Deeply Activated Salient Region for Instance Search

HUI-CHU XIAO, WAN-LEI ZHAO, JIE LIN, and YI-GENG HONG, Xiamen University
CHONG-WAH NGO, Singapore Management University

The performance of instance search relies heavily on the ability to locate and describe a wide variety of object instances in a video/image collection. Due to the lack of a proper mechanism for locating instances and deriving feature representation, instance search is generally only effective when the instances are from known object categories. In this article, a simple but effective instance-level feature representation approach is presented. Different from the existing approaches, the issues of class-agnostic instance localization and distinctive feature representation are considered. The former is achieved by detecting salient instance regions from an image by a layer-wise back-propagation process. The back-propagation starts from the last convolution layer of a pre-trained CNNs that is originally used for classification. The back-propagation proceeds layer by layer until it reaches the input layer. This allows the salient instance regions in the input image from both known and unknown categories to be activated. Each activated salient region covers the full or, more usually, a major range of an instance. The distinctive feature representation is produced by average-pooling on the feature map of a certain layer with the detected instance region. Experiments show that this kind of feature representation demonstrates considerably better performance than most of the existing approaches.

CCS Concepts: • **Information systems → Image search;**

Additional Key Words and Phrases: Instance search, back-propagation, response peak, instance-level

ACM Reference format:
Hui-Chu Xiao, Wan-Lei Zhao, Jie Lin, Yi-Geng Hong, and Chong-Wah Ngo. 2022. Deeply Activated Salient Region for Instance Search. *ACM Trans. Multimedia Comput. Commun. Appl.* 18, 3s, Article 147 (October 2022), 19 pages.
https://doi.org/10.1145/3510004

1 INTRODUCTION

Different from image search, instance search hunts for images with the same object instances as a query image. The query instance is usually specified by a bounding box within an image or a video frame. To provide evidence of a search result, the location where a visual instance resides in a retrieved image should be presented for inspection. Instance search is widely used in various multimedia applications. In video editing, instance search serves as a function to return all spatiotemporal locations of a query object instance, such as a character, in a full-length video. In an

This work was supported by National Natural Science Foundation of China under grants 61572408 and 61972326, and the grants of Xiamen University 20720180074.

Authors' addresses: H.-C. Xiao, W.-L. Zhao (corresponding author), J. Lin, and Y.-G. Hong, Haiyun Campus, Xiamen University, Xiamen, Fujian, China, 361005; emails: hcxiao@stu.xmu.edu.cn, wlzhao@xmu.edu.cn, linjie037@stu.xmu.edu.cn, yghong@stu.xmu.edu.cn; C.-W. Ngo, Singapore Management University, Singapore, Singapore; email: cwngo@smu.edu.sg.

1551-6857/2022/10-ART147 $15.00
https://doi.org/10.1145/3510004

online survey, instance search is deployed to estimate the popularity of a brand (e.g., Coca-Cola) by counting its appearance frequency over a large pool of Internet images. In online shopping, instance search enables fine-grained retrieval of product instances specific to a brand or one style that a customer requests.

In instance search, the relevancy is grounded on the existence of an instance rather than the visual similarity of the whole image. Therefore, the conventional content-based image retrieval approaches that capture the global visual distribution of an image fall short of this problem. Typically, these approaches collapse features of different image regions into an embedded vector for retrieval. The visual characteristics unique to an instance may have been smoothed out during the embedding. As a consequence, the global feature is no longer distinctive for the identification of individual instances, let alone the localization of instances as evidence of search result. The problem not only persists in handcrafted visual features such as GIST [4] but also in deep features globally extracted from various neural networks [2, 28].

When instance search was first addressed in the work of Awad et al. [1], the problem was coined as a sub-image retrieval task. Handcrafted features such as SIFT [22] and SURF [3] that are superior in local image matching were de facto descriptors at that time. Although encouraging results are reported [1], these approaches are known to be limited to match textureless image patches and instances that have undergone non-rigid motions. Although most of the descriptors are capable of generating thousands of local features from an image for matching, these features are extracted from regions rich of textures or corners. As a result, the object instances with textureless regions are under-represented. In addition to being invariant to 2D geometric transformations, local features can only tolerate certain degree of viewpoint and lighting changes. Particularly, the features are vulnerable to non-rigid deformations, which are widely observed in the real scenarios.

Recently, due to the great success of **convolutional neural networks (CNNs)** in learning high-level semantic features for image classification [18], object detection [6, 7, 30, 31], and instance segmentation [8, 19], CNNs have been introduced to instance search [43]. Using Fast R-CNN [6] as an example, the instance-wise vector representation is produced through RoI-pooling from the region of feature maps corresponding to a candidate object bounding box. The feature captures textureless regions and is relatively robust to object deformation when compared with global and local features. Despite satisfactory performance in instance search as reported by Zhan and Zhao [43], the main drawback of CNN-based solutions is their stringent demand for training data. In their work [43], for example, pixel-wise annotation of instance location is required. The annotation effort is expensive and labor intensive. Furthermore, the learning process makes the CNNs more sensitive to object instances of known categories by treating unseen categories as background class [8]. As a result, approach in the work of Zhan and Zhao [43] is only able to deal effectively with instances belonging to known object categories. This problem remains unaddressed if one switches to relying on CNN-based object detection framework [7, 30, 31] for instance-level feature extraction (e.g., [32]).

This article aims for class-agnostic feature representation and localization for instance search. Leveraging on the pre-trained CNNs for visual classification, a new approach is proposed to detect the potential instances in an image. Starting from the last convolution layer, our approach detects response peaks and back-propagates them layer by layer. Those peaks support classification decision and generally refer to the regions residing on a visual instance. Through back-propagation, the effective receptive size of a salient region that corresponds to an instance and supports classification is activated at each layer. When reaching the outermost layer (i.e., the input image), the locations where the salient regions of instances reside can be uncovered. As the back-propagation starts from the last convolution layer rather than the prediction layer, the uncovered instances are class-agnostic and not restricted to the known categories. Figure 1 shows the examples of instance

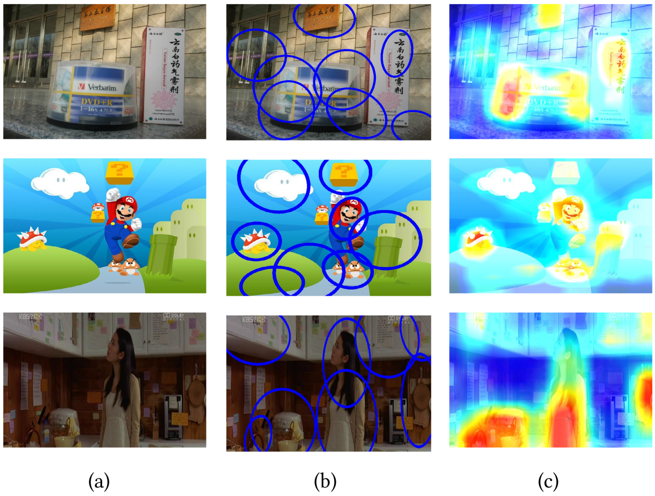

(a) (b) (c)

Fig. 1. Instance localization: original image (a), salient regions of instances being detected (b), and mean activation map by overlapping heat map on the original image (c). The map is color coded with red indicating high response and blue indicating low response. Each localized instance region is regularized by an estimated ellipse.

regions being detected, which are correspondent to the response peaks highlighted by the heat maps. A descriptor is then proposed to extract the feature of a salient region by average-pooling over the feature maps corresponding to the region location.

The rest of the article is organized as follows. Section 2 reviews the state-of-the-art works in instance search and weakly supervised object localization. Our instance-level feature, namely **deeply activated salient region (DASR)**, is presented in Section 3. The effectiveness of the proposed new feature representation is studied on the instance search in Section 4. Section 5 concludes the article.

2 RELATED WORK

2.1 Instance Search

Instance search was addressed as a sub-image retrieval task before CNNs were introduced [1] to visual object detection. The main image features being employed for this task are handcrafted local descriptors such as SIFT and SURF. Through matching of local features, the instances relevant to a query are discovered in the candidate images for similarity search. Due to the high computational cost of direct point-to-point matching, encoding approaches such as BoVW [34] and VLAD [14] were introduced to speed up the similarity computation between images. This line of approaches suffers from several limitations. First, non-rigid objects cannot be effectively handled [43]. Local features are vulnerable to non-rigid deformations and heavy viewpoint changes. Second, there is no guarantee that a feature being extracted will be unique to a particular instance. Instead, the features are often polluted by background or nearby objects of an instance. In most of the descriptors, the features are mostly extracted from local image patches located along the boundaries or corners of an object. When object instances are clustered in proximity, the image patch from which a feature is derived can occupy the partial regions of multiple instances. The problem also exists in local features extracted from deep neural networks [25, 27]. Third, matching hundreds or even thousands of local features across two images is computationally prohibitive. Although matching can be considerably sped up by BoVW or VLAD representation, the search quality is also inevitably degraded due to the vector quantization error.

Due to the satisfactory performance in image classification, pre-trained CNNs on classification tasks have been introduced to instance search. With the feature maps obtained by pre-trained models, **regional maximum activation of convolutions (R-MAC)** [37] aggregates features from several local regions into a global feature. Although encouraging results are obtained on image retrieval tasks, global features are infeasible for instance search. Weight aggregation strategies are employed by CroW [16], the **class activation map (CAM)** [15], BLCF-SalGAN [24], and Regional Attention [17] to address this problem. Region-level feature weighting allows the matching between global features to reflect the similarity between embedded instance features. The key idea is to assign weights to different channels or different regions in the feature map during the feature pooling. The channels or regions that contribute more to the classification decision are assigned with higher weights. Due to the weighting scheme, the instance that dominates in the image is highlighted in the embedded feature vector. Although the scheme enables effective instance search, localization of query instance in the candidate images is not possible.

Recently, several attempts have been devoted to instance-wise feature representation. The works rely on the fine-tuned CNNs that are designed for object detection or instance segmentation tasks. DeepVision [32] extracts region-level features from the bounding boxes generated by the object detection framework. Due to the high computation cost, the features are only leveraged to rerank images at the top of a rank list. FCIS+XD [43], instead, pixel-wisely extracts instance-level features from the instance segmentation map of a **fully convolutional network (FCN)**. The instances that the trained model could detect are restricted to a limited number of object categories. PCL*+SPN [20] extracts features from the object detection framework trained with image-level features. Despite leveraging on weakly supervised networks, similar retrieval performance as FCIS+XD is reported in the work of Lin et al. [20]. The pitfall of this approach is that the network requires extra training stages and its discriminativeness toward the unknown categories is undermined due to the extra training.

2.2 Weakly Supervised Object Localization

In a nutshell, robust instance-level feature representation relies on the ability to locate a wide variety of object instances. Compared to the fully supervised CNNs, weakly supervised networks that require only image-level labels for instance localization are more capable of dealing with a larger number of object categories. Specifically, object regions are automatically inferred rather than manually provided during network learning. The existing approaches include **proposal clustering learning (PCL)** [35], **multiple instance learning (MIL)** [23, 36, 38], and weakly supervised instance segmentation [46]. In PCL, a group of proposals are produced around the regions that contribute to the classification score of one category. The proposals are reduced to several cluster centers during the learning, each of which is expected to cover a latent object of that category. In MIL, an image is viewed as a bag of object proposals. One object proposal is potentially a visual instance. During the training, MIL iteratively selects the instance with the highest confidence score until all the latent instances are detected. Excitation Backprop [44] produces task-specific attention maps that are able to roughly localize the instances in an image. Similar to Zhang et al. [44], the **peak response map (PRM)** [46] leverages the instance-level visual cues inside CAMs [45]. The latent instance regions are highlighted by back-propagating iteratively the class-aware response peaks. The instance-aware cues are combined with class-aware cues and spatial continuity priors to produce instance segmentation masks. The best instance mask is selected after **non-maximum suppression (NMS)** for each latent instance. Different from Zhang et al. [44] and Zhou et al. [46], the activation map in the work of Wei et al. [40] is produced by a simple aggregation of all feature maps from $Pool_5$ of VGG-16 [33]. The regions whose activation values are higher than a threshold are detected as being part of a latent instance. The detected neighboring regions are combined as the main instance, from where the feature representation is derived. This resulting feature is applied in the fine-grained image retrieval. Since only one instance is detected from one image, the approach is only applicable for single object instance retrieval.

As witnessed in several recent works [42, 44, 46], salient regions of visual instances contribute significantly to the prediction of a CNN. This property has been originally leveraged to interpret the behavior of a CNN [42]. The salient regions, despite not enclosing the entire instances, have been explored in various ways. Examples include modeling the top-down attention of a CNN [44] and weakly supervised instance segmentation by integration of salient regions and other visual cues [46]. In all of these works, the salient regions are detected by back-propagating response peaks located in the classification-related layer. The back-propagation is essentially driven by the known categories that produce high classification confidence. As a result, the salient regions of unknown categories are overlooked throughout the process. The mechanism is not appropriate for instance search, which targets all instances beyond the known categories of a CNN.

Similar to the work of Zhang et al. [44], this work performs object localizations based on pre-trained CNNs that are used for image classification. The instance regions are localized by identifying the regions with high responses in the iteratively back-propagated activation map. However, our approach differs from the existing works in three major aspects. First, our approach does not intend to localize the full range of an instance. Instead, only the instance regions with high responses in the activation map are localized. A region usually corresponds to the major part of an instance being investigated by a network for classification. Second, the back-propagation starts from the last convolution layer of a network instead of the prediction layer. Finally, since no class-aware response is considered in our approach, no fine-tune training is involved. The advantages of performing instance localization based on a pre-trained classification network are twofold. First, the detection makes the localization remain sensitive to regions from the unknown instance categories.

3 DEEPLY ACTIVATED INSTANCE REGION DETECTION

In this work, a back-propagation process is leveraged to highlight class-agnostic latent instances, followed by a shape estimation module to further generate the localization results. The back-propagation is designed to start from the last convolution layer, specifically the layer prior to the classification layer. The local maximums in this layer are detected and back-propagated layer by layer until reaching the input layer. In this way, salient regions of both known and unknown categories, which are activated layer-wisely, can be seamlessly located on the original image. In this section, building upon an off-the-shelf pre-trained CNN, we present an end-to-end instance-level feature extraction framework.

3.1 Activated Region Localization

The forward-pass of a pre-trained network $\mathcal{N}$ produces a series of feature maps for an image I. Denoting $X \in \mathbf{R}^{W \times H \times C}$ as the feature maps of the last convolution layer, the activation map is defined as the mean feature map of X, namely $\overline{X} \in \mathbf{R}^{W \times H}$. The map, which can be easily obtained by taking the average of X over C channels, signals the presence of instances (Figure 2). The regions where $\overline{X}$ exhibits high values give clues to the confidence and positions of object instances. In a typical CNN, the peaks on the feature maps are assembled to support the decision making in the next classification layer. Notice that $\overline{X}$ is prior to the classification layer, and even the responses from unknown categories are visible as they are not suppressed by the classification layer. Under this observation, the local maximums are detected on $\overline{X}$ with a window size 3×3. These local maximums are viewed as the response peaks that network $\mathcal{N}$ discovers on image I.

Denote the set of peaks as Q, where each peak q is attached with x-y position and a response value as the confidence score. Given the position of each peak $q \in Q$, a probability

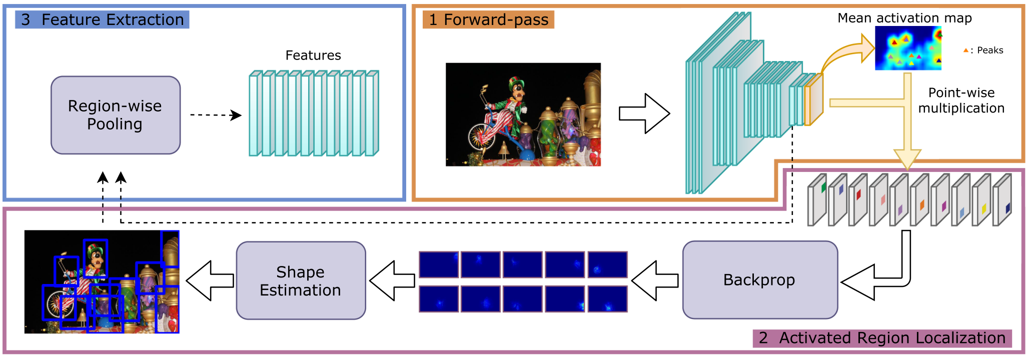


Fig. 2. The pipeline of instance-level features extraction based on activated salient instance region. The mean activation map generated by a single forward-pass indicates the response of each potential instance region. The pattern localization process further localizes each salient region with a bounding box through a back-propagation and a shape estimation module. Final feature representations are built upon those localized boxes.

back-propagation process is adopted to locate the corresponding salient region on the input image. Following a similar process as Zhang et al. [44] except starting from the last convolution layer, a top-down attention model is introduced to identify task-relevant input neurons that support the response peak q in the last convolution layer.

Given that no sub-sampling is performed in the convolution network, the convolution filter of one intermediate convolution layer is denoted as $F \in \mathbf{R}^{W_f \times H_f \times C_{out} \times C_{in}}$, where $W_f \times H_f$ is the spatial size of a filter. C_{in} and C_{out} are the channel dimensions of input and output feature maps, respectively. The input and output feature maps of this convolution layer are denoted as A and B. The activation from each spatial location in A and B could be accessed by $A_{x,y}$ and $B_{i,j}$, respectively. The trained weights related to $A_{x,y}$ and $B_{i,j}$ are accessed with $F_{x-i,y-j}$. The feed-forward process to generate the output tensor B is formulated as

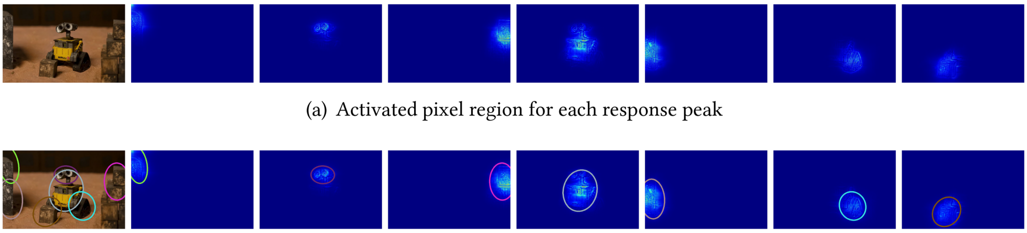
$$B_{i,j} = \sigma\left(\sum_{x=i-\frac{W_f}{2}}^{i+\frac{W_f}{2}} \sum_{y=j-\frac{H_f}{2}}^{j+\frac{H_f}{2}} F_{x-i,y-j} A_{x,y} + b \right), \tag{1}$$

where b is the bias of convolution layer and σ represents the non-linear activation function.

Now let us consider back-propagating peak pixels in the last output layer B. Notice that only the peak positions detected from the last convolution layer are considered. Precisely, the idea is to identify the positions in A that contribute to the score of response peak at $B_{i,j}$ (i.e., q). Following Zhang et al. [44] and Zhou et al. [46], this issue is modeled as a prior probability distribution $P(A_{x,y})$ over output response. $B_{i,j}$ is assumed to be the only winner that takes responses from all positions in A. Therefore, given that $P(B_{i,j})$ and $P(A_{x,y}|B_{i,j})$ are known, we are able to work out $P(A_{x,y})$, namely the probability that the task-relevant neurons in B come from $A_{x,y}$.

For computational convenience, $P(B_{i,j})$ is approximated by $B_{i,j}$ in the last convolution layer. As a consequence, the prior probability of input A is given as

$$P(A_{x,y}) = \sum_{i=x-\frac{W_f}{2}}^{x+\frac{W_f}{2}} \sum_{j=y-\frac{H_f}{2}}^{y+\frac{H_f}{2}} P(A_{x,y}|B_{i,j}) P(B_{i,j}). \tag{2}$$



(a) Activated pixel region for each response peak

(b) Ellipse estimation in each activated region

Fig. 3. The illustration of DASRs in an input image and ellipse estimation of each activated region. The first row shows the input image and activated regions with all seven detected peaks. The corresponding estimated ellipse for each activated region is shown in the second row.

In Equation (2), the conditional probability $P(A_{x,y}|B_{i,j})$ is defined as

$$P(A_{x,y}|B_{i,j}) = \begin{cases} Z_{i,j} A_{x,y} F_{x-i,y-j}, & \text{if } F_{x-i,y-j} > 0 \\ 0, & otherwise, \end{cases} \quad (3)$$

where $Z_{i,j}$ is a normalization factor to make sure that $\sum_{i,j} P(A_{x,y}|B_{i,j}) = 1$. The preceding conditional probability estimates the winning probability of position (x, y) in A given that position (i, j) in B is a winning neuron. The estimation is affected by the activation $A_{x,y}$ and the value within convolution filter $F_{x-i,y-j}$ that relates to $A_{x,y}$ and $B_{i,j}$.

With Equation (2), each position in A is assigned with a probability weight. In the next round of back-propagation, the resulting $P(A_{x,y})$ becomes $P(B_{i,j})$, and $P(A_{x,y}|B_{i,j})$ can be easily estimated in the same manner with Equation (3).

In addition to convolution layers, the back-propagation process also passes through other intermediate layers (e.g., pooling layers). The average-pooling layers are regarded as performing an affine transformation on the response values of the input neurons [44]. Therefore, the average-pooling layer is treated as a convolution layer that is performed within a one-to-one feature map pair. For max-pooling layers, error back-propagation is adopted to perform back-propagation in the work of Zhang et al. [44]. However, blanks are introduced for sub-sampled max-pooling layers. To avoid such blanks, the same back-propagation process as the convolution layer is used for max-pooling layers within one-to-one feature map pairs, with the weights of all-one values.

To this end, all types of layers that the back-propagation may pass through are appropriately considered in the same manner. Equation (2) applies to all layers throughout the convolution network $\mathcal{N}$. Therefore, the back-propagation process proceeds layer by layer smoothly until it reaches the input layer. Finally, the probability that each pixel in image I contributes to a final response peak q is estimated. This leads to a probability map M, which is the same size as image I, for one response peak q. The probabilities in M are normalized to the range $[0, 1]$.

Values in M indicate the degree that corresponding pixels contribute to peak q. Due to the large receptive field of the last convolution layer, pixels that do not contribute to the response peak are still assigned with low probabilities. As a result, the activated region is usually larger than it is supposed to be. A threshold τ is introduced to filter out pixels with little contribution. In this work, τ is fixed to 0.1. As shown in Figure 3, the activated pixels in general concentrate on a local region, which basically implies a potential instance in the image. With all pixels $r(x, y)$ in M that are greater than τ, this activated local region is approximated by an ellipse. The parameters of the ellipse are regularized by the second moment matrix derived from all pixels $r(x, y)$:

$$\sum_{r(x,y) \geq \tau} \begin{bmatrix} x^2 & x \cdot y \\ x \cdot y & y^2 \end{bmatrix}. \quad (4)$$

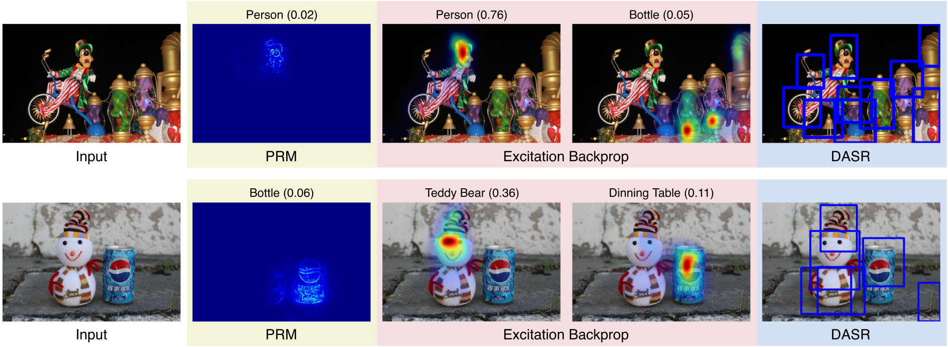


Fig. 4. Examples showing the instances localized by PRM, Excitation Backprop, and DASR. The results of PRM and Excitation Backprop are listed along with the predicted classes and their confidence scores. The results of DASR are highlighted with bounding boxes for presentation clarity. Notice that there are several unknown objects being localized by DASR for both examples. For example, the bell and beams are the unknown categories being successfully detected in the first image by DASR. Similarly, the snowman and plant are successfully localized by DASR.

Figure 3(a) illustrates the probability maps produced from seven response peaks in one image. The corresponding shape estimation results are shown in the second row of Figure 3. As shown in the figure, each detected region corresponds to one salient region in the image. It could cover an entire instance or a major salient region of an instance. The final localization bounding box is the circumscribed rectangle of the estimated ellipse. The feature used to describe this detected region could be derived from the corresponding area of a feature map. Since this feature is produced by activating the salient region via a deep convolution network, it is called *deeply activated salient region* (DASR).

Discussion. Note that the proposed back-propagation can start from the response peaks of any convolution layer. The peaks in a shallow layer correspond to regions with more fine-grained local patterns. In our case, the aim is to discover latent instances. The last convolution layer is the one that directly supports the classification decision. A high response peak in this layer is an integral of visual clues from one instance of a known or unknown category. One or several response peaks of one category from this layer are further integrated by the next layer to make a classification decision. It is therefore appropriate to choose the last convolution layer in our case. However, it is possible to select other layers when the task changes. Our approach offers a generic pipeline for feature extraction applicable to any pre-trained CNNs for image classification.

To contrast the difference between starting back-propagation from the last convolution layer and the prediction layer, Figure 4 shows examples comparing DASR with PRM [46] and Excitation Backprop [44]. PRM only manages to detect instances of a known category by the CNN. By selecting more than one class with high confidence scores, Excitation Backprop produces multiple instances. DASR, being class-agnostic, is capable of detecting multiple regions of instances regardless of whether these regions belong to known or unknown classes. Note that Excitation Backprop indeed produces one activation map per class. When there are multiple instances of a class, the response peaks are not guaranteed to highlight all of the instances. Furthermore, the peaks may not be easily separated for localization of different instances. These properties make the result unpredictable, adversely affecting the robustness of instance search. DASR, by propagating from the last convolution layer, does not suffer from these problems because instances are not suppressed in the prediction layer to optimize classification performance.

3.2 Enhanced Instance Region Detector

In the preceding activation process, only the pixels with peak responses in the last convolution layer are back-propagated. In practice, it is possible that more than one instance shares one peak response as they are close to each other. In this case, a detected salient region will only cover one of the instances. The other neighboring instances are overshadowed. Different from Zhou et al. [46], to alleviate this issue, we consider to back-propagate more pixels in $\overline{X}$. Specifically, all pixels whose responses are higher than the average value of $\overline{X}$ are back-propagated one by one. As a result, a greater number of salient regions are produced than before. It is possible that two salient regions overlap each other and cover the same instance. To reduce the representation redundancy and select out the most salient regions, NMS is employed.

The NMS is operated as follows. The **intersection-over-union (IoU)** threshold of NMS is given as β. Each candidate salient region is attached with a corresponding response value in $\overline{X}$. NMS starts by selecting the salient regions with the highest score uniformly across the space of $\overline{X}$. The remaining regions are screened by comparing their IoU with the set of selected salient regions. Specifically, a region is discarded if its IoU with one of the already selected regions is greater than β. The valid setting for parameter β is further studied in Section 4.

With the new detection procedure, salient regions that attain the highest response in a local are kept, whereas the regions from other potential instances, which have been over-shadowed before, could be activated as long as their overlapping with the most salient region in the local is below a threshold. On average, 12 regions (in contrast to 7 regions before)[1] are detected in one image after NMS when β is set to 0.3. This enhanced detector is called the *DASR**. Its effectiveness is further verified in Section 4. In terms of speed efficiency, it takes 0.25 and 2.02 seconds for DASR and DASR*, respectively,[2] to process one image. DASR* is around 10 times slower than DASR, as DASR* back-propagates 10 times more peak pixels. DASR* can be sped up as the back-propagation can be undertaken in parallel.

3.3 Feature Description

The descriptor of a salient region can be extracted by max-pooling or average-pooling over the feature maps of its corresponding location. The feature descriptor will be compact and uniform in length. In our work, average-pooling is adopted as the default configuration. The performance of max-pooling is also reported in our experimental study in the following section. As one will see, average-pooling is relatively stable despite the fact that the performance from both strategies is close. Theoretically speaking, a feature map from any layer could be used to derive the feature descriptor. However, the distinctiveness varies from layer to layer. For instance, we find that a feature derived from "Block4" in ResNet-50 [9] shows considerably better performance over other layers across different datasets. The details will be followed up in the ablation study. The generated features are first l_2-normalized, then PCA whitening is applied before the second round of l_2-normalization. We call the instance-level feature a *DASR descriptor*.

4 EXPERIMENTS

4.1 Datasets and Experimental Setup

The proposed instance-level feature DASR is evaluated in the instance search task. Instance search is conducted on three benchmarked datasets: Instance-160 [43], Instance-335, and INSTRE [39]. Instance-160 and Instance-335 datasets are derived from the video sequences originally used

[1] The statistics are made on 1 million images crawled from Flickr.

[2] The statistics are conducted on three datasets.

for single visual object tracking evaluation. In Instance-160, there are 160 queries and 11,885 reference images. The query instances belong to 80 object categories labeled in the Microsoft COCO dataset [21]. To test the scalability of the proposed instance-level feature, Instance-160 is augmented with 175 extra queries that are harvested from GOT-10K [11], YouTube-BoundingBoxes [29], and LaSOT [5]. These video datasets were originally designed for object tracking evaluation. These newly added 175 query instances are mostly outside the coverage of Microsoft COCO 80 categories, and the backgrounds are under severe variations. This leads to an augmented evaluation dataset Instance-335. In this dataset, there are 335 queries and 40,914 reference images. In the INSTRE dataset, there are 28,543 images in total. Following the evaluation protocol in the work of Iscen et al. [12], 1,250 images[3] are treated as queries, leaving the remaining 27,293 images as references. For all three datasets, the bounding boxes are provided both in the query and in the relevant reference images.

In our implementation, the images of these datasets are re-sized to 512 pixels on the long side while preserving the aspect ratio of the original images. Following the convention in the literature, the search performance is measured with **mean average precision (mAP)**. For Instance-160 and Instance-335, the search performance is evaluated with varying top-k, where k is set to 50, 100, and the number of images in a dataset. This is because the number of true positives for each instance query varies from several to a few hundred for both Instance-160 and Instance-335.

The proposed feature extraction can be carried out using any CNN classification network. Here, we report experimental results based on ResNet-50 and VGG-16, which are widely used by different applications. As revealed in the later experiment, the performance with ResNet-50 is considerably higher. Hence, most of the presented results will be based on ResNet-50 by default unless otherwise stated. The feature extraction is implemented under the TensorFlow framework. Experiments are run on an Nvidia GTX 1080 Ti.

In the first experiment, an ablation study is conducted to investigate the suitability of feature maps at different layers for feature extraction. In addition, we also verify the parameter setting in NMS (i.e., the IoU rate β for pruning instance candidates). The distinctiveness of DASR is further studied when it is under PCA dimension reduction. To this end, three groups of comparative studies are presented. The performance of DASR is studied in comparison to R-MAC [37], CroW [16], CAM [15], BLCF [24], BLCF-SalGAN [24], Regional Attention [17], DeepVision [32], FCIS+XD [43], and PCL*+SPN [20] in the instance search task. Note that the comparison is based on instance-level matching.

4.2 Ablation Study

4.2.1 Feature Selection. Given the detected salient region, the feature map from each convolutional layer could be used to derive the feature descriptor. Nevertheless, it has been widely witnessed that the search performance varies across different layers [2, 20, 43]. For this reason, ablation analysis is conducted to seek for the best suitable layer for instance search. Layers from the last two blocks of ResNet-50, namely Block3 and Block4, are investigated since deeper layers are observed to contain semantic-level information. Following the original implementation of ResNet-50, six and three bottlenecks are built within Block3 and Block4, respectively. The output feature maps from preceding nine bottlenecks are respectively used to derive features for DASR regions. The first bottleneck in Block3 is given as Block3_unit1, and the rest are denoted in the same manner. For the VGG-16 network, the back-propagation starts from the feature map of the fifth pooling layer, which is the last layer prior to the fully connected layers. Features are extracted from feature maps of the three convolutional layers on the fifth stage. They are given as conv5_1,

[3] One query is selected from one image.

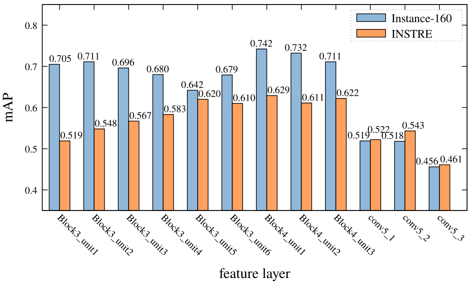


Fig. 5. Performance of DASR on Instance-160 and INSTRE datasets with features derived from different convolutional layers of ResNet-50 and VGG-16.

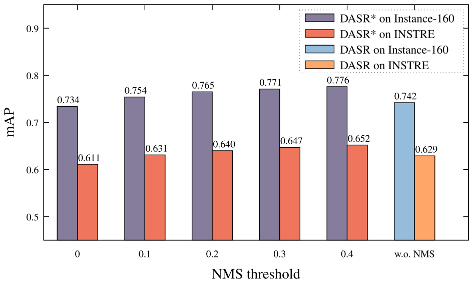


Fig. 6. The performance of DASR on Instance-160 and INSTRE datasets with different NMS threshold β and without NMS.

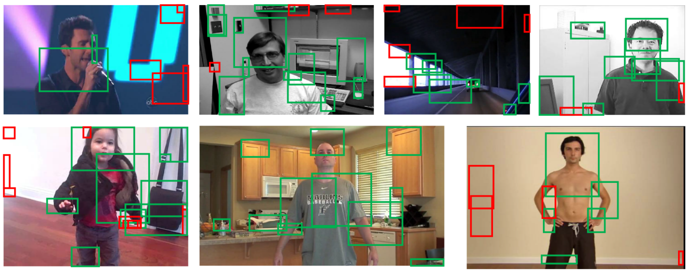


Fig. 7. The illustration of meaningless regions detected by DASR. Due to noises, the objectless region is activated in the CAMs. These regions will be falsely viewed as instance regions by our approach.

conv5_2 and conv5_3, respectively. In the experiment, there was no NMS adopted in the region detector.

Figure 5 shows the performance of DASR on Instance-160 and INSTRE with features output from different layers of two backbones. A wide performance gap is observed between the networks. The gap is due to the difference in encoded patterns and feature discriminability between the networks. The result of instance localization is directly influenced by the encoded patterns. In general, the regions derived via ResNet-50 show high localization accuracy. Moreover, feature maps from ResNet-50 are more discriminative than those of VGG-16, which is in line with the observations from many other works. Overall, features derived from Block4_unit1 show the best performance on both datasets. As a result, it is selected as the default configuration in the rest of the experiments.

Due to the interference from the noise, not all detected regions are meaningful. Regions that barely cover any instance in the image can be falsely detected. As shown in Figure 7, the regions within the red bounding box are detected as potential instance regions, although they are actually flat patterns. Features will be extracted from these regions by average-pooling as with the green ones. Nevertheless, the queries supplied by the users are assumed to be meaningful instances. These meaningless regions in the repository will be hardly matched to the query instances due to the apparent visual differences. As a result, they will make the repository larger than we expected but have little impact on the instance search quality.

4.2.2 Configurations on DASR.* In the second study, we further investigate the effectiveness of the enhanced detector DASR* and the appropriate setting for overlapping rate parameter β in NMS. In this study, the enhanced detection procedure presented in Section 3.2 is performed on

Table 1. Performance Comparison with Dimension Reduction

Method	Dim.	Instance-160			Instance-335			INSTRE
		Top-50	Top-100	All	Top-50	Top-100	All	All
DASR	2,048	0.591	0.680	0.742	0.419	0.558	0.699	0.629
	1,024	0.574	0.660	0.721	0.406	0.543	0.682	0.609
	512	0.543	0.624	0.684	0.389	0.521	0.656	0.583
DASR*	2,048	0.614	0.711	0.771	0.433	0.580	0.724	0.647
	1,024	0.599	0.690	0.751	0.424	0.567	0.711	0.626
	512	0.575	0.662	0.723	0.408	0.548	0.687	0.606

ResNet-50. Performance under different settings of β is presented in Figure 6. The performance is also compared to the one without NMS. As shown in the figure, DASR* outperforms DASR when the overlapping rate is higher than 0.1. Moreover, the larger overlapping rate β leads to better performance, since more salient regions are kept for one image. The highest performance is attained when $\beta = 0.4$, which also leads to much more number of detected regions. Specifically, the number of detected regions is roughly doubled over the case of being without NMS. As a trade-off between performance and computational cost, β is set to 0.3 in the rest of our experiments.

4.2.3 Dimension Reduction. In this experiment, we investigate the distinctiveness of our feature with different degrees of dimension reduction by PCA. The original feature dimension is 2,048. The performance trend of DASR and DASR* is studied when they are further reduced to 1,024 and 512 by PCA, respectively. The performance of different feature dimensions is reported on Instance-160, Instance-335, and INSTRE and presented in Table 1. As shown in the table, both DASR and DASR* suffer 3% to 4% performance degradation when they are projected to 512 dimensions. Notice that DASR is derived from the pre-trained model. The features cannot be as compact as features that are derived from fine-tuned models, such as PCL*+SPN. However, as one will see in the following experiments, they are still superior over most of the existing approaches that are of similar feature size.

4.3 Instance Search

4.3.1 Comparison to State-of-the-Art Approaches. DASR is compared against several representative approaches in the literature. The approaches are categorized according to the degree of supervision involved to train a network for instance search. The first group of approaches capitalizes on the convolutional features derived from pre-trained CNNs without model fine-tuning. These approaches include R-MAC [37], CroW [16], CAM [15], BLCF [24], BLCF-SalGAN [24], and Regional Attention [17]. In contrast, the second group of approaches fine-tune the pre-trained CNNs with extra training examples in the COCO dataset. The only approach that falls into this group is PCL*+SPN [20]. The last group of approaches includes DeepVision [32] and FCIS+XD [43], which leverage Faster R-CNN [31] and FCN, respectively, to extract features. As with the second group, the object detection models are also fine-tuned with the training data in the COCO dataset. Additionally, object-level labels are required. For instance, DeepVision is trained with object bounding boxes, whereas FCIS+XD requires the instance masks of objects. Note that DeepVision also adopts re-ranking and query expansion strategies to improve search performance.

During the retrieval, the first group of approaches collapse all features into one vector for retrieval, which makes the feature representations unfeasible to separate individual instances. Furthermore, most of them are only capable of highlighting the latent objects coarsely with an attention map. The latent instances are not localized explicitly. Thanks to the detection backbones,

Table 2. Performance Comparison on Instance-160 and Instance-335

(a) Instance-160

Approach	Model Type	Loc.	Dim.	Top-50	Top-100	All	TM
R-MAC [37]	Pre-trained	Image	512	0.268	0.307	0.358	0.066
CroW [16]	Pre-trained	Image	512	0.239	0.284	0.338	0.061
CAM [15]	Pre-trained	Image	512	0.256	0.302	0.358	1.111
BLCF [24]	Pre-trained	Image	336	0.487	0.592	0.653	0.230
BLCF-SalGAN [24]	Pre-trained	Image	336	0.493	0.596	0.656	0.269
Regional Attention [17]	Pre-trained	Image	2,048	0.318	0.389	0.459	0.094
DeepVision [32]	Strong	Region	512	0.541	0.666	0.731	0.131
FCIS+XD [43]†	Strong	Pixel	1,536	0.575	0.659	0.724	0.874
PCL*+SPN [20]	Weak	Region	1,024	0.583	0.661	0.724	1.116
DASR	Pre-trained	Region	2,048	0.591	0.680	0.742	0.284
DASR*	Pre-trained	Region	2,048	**0.614**	**0.711**	**0.771**	2.368
DASR-m	Pre-trained	Region	2,048	0.553	0.640	0.700	0.282
DASR-m*	Pre-trained	Region	2,048	0.579	0.673	0.733	2.328

(b) Instance-335

Approach	Model Type	Loc.	Dim.	Top-50	Top-100	All
R-MAC [37]	Pre-trained	Image	512	0.234	0.315	0.375
CroW [16]	Pre-trained	Image	512	0.159	0.225	0.321
CAM [15]	Pre-trained	Image	512	0.194	0.263	0.347
BLCF [24]	Pre-trained	Image	336	0.246	0.358	0.483
BLCF-SalGAN [24]	Pre-trained	Image	336	0.245	0.350	0.469
Regional Attention [17]	Pre-trained	Image	2,048	0.242	0.351	0.488
DeepVision [32]	Strong	Region	512	0.402	0.521	0.620
FCIS+XD [43]	Strong	Pixel	1,536	0.403	0.500	0.593
PCL*+SPN [20]	Weak	Region	1,024	0.380	0.475	0.580
DASR	Pre-trained	Region	2,048	0.419	0.558	0.699
DASR*	Pre-trained	Region	2,048	**0.433**	**0.580**	**0.724**
DASR-m	Pre-trained	Region	2,048	0.411	0.533	0.662
DASR-m*	Pre-trained	Region	2,048	0.428	0.560	0.694

Note: The average time cost (TM/s) of processing one image is reported in the last column of (a).
†Results are cited from the referred paper.

the second and third groups, instead, localize instances with bounding boxes or masks and treat each instance individually as a retrieval unit. Specifically, all instances from each reference image are compared to the query instance, and the similarity is set equal to the instance with the highest matching score. DASR and DASR*, similar to the first group, require only pre-trained CNN. However, as in the second and third groups, the localized bounding boxes for instances are generated and the extracted instances from an image are treated independently during retrieval. For convenience, we name the three groups of approaches *pre-trained*, *weak*, and *strong*, respectively.

Not all approaches considered in the comparison are able to localize the instances in an image. From this sense, approaches are grouped into “image,” “region,” and “pixel.” For the approaches labeled as “image,” they are unable to localize the instances in the retrieved image. For the approaches labeled as “region,” they are able to localize instance with a bounding box. For the approaches labeled as “pixel,” they are able to localize the instance at a pixel-level.

Tables 2 and 3 show the performance of different approaches on three datasets. In general, all approaches show steady performance degradation when being tested on the more challenging dataset Instance-335. Among all of the approaches, DeepVision, FCIS+XD, PCL*+SPN, and DASR that produce instance-level features demonstrate better performance on all datasets. The instance-level features are more robust to background variations, which has been well illustrated in the

Table 3. Performance Comparison on INSTRE

Approach	Model Type	Loc.	Dim.	All	TM
R-MAC [24]†	Pre-trained	Image	512	0.523	0.115
CroW [24]†	Pre-trained	Image	512	0.416	0.082
CAM [15]	Pre-trained	Image	512	0.320	1.178
BLCF [24]†	Pre-trained	Image	336	0.636	0.183
BLCF-SalGAN [24]†	Pre-trained	Image	336	**0.698**	0.239
Regional Attention [17]	Pre-trained	Image	2,048	0.542	0.101
DeepVision [32]	Strong	Region	512	0.197	0.148
FCIS+XD [43]	Strong	Pixel	1,536	0.067	1.794
PCL*+SPN [20]†	Weak	Region	1,024	0.575	1.359
DASR	Pre-trained	Region	2,048	0.629	0.308
DASR*	Pre-trained	Region	2,048	0.647	2.626
DASR-m	Pre-trained	Region	2,048	0.671	0.307
DASR-m*	Pre-trained	Region	2,048	0.692	2.644

Note: The average time cost (TM/s) of processing one image is reported in the last column.
†Results are cited from the referred paper.

work of Lin et al. [20]. Among these instance-level features, DASR and DASR* show consistently satisfactory performance on both datasets. In contrast, the performance from FCIS+XD drops considerably on Instance-335 and INSTRE. This is simply because there are many instance categories outside the coverage of Microsoft COCO-80 on which FCIS+XD training fully relies. Interestingly, DASR* even outperforms FCIS+XD considerably on Instance-160, where all query instances are well trained in FCIS+XD. FCIS+XD is capable of generating more precise instance regions. The reason that DASR* outperforms FCIS+XD mainly is attributed to the better discriminativeness of the feature maps. Notice that the feature maps in FCIS+XD are trained for instance segmentation. It carries more localization information rather than semantic information of an instance. BLCF-SalGAN, although showing the overall best performance on INSTRE, is sensitive to various image transformations. This is evidenced on the datasets Instance-160 and Instance-335, where its performance is not satisfactory with the presence of non-rigid transformation.

As shown in the tables, only approaches such as DeepVision, FCIS+XD, PCL*+SPN, and DASR are able to localize the instances in the image. Among them, DeepVision only localizes the instances in the top-ranked images because of its high computational complexity. FCIS+XD is able to localize instances on pixel-level. However, pixel-wise annotation is required. It fails for instances of unknown categories. Compared to other pre-trained approaches, DASR is several times faster than BLCF, BLCF+SalGAN, and CAM. It is the only pre-trained approach that is capable of both feature extraction and instance localization.

In Tables 2 and 3, we also report the results of DASR and DASR* with max-pooling strategy in the feature extraction. They are denoted as DASR-m and DASR-m*, respectively. Compared to average-pooling, the performance from max-pooling degrades by several percentages on Instance-160 and Instance-335 datasets, although showing slightly better performance on the INSTRE dataset. The major reason is that average-pooling strategy accumulates more background information compared to max-pooling. On Instance-160 and Instance-335, the interference from the backgrounds is minor as the query and the reference images share similar backgrounds. However, on INSTRE, the instances are embedded into different backgrounds, and max-pooling turns out to be more robust in this case. Overall, both pooling strategies show superior performance over the existing approaches. In practice, it is left to the reader to select the appropriate pooling strategy according to the application context.

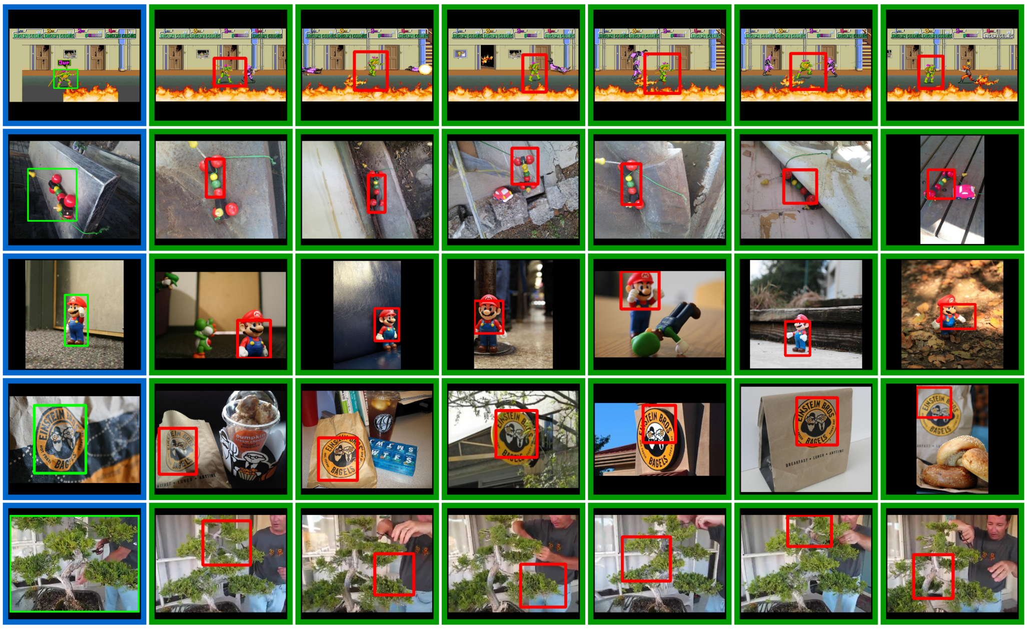
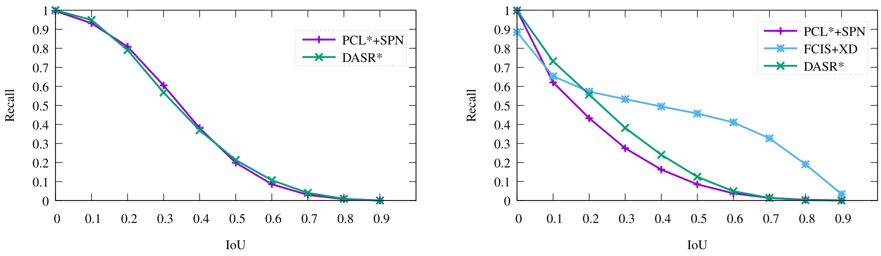


Fig. 8. Examples of the top-six retrieved instances by five query examples in Instance-335 and INSTRE. The leftmost column shows the queries, whereas the remaining columns display the retrieved images sorted in descending order. The true-positive and false-positive images are enclosed by green and red borders, respectively.

In the last column of Tables 2(a)[4] and 3, the average time costs of extracting features from one image for all approaches are presented. The trends of time complexity on the two datasets are generally similar. The approaches that are only able to extract image level features show relatively higher speed efficiency since no instance region localization operation is involved. Among these approaches, CAM [15], BLCF [24], and BLCF-SalGAN [24] are considerably slower than others. For CAM [15], the operation of extracting CAMs is computationally intensive. The weighting strategy in BLCF [24] requires additional feature post-processing. BLCF-SalGAN [24] additionally employs SalGAN [26] to extract saliency maps. For the approaches that are able to localize instance regions, they are usually slower because additional costs are spent on instance localization. Our approach DASR is relatively fast among region-level approaches. DASR* turns out to be much slower due to more regions to be localized. Additionally, extracting features by max-pooling (namely, DASR-m and DASR-m*) has similar costs as that of average-pooling (DASR and DASR*).

Top-6 retrieval results from our approach are illustrated in Figure 8. The first two rows show a cartoon character and a toy caterpillar as the query instances, which do not belong to any known categories in the COCO dataset. DASR* successfully retrieves and localizes the instances in the top-six ranked images. This does indicate that the proposed back-propagation mechanism is able to capture the instances of categories new to a pre-trained network. However, DASR* could be sensitive to instances with similar shape or appearance but different in details. One example is shown in the fourth row, where a logo with different printing and icon from the query logo is retrieved. Since DASR is an unsupervised approach, one could not expect precise localization for the retrieved instances. However, as shown in a later experiment, its localization accuracy is

[4] Since the time costs we observed on Instance-335 are very close to that of Instance-160, they are not reported.



(a) INSTRE (b) Instance-335

Fig. 9. Recall-IoU curves on INSTRE and Instance-335.

actually higher than weakly supervised approaches such as PCL*+SPN. The last row shows a typical example where our approach fails due to the small size of the detected object.

4.3.2 Instance Localization Accuracy. In this experiment, we further study how well the detected regions overlaps with instances in the ground truth. Following Hosang et al. [10], recall is adopted to measure the fraction of a detected region that overlaps with its ground-truth instance based on the IoU threshold. Different values of recall with varying IoU thresholds are reported. The experiment compares DASR* with FCIS+XD and PCL*+SPN since these are the only approaches capable of locating object instances. Figure 9 shows the recall-IoU curves of different approaches on two datasets. The localization performance of FCIS+XD is not compared on INSTRE since a large portion of the instance categories are not covered in its training dataset.

In Figure 9(a), our approach shows similar or even slightly better performance over PCL*+SPN, which is designed to localize the whole instance from images. In Figure 9(b), FCIS+XD outperforms the other two approaches with large performance margin. The result is not surprising due to additional use of training examples by FCIS+XD to fine-tune FCN for instance segmentation. FCIS+XD, nevertheless, is hard to scale up to cope with a dataset with unknown categories of instances. Since a whole region of an instance cannot be activated during the backward pass, detecting the whole instance region is hardly achievable based on the pre-trained network alone. This issue is observed in other works [44, 46]. Compared to the weakly supervised approach PCL*+SPN, DASR*, which only leverages a pre-trained model, shows superior performance on INSTRE and better localization accuracy on Instance-335. DASR* is more cost effective in terms of training and generic in detecting instances of both known and unknown categories.

4.3.3 Scalability Test. In this experiment, the scalability of DASR* is further studied on Instance-160 by incrementally adding in 1 million distracting reference images. The 1 million distractors are crawled from Flickr. For each image, the DASR* feature is extracted with the same processing flow as before. A total of 7,014,819 regions are detected by DASR, whereas 12,486,461 regions are detected by DASR*. The scalability of DASR* is studied in comparison to several state-of-the-art approaches ranging from conventional BoVW [34], BoVW+HE [13] approaches, and recent approaches R-MAC, CroW, DeepVision, FCIS+XD, and PCL*+SPN.

The result is shown in Figure 10. Note that the performance of BoVW, BoVW+HE, R-MAC, and CroW is not reported for sizes beyond 100K. This is simply because their performance is already far behind DASR at the size of 100K. Overall, DASR* outperforms all approaches, including FCIS+XD based on a fully supervised model, on three different testing scales. The performance gap is mainly due to the ability of DASR* to detect the salient regions of an instance despite that the regions may not fully occupy the entire instance as FCIS+XD. The salient regions play an important role in guaranteeing that the features generated by DASR are more discriminative.

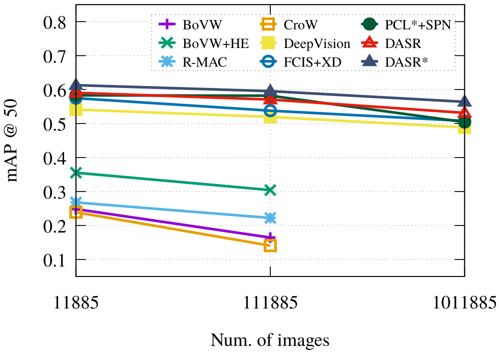


Fig. 10. Scalability test on Instance-160 in comparison with several state-of-the-art approaches. The performance is measured by mAP@top-50 and reported as a function about the number of reference images.

5 CONCLUSION

We have presented our solution for visual instance search. The focus is on the instance-level feature representation. A novel feature descriptor, namely DASR, is proposed. The features are extracted from the semantically salient regions of an image that are activated by a back-propagation process. Both the instance localization and the instance-level feature description are achieved on a pre-trained classification network, without any further fine-tuning. This approach is generic in the sense that the back-propagation could be built upon any pre-trained CNNs classification network. Since no fine-tune training is required, the descriptor remains effective for instances from both known and unknown categories, which is hardly achievable with the existing approaches. Since DASR is built based on a pre-trained classification network only, one cannot expect to detect the region of a complete instance in all cases. Recently, we improved the instance localization based on the clues provided by the query instance [41]. This is similar to the one-shot object detection that is applied on top-ranked search results.

In addition to instance search, our approach is also potentially useful for search-driven annotation. Specifically, an annotator only needs to label a few examples of instances for an object category as queries. By our approach, all instances of that category can be automatically retrieved from an image or video collection, with ellipses or bounding boxes indicating the instance positions. In such a way, labeling effort can be significantly reduced by requiring only the adjustment of the bounding boxes of instances.

REFERENCES

[1] George Awad, Wessel Kraaij, Paul Over, and Shin'ichi Satoh. 2017. Instance search retrospective with focus on TRECVID. *International Journal of Multimedia Information Retrieval* 6, 1 (2017), 1–29.

[2] Artem Babenko, Anton Slesarev, Alexandr Chigorin, and Victor Lempitsky. 2014. Neural codes for image retrieval. In *European Conference on Computer Vision*. Springer, 584–599.

[3] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. 2006. Surf: Speeded up robust features. In *European Conference on Computer Vision*. Springer, 404–417.

[4] Matthijs Douze, Herve Jegou, Harsimrat Sandhawalia, Laurent Amsaleg, and Cordelia Schmid. 2009. Evaluation of GIST descriptors for web-scale image search. In *Proceedings of the ACM International Conference on Image and Video Retrieval*. 1–8.

[5] Heng Fan, Liting Lin, Fan Yang, Peng Chu, Ge Deng, Sijia Yu, Hexin Bai, Yong Xu, Chunyuan Liao, and Haibin Ling. 2019. LaSOT: A high-quality benchmark for large-scale single object tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 5374–5383.

[6] Ross Girshick. 2015. Fast R-CNN. In *Proceedings of the IEEE International Conference on Computer Vision*. 1440–1448.

[7] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 580–587.

[8] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. 2017. Mask R-CNN. In *Proceedings of the IEEE International Conference on Computer Vision.* 2961–2969.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 770–778.

[10] Jan Hosang, Rodrigo Benenson, Piotr Dollár, and Bernt Schiele. 2015. What makes for effective detection proposals? *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38, 4 (2015), 814–830.

[11] Lianghua Huang, Xin Zhao, and Kaiqi Huang. 2019. GOT-10k: A large high-diversity benchmark for generic object tracking in the wild. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 99 (2019), 1.

[12] Ahmet Iscen, Giorgos Tolias, Yannis Avrithis, Teddy Furon, and Ondrej Chum. 2017. Efficient diffusion on region manifolds: Recovering small objects with compact CNN representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 2077–2086.

[13] Herve Jegou, Matthijs Douze, and Cordelia Schmid. 2008. Hamming embedding and weak geometric consistency for large scale image search. In *European Conference on Computer Vision.* Springer, 304–317.

[14] Herve Jegou, Florent Perronnin, Matthijs Douze, Jorge Sánchez, Patrick Perez, and Cordelia Schmid. 2011. Aggregating local image descriptors into compact codes. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34, 9 (2011), 1704–1716.

[15] Albert Jimenez, Jose M. Alvarez, and Xavier Giró-i-Nieto. 2017. Class-weighted convolutional features for visual instance search. In *Proceedings of the British Machine Vision Conference.*

[16] Yannis Kalantidis, Clayton Mellina, and Simon Osindero. 2016. Cross-dimensional weighting for aggregated deep convolutional features. In *European Conference on Computer Vision.* Springer, 685–701.

[17] Jaeyoon Kim and Sung-Eui Yoon. 2018. Regional attention based deep feature for image retrieval. In *Proceedings of the British Machine Vision Conference.* 209.

[18] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2012. ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems.* 1097–1105.

[19] Yi Li, Haozhi Qi, Jifeng Dai, Xiangyang Ji, and Yichen Wei. 2017. Fully convolutional instance-aware semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 2359–2367.

[20] Jie Lin, Yu Zhan, and Wan-Lei Zhao. 2019. Instance search based on weakly supervised feature learning. *Neurocomputing* 424 (2019), 117–124.

[21] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision.* Springer, 740–755.

[22] David G. Lowe. 2004. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision* 60, 2 (2004), 91–110.

[23] Oded Maron and Tomás Lozano-Pérez. 1998. A framework for multiple-instance learning. In *Advances in Neural Information Processing Systems.* 570–576.

[24] Eva Mohedano, Kevin McGuinness, Xavier Giró-i-Nieto, and Noel E. O'Connor. 2018. Saliency weighted convolutional features for instance search. In *Proceedings of the International Conference on Content-Based Multimedia Indexing*. IEEE, Los Alamitos, CA, 1–6.

[25] Hyeonwoo Noh, Andre Araujo, Jack Sim, Tobias Weyand, and Bohyung Han. 2017. Large-scale image retrieval with attentive deep local features. In *Proceedings of the IEEE International Conference on Computer Vision.* 3456–3465.

[26] Junting Pan, Cristian Canton-Ferrer, Kevin McGuinness, Noel E. O'Connor, Jordi Torres, Elisa Sayrol, and Xavier Giró-i-Nieto. 2017. SalGAN: Visual saliency prediction with generative adversarial networks. *arXiv:1701.01081* (2017).

[27] Mattis Paulin, Matthijs Douze, Zaid Harchaoui, Julien Mairal, Florent Perronin, and Cordelia Schmid. 2015. Local convolutional features with unsupervised training for image retrieval. In *Proceedings of the IEEE International Conference on Computer Vision.* 91–99.

[28] Ali S. Razavian, Josephine Sullivan, Stefan Carlsson, and Atsuto Maki. 2016. Visual instance retrieval with deep convolutional networks. *ITE Transactions on Media Technology and Applications* 4, 3 (2016), 251–258.

[29] Esteban Real, Jonathon Shlens, Stefano Mazzocchi, Xin Pan, and Vincent Vanhoucke. 2017. YouTube-BoundingBoxes: A large high-precision human-annotated data set for object detection in video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 5296–5305.

[30] Joseph Redmon and Ali Farhadi. 2018. YOLOv3: An incremental improvement. *arXiv preprint arXiv:1804.02767* (2018).

[31] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems.* 91–99.

[32] Amaia Salvador, Xavier Giró-i-Nieto, Ferran Marqués, and Shin'ichi Satoh. 2016. Faster R-CNN features for instance search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 9–16.

[33] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).

[34] Josef Sivic and Andrew Zisserman. 2003. Video Google: A text retrieval approach to object matching in videos. In *Proceedings of the IEEE International Conference on Computer Vision*. 1470–1477.

[35] Peng Tang, Xinggang Wang, Song Bai, Wei Shen, Xiang Bai, Wenyu Liu, and Alan Yuille. 2018. PCL: Proposal cluster learning for weakly supervised object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42, 1 (2018), 176–191.

[36] Peng Tang, Xinggang Wang, Xiang Bai, and Wenyu Liu. 2017. Multiple instance detection network with online instance classifier refinement. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2843–2851.

[37] Giorgos Tolias, Ronan Sicre, and Hervé Jégou. 2015. Particular object retrieval with integral max-pooling of CNN activations. *arXiv preprint arXiv:1511.05879* (2015).

[38] Fang Wan, Chang Liu, Wei Ke, Xiangyang Ji, Jianbin Jiao, and Qixiang Ye. 2019. C-MIL: Continuation multiple instance learning for weakly supervised object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2199–2208.

[39] Shuang Wang and Shuqiang Jiang. 2015. INSTRE: A new benchmark for instance-level object retrieval and recognition. *ACM Transactions on Multimedia Computing, Communications, and Applications* 11, 3 (2015), 37.

[40] Xiu-Shen Wei, Jian-Hao Luo, Jianxin Wu, and Zhi-Hua Zhou. 2017. Selective convolutional descriptor aggregation for fine-grained image retrieval. *IEEE Transactions on Image Processing* 26, 6 (2017), 2868–2881.

[41] Hong Yi-Geng, Xiao Hui-Chu, and Zhao Wan-Lei. 2021. Towards accurate localization by instance search. In *Proceedings of ACM International Conference on Multimedia*. 3807–3815.

[42] Matthew D. Zeiler and Rob Fergus. 2014. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision*. Springer, 818–833.

[43] Yu Zhan and Wan-Lei Zhao. 2018. Instance search via instance level segmentation and feature representation. *arXiv preprint arXiv:1806.03576* (2018).

[44] Jianming Zhang, Sarah Adel Bargal, Zhe Lin, Jonathan Brandt, Xiaohui Shen, and Stan Sclaroff. 2018. Top-down neural attention by excitation backprop. *International Journal of Computer Vision* 126, 10 (2018), 1084–1102.

[45] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. 2016. Learning deep features for discriminative localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2921–2929.

[46] Yanzhao Zhou, Yi Zhu, Qixiang Ye, Qiang Qiu, and Jianbin Jiao. 2018. Weakly supervised instance segmentation using class peak response. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3791–3800.

Received 6 July 2020; revised 16 November 2021; accepted 22 December 2021