39.9	66.2	**27.2**	54.0	38.3	48.5	56.5	70.1	43.2	42.6	52.2	34.8	68.1	43.4	–	48.1
4	Ours (1 expl)	38.5	32.4	48.7	31.4	27.5	80.4	52.7	68.2	25.9	62.6	30.0	68.6	62.5	65.1	45.5	37.6	61.2	39.5	66.8	52.4	1.85	50.5
5	Ours (2 expls)	48.0	32.6	**51.4**	34.2	32.5	78.5	54.1	69.0	25.1	62.7	36.6	69.0	61.3	64.0	46.9	41.1	57.1	**42.9**	70.0	**58.4**	1.63	52.0
6	Ours (3 expls)	52.7	36.9	46.4	42.3	43.5	81.8	55.6	69.0	27.1	**69.1**	38.1	67.5	61.7	64.0	**58.5**	43.2	59.4	40.8	71.4	57.0	1.51	54.3
7	Ours (4 expls)	**55.9**	37.9	45.6	43.8	**47.3**	**83.6**	**57.8**	**69.4**	22.7	68.5	37.1	**72.8**	**63.7**	69.0	57.5	**43.9**	**66.6**	38.3	**75.1**	56.7	**1.41**	**56.2**

Table 2: Top-down saliency precision rates (%) at EER on the Pascal VOC 2012 validation set. All the compared methods (rows 1 - 3) are category-specific approaches.



(a) Inputs (b) Using 1 exemplar (c) Using 2 exemplars (d) Using 3 exemplars (e) Using 4 exemplars

Figure 3: Saliency maps generated by the proposed method using different numbers of exemplars. The target objects (top to bottom) are cow, person, and car.

Relationship between exemplars and query object. We further explore if more exemplars help the identification performance. As shown in rows 4 - 7 of Table 1, more exemplars indeed improve identification performance. There are two possible reasons. First, the chance of selecting good exemplars is higher with more inputs. Second, intra-class association is learned more robustly with more exemplars. Regarding the first conjecture, it is also related to whether exemplar quality affects identification performance. We report the average per-class standard deviation in the second last column. We can see that the variances of training with more exemplars are smaller than those with fewer inputs. This indicates that the association trained with fewer exemplars relies more on input quality and vice versa. However, in general, the small standard deviations show that the exemplar quality will not significantly influence the identification performance. We will further demonstrate this observation in Section 4.2 qualitatively.

Compared with a category-specific network. In this experiment, a category-specific VGG-f network pre-trained on ImageNet and fine-tuned on Pascal VOC 2012 (using the same learning rate of 0.001) is used as a baseline. Its performance is shown in row 8 of Table 1. Surprisingly, this category-specific network performs only similarly to ours when using 2 exemplars. It is even worse than ours when using 3 or 4 exemplars. We attribute this to our restricted multi-exemplar model. Exemplars are selected from the same class, which reduces the learning ambiguity. This also suggests that once the network has learnt the association, exemplars provide powerful guidance.

4.2. Top-down Saliency Detection

We then examine the performance of top-down saliency detection and delve into the learnt association. We compare our method with two latest top-down saliency detection methods [38, 23], and one latest object localization method [29]. All these methods compared are category-specific. We use the sliding window sampling strategy in Section 3.3 to extract patches for saliency detection. Similarly, we randomly generate 5 sets of exemplars per test image for comparison. We evaluate the top-down saliency maps using the segmentation annotation of the Pascal VOC 2012 validation set, which consists of 1,449 images. Unlike the evaluation setting in [38], we evaluate the saliency map in pixel-level rather than patch-level for higher accuracy. We first binarize the saliency map for every threshold in the range of [0, 255] to generate the precision-recall curves (P-R curves), and the performance of each category is summarized by the precision rates at equal error rates (EER, where precision is equal to recall). The performances of different methods are shown in Table 2. The two state-of-the-art top-down saliency detection methods (rows 1 - 2 in Table 2) encode object information using dictionary learning, but the large appearance differences among the objects of the same class are difficult to capture using their approach. The CNN-based approach (row 3 in Table 2) performs not as good as ours, due to the learning guidance provided by our two-stage training process.

Relationship between exemplars and query object. As shown in Table 2, the performance of our method increases with the number of exemplars, and the per-class variations are also small. This is similar to the last experiment. Here, we mainly explore how exemplar quality influences detection performance. Figure 3 shows some top-down saliency detection examples. The saliency maps are produced using the same sets but diffennt numbers of examplars. It demonstrates how each additional exemplar may affect the result. We can also see that a bad exemplar harms the detection. In the first example, the second exemplar is a bad one and it distracts the detection to the wooden wall (due to a similar color). In the second example, the second exemplar of the human feet causes the second saliency map to focus on the human face. In the third example with 3 exemplars, the proposed method wrongly renders the bus as salient, since it shares a similar appearance to the added exemplar. All these cases suggest that color similarity is one of the main influential factors. However, bad exemplar only produces false positive and will not significantly affect the true positive results. In addition, the association learnt using more exemplars is more robust to outliers. As we can see from all three examples with 4 exemplars, the 4-exemplar network is more capable of tolerating bad exemplars and can properly predict salient regions.

4.3. Object Location Prediction

We further evaluate the accuracy of the predicted object locations. As mentioned above, a simple max-pooling operator applied on the saliency map is able to predict an object location for a target category. Here, we compare to the three methods used in Section 4.2. In addition, we add the stage-of-the-art object detector RCNN [13] as a baseline, which outputs a bunch of bounding boxes along with the confidence values in order to cover all the objects in the image. We select the bounding box with the highest confidence value for each target category, and pick the center pixel as the object location. The localization performances of all these methods are examined by simply labeling the predicted location as correct if it falls into the ground truth bounding box of the target category, and negative otherwise. Unlike [29], which sets a 18-pixel tolerance to the predicted location, we restrict the correct predicted location to be within the ground truth bounding box for a more accurate evaluation. The confidence values of the predicted locations are used to generate the P-R curves, and the final performance of each category is summarized by Average Precision (AP). We note that his metric can be challenging for cluttered scenarios. The location prediction experiment is conducted on the Pascal VOC 2012 validation set.

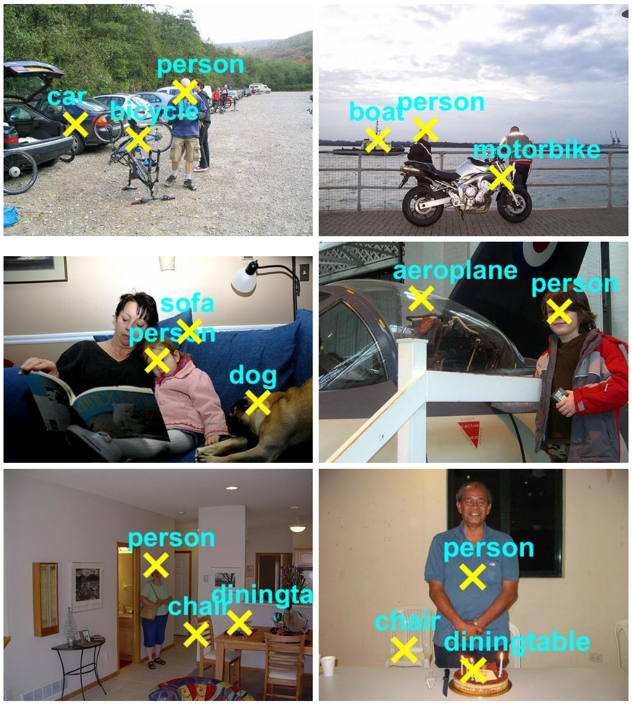


Figure 4: Examples of object location prediction.

The location prediction results are shown in Table 3. Our method with 2, 3 or 4 exemplars outperforms all three existing methods and the state-of-the-art object detector overall. Our method with 4 exemplars achieves the best performance in most of the classes. Note that the bounding box sampling strategy affects prediction performance. Since most of the top-down saliency detection methods (in-

#	Method	plane	bike	bird	boat	btl	bus	car	cat	chair	cow	table	dog	horse	moto	pers	plant	sheep	sofa	train	tv	mStd	mAP
1	Yang [38]	57.2	49.6	47.6	45.0	10.3	58.5	41.0	54.6	14.5	40.1	21.4	49.7	57.6	50.1	58.3	22.7	54.4	17.2	51.6	36.3	–	41.9
2	Kocak [23]	70.7	55.4	60.9	53.4	27.3	68.4	52.3	75.4	31.8	60.1	36.1	64.9	70.5	69.6	71.8	33.3	68.2	29.2	70.8	52.5	–	56.1
3	Oquab [29]	83.2	68.2	71.9	**69.2**	33.7	79.0	57.8	73.8	42.0	75.8	50.1	72.7	75.7	75.7	77.6	37.1	76.7	44.2	81.1	60.6	–	65.3
4	RCNN [13]	86.5	72.1	**74.2**	66.7	**43.1**	78.3	**68.8**	80.8	44.9	62.3	**51.1**	74.4	73.6	83.0	83.0	49.2	78.4	40.6	74.1	**69.2**	–	67.7
5	Ours (1 expl)	77.4	81.9	67.6	40.6	26.4	85.0	52.2	85.4	38.1	87.3	33.8	80.5	84.0	87.5	79.6	51.4	85.5	49.6	79.7	53.8	2.03	66.4
6	Ours (2 expls)	84.1	80.3	69.8	40.6	26.8	87.5	55.1	92.7	38.7	92.7	37.7	84.9	87.8	90.9	86.7	51.9	89.7	55.2	80.0	54.5	1.78	69.4
7	Ours (3 expls)	**87.1**	85.5	71.3	43.6	30.8	87.3	58.0	93.9	**45.3**	93.6	40.5	84.3	88.7	91.8	85.8	**57.8**	90.9	55.7	83.9	59.2	1.59	71.7
8	Ours (4 expls)	86.8	**87.2**	72.7	46.8	31.7	**91.0**	58.6	**95.2**	44.5	**94.8**	41.5	**87.0**	**91.4**	**94.3**	**89.2**	57.7	**93.5**	**59.2**	**84.7**	60.5	**1.53**	**73.4**

Table 3: Object location prediction results on the Pascal VOC 2012 validation set.

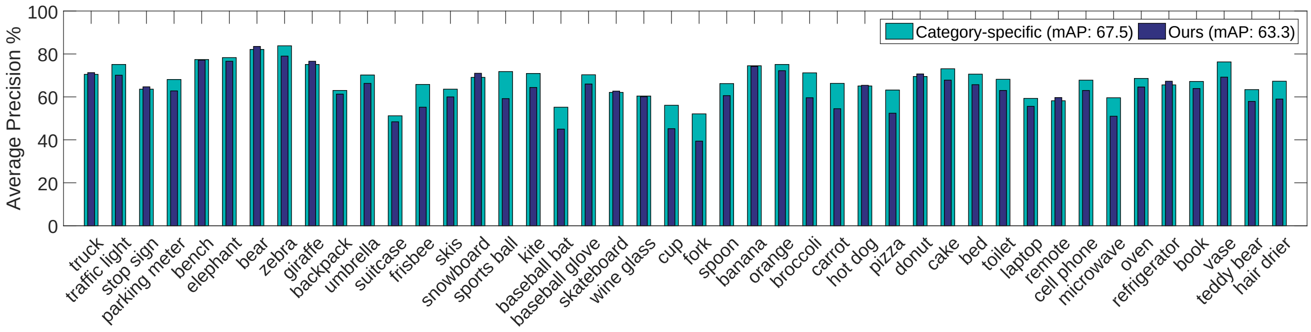


Figure 5: Same-class identification results for unseen categories on the MS COCO validation set. The category-specific network is trained on the entire training set. The proposed method is trained on a subset of categories and tested on the unseen ones.

cluding Yang *et al.* [38], Oquab *et al.* [29], and ours) detect objects in a sliding window fashion, they may not be able to precisely locate small scale objects, e.g., bottles. On the contrary, RCNN uses a large number of object proposals and can thus capture objects in different sizes and scales. However, the large number of object proposals increases the error rates due to false positive. As a result, its overall performance is not as good as ours. Some examples of object location predictions from our method are shown in Figure 4.

4.4. Unseen Category Evaluation

The high accuracy of the proposed method on the Pascal VOC 2012 dataset does not guarantee that the learnt association can be generalized to unseen categories. To evaluate the generalization capability of the proposed method, we apply it to the much larger MS COCO dataset [26], which consists of 80 classes. Due to the significant increase in the number of categories from Pascal VOC 2012 (which has 20 categories), we fine-tune the proposed network by randomly selected additional 16 categories for training, leaving us 44 unseen categories for evaluation.

We test the unseen categories on same-class identification using 4 exemplars. Again, a category-specific VGG-f network trained on the MS COCO training set is used as the baseline. Figure 5 shows results on the unseen categories. We can see that the proposed method performs slightly worse than, but still comparable to, the category-specific network on unseen objects. This suggests that the proposed network has good generalization capability to unseen classes.

5. Conclusion

In this paper, we have proposed a novel locate-by-exemplar top-down saliency detection framework. With this approach, object association is captured by a multi-exemplar network and learnt in a two-stage training process. We have shown that the network learnt with more exemplars achieves more robust association quantitatively and qualitatively. We have also shown that the proposed network outperforms the state-of-the-art category-specific methods in different tasks. Even for unseen objects, the proposed network can infer the association from learnt knowledge.

The proposed *same-class identification* is a fundamental task for a lot of vision applications. As a future work, we aim to extend it to *same-object identification*, which would be useful for visual object tracking to identify objects under different circumstances.

Acknowledgements. This work was partially supported by a GRF grant from the RGC of Hong Kong (Project Number: CityU 115112) and the Early Career Scheme through the Research Grants Council, University Grants Committee, Hong Kong (Project Number: CityU 21201914). We thank Qingxiong Yang for his funding support for Shengfeng He's Post-Doctoral fellowship.

References

[1] P. Arbelaez, J. Pont-Tuset, J. Barron, F. Marqués, and J. Malik. Multiscale combinatorial grouping. In *CVPR*, 2014.

[2] M. Bar. The proactive brain: using analogies and associations to generate predictions. *Trends in Cognitive Sciences*, 11(7):280 – 289, 2007.

[3] A. Borji. Boosting bottom-up and top-down visual features for saliency estimation. In *CVPR*, pages 438–445, June 2012.

[4] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah. Signature verification using a "siamese" time delay neural network. In *NIPS*, pages 737–744. 1994.

[5] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. In *BMVC*, 2014.

[6] M. Cheng, Z. Zhang, W. Lin, and P. Torr. BING: Binarized normed gradients for objectness estimation at 300fps. In *CVPR*, 2014.

[7] H. Cholakkal, D. Rajan, and J. Johnson. Top-down saliency with locality-constrained contextual sparse coding. In *B-MVC*, 2015.

[8] Q. Dai and D. Hoiem. Learning to localize detected objects. In *CVPR*, pages 3322–3329, June 2012.

[9] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009.

[10] K. Ehinger, B. Hidalgo-Sotelo, A. Torralba, and A. Oliva. Modeling search for people in 900 scenes: A combined source model of eye guidance. *Visual cognition*, 17(6-7):945–978, 2009.

[11] M. Everingham, S. Eslami, L. Van Gool, C. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge: A retrospective. *IJCV*, 111(1):98–136, Jan. 2015.

[12] D. Gao, S. Han, and N. Vasconcelos. Discriminant saliency, the detection of suspicious coincidences, and applications to visual recognition. *IEEE TPAMI*, 31(6):989–1005, June 2009.

[13] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014.

[14] M. Guillaumin and V. Ferrari. Large-scale knowledge transfer for object localization in imagenet. In *CVPR*, pages 3202–3209, 2012.

[15] X. Han, T. Leung, Y. Jia, R. Sukthankar, and A. Berg. Matchnet: Unifying feature and metric learning for patch-based matching. In *CVPR*, 2015.

[16] S. He and R. Lau. Saliency detection with flash and no-flash image pairs. In *ECCV*, pages 110–124, 2014.

[17] S. He, R. Lau, W. Liu, Z. Huang, and Q. Yang. SuperCNN: A superpixelwise convolutional neural network for salient object detection. *IJCV*, 115(3):330–344, 2015.

[18] S. He and R. W. Lau. Oriented object proposals. In *ICCV*, pages 280–288, 2015.

[19] X. Hou and L. Zhang. Saliency detection: A spectral residual approach. In *CVPR*, pages 1–8, 2007.

[20] L. Itti and C. Koch. Computational modelling of visual attention. *Nature Reviews Neuroscience*, 2(3):194–203, Mar 2001.

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

[22] C. Kanan, M. Tong, L. Zhang, and G. Cottrell. Sun: Top-down saliency using natural statistics. *Visual Cognition*, 17(8):979–1003, 2009.

[23] A. Kocak, K. Cizmeciler, A. Erdem, and E. Erkut. Top down saliency estimation via superpixel-based discriminative dictionaries. In *BMVC*, 2014.

[24] C. Lampert, M. Blaschko, and T. Hofmann. Beyond sliding windows: Object localization by efficient subwindow search. In *CVPR*, pages 1–8, 2008.

[25] Y. LeCun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard, and L. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4):541–551, Dec 1989.

[26] T. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. Zitnick. Microsoft COCO: Common objects in context. In *ECCV*. 2014.

[27] R. Nosofsky. Attention, similarity and the identification-categorization relationship. *Journal of Experimental Psychology: General*, 115:39–57, 1986.

[28] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *CVPR*, 2014.

[29] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Is object localization for free? - weakly-supervised learning with convolutional neural networks. In *CVPR*, 2015.

[30] P. O. Pinheiro, R. Collobert, and P. Dollar. Learning to segment object candidates. In *NIPS*, pages 1990–1998. 2015.

[31] J. Serrano and D. Larlus. Predicting an object location using a global image representation. In *ICCV*, pages 1729–1736, 2013.

[32] H. Song, R. Girshick, S. Jegelka, J. Mairal, Z. Harchaoui, and T. Darrell. On learning to localize objects with minimal supervision. In *ICML*, June 2014.

[33] A. Torralba, A. Oliva, M. Castelhano, and J. Henderson. Contextual guidance of eye movements and attention in real-world scenes: the role of global features in object search. *Psychological Review*, 113(4):766–786, Oct. 2006.

[34] J. Uijlings, K. van de Sande, T. Gevers, and A. Smeulders. Selective search for object recognition. *IJCV*, 2013.

[35] A. Vedaldi and K. Lenc. MatConvNet – convolutional neural networks for MATLAB. In *ACM Multimedia*, 2015.

[36] E. Vig, M. Dorr, and D. Cox. Large-scale optimization of hierarchical features for saliency prediction in natural images. In *CVPR*, pages 2798–2805, 2014.

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

[38] J. Yang and M. Yang. Top-down visual saliency via joint crf and dictionary learning. In *CVPR*, pages 2296–2303, 2012.

[39] J. Zbontar and Y. LeCun. Computing the stereo matching cost with a convolutional neural network. In *CVPR*, pages 1592–1599, 2015.

[40] J. Zhang, S. Sclaroff, Z. Lin, X. Shen, B. Price, and R. Měch. Minimum barrier salient object detection at 80 fps. In *ICCV*, 2015.

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

[42] C. Zitnick and P. Dollár. Edge boxes: Locating object proposals from edges. In *ECCV*. 2014.