Attributes Make Sense on Segmented Objects

Zhenyang Li, Efstratios Gavves, Thomas Mensink, and Cees G.M. Snoek

ISLA, Informatics Institute, University of Amsterdam, The Netherlands

Abstract. In this paper we aim for object classification *and* segmentation by attributes. Where existing work considers attributes either for the global image or for the parts of the object, we propose, as our first novelty, to learn and extract attributes on segments containing the entire object. Object-level attributes suffer less from accidental content around the object and accidental image conditions such as partial occlusions, scale changes and viewpoint changes. As our second novelty, we propose joint learning for simultaneous object classification and segment proposal ranking, solely on the basis of attributes. This naturally brings us to our third novelty: object-level attributes for zero-shot, where we use attribute descriptions of unseen classes for localizing their instances in new images and classifying them accordingly. Results on the Caltech UCSD Birds, Leeds Butterflies, and an a-Pascal subset demonstrate that *i)* extracting attributes on oracle object-level brings substantial benefits *ii)* our joint learning model leads to accurate attribute-based classification and segmentation, approaching the oracle results and *iii)* object-level attributes also allow for zero-shot classification and segmentation. We conclude that attributes make sense on segmented objects.

Keywords: attributes, segmentation, zero-shot classification.

1 Introduction

The goal of this paper is object classification *and* segmentation using attributes. Representing an image by attributes [17, 19, 25] like *big ear*, *trunk*, and *gray color* is appealing when examples are rare or non-existent, feature encodings are non-discriminative, or a semantic interpretation of the representation is desired. Consequently, attributes are a promising solution for fine-grained and zero-shot object classification [7], personalized object search [23], object description [17], and many other current challenges in computer vision. Different from existing work, which computes object attributes either on the entire image [1, 25, 28] or on parts of the object [6, 14, 19], we propose to predict the best possible segment that contains the entire object and compute all attributes on this segment.

One approach to object classification by attributes is to compute the attributes globally; see Fig. 1a. Lampert *et al.* [25] introduce a directed graphical attribute model for the recognition of animal categories, even in the absence of training examples, which is called zero-shot classification. Since their model optimizes attribute prediction, and not object classification, Akata *et al.* [1] adapt the model of [39] and propose attribute embedding learning for supervised

D. Fleet et al. (Eds.): ECCV 2014, Part VI, LNCS 8694, pp. 350–365, 2014.

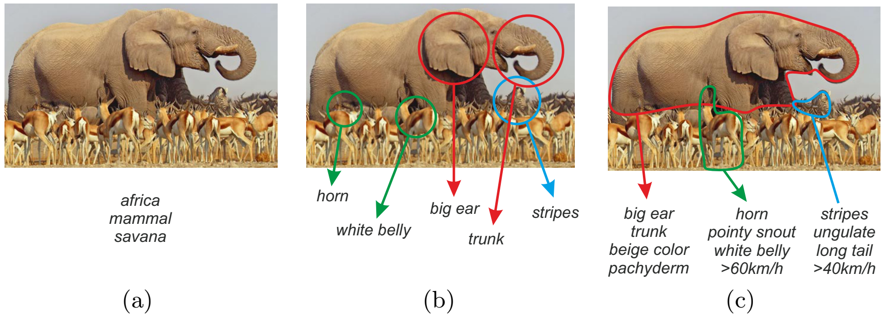


Fig. 1. Attributes make sense on segmented objects. Illustration of different level of attributes, in a picture showing several animals: antelopes, an elephant, and a small zebra to the right of the elephant. (a) Considering the full image, one can only expect to describe generic attributes that apply to the whole scene. (b) When localizing attributes, one faces the problem that not all attributes can be localized. Partial occlusions, small scales, or uncommon viewpoints might reduce the visibility of a particular attribute. Moreover, several class-specific attributes are very hard to localize in practice. (c) We propose to constrain object-specific attributes on segmented objects, which allows for object-specific description and suppresses irrelevant background signal.

and zero-shot object classification. Inspired by Akata *et al.*, we also optimize attribute learning for object classification, including the challenging zero-shot setting. However, we observe that attributes most often reflect object level properties, *e.g.,* that an antelope has a *pointy snout.* Hence, considering the whole image as indicative of the attribute *pointy snout* is counter-intuitive.

The second approach to object classification by attributes focuses on localizing salient parts reflecting those attributes; see Fig. 1b. Ferrari and Zisserman [19] propose local attributes in superpixels and use them for attribute classification in novel images. Similarly, Bourdev *et al.* [6] localize attributes by employing part-based poselet responses, *e.g.,* cropped images of people *wearing hats.* These approaches assume that humans should provide the annotations for existing, pre-defined attributes. For this reason Duan *et al.* [14] propose to discover discriminative and localized attributes which have to be nameable and approved by humans. Jia *et al.* [13] and Wah *et al.* [36] also propose to build interactive systems to discover and localize attributes with humans in the loop. Generally speaking, pinpointing to certain image or object locations allows for finer definition of attributes. However, such methods appear to face certain limitations. For one, challenging image conditions, such as partial occlusion, small scales, or uncommon viewpoints make certain attribute locations visually unidentifiable. Second, certain object attributes simply cannot be localized. For example, localizing the attribute of an elephant being a *pachyderm* is impossible. Hence, an overly precise localization of attributes is often not needed, as attributes could be locally untraceable, either due to their nature or the imaging circumstances.

In this paper we make a case for a third approach in learning and using attributes. More specifically, and as part of our first novelty, we propose to learn and extract attributes from segmented objects. For this we assume category-level attributes, since it is the class and not the instance that pertains the object properties; for example in Fig. 1c. a zebra is a *quadruped* animal, even if just the upper body is visible in this specific picture. As part of our second novelty, we propose joint learning of attributes for simultaneous segmentation and classification of the object. This naturally brings us to our third novelty, which is zero-shot classification. Our joint model performs segmentation and classification using object-level attributes, enabling us to use attribute descriptions of unseen classes to localize their instances in new images and classify them accordingly.

The paper is organized as follows. We first discuss the related work in Section 2. In Section 3 we describe our model and how we efficiently infer the segment from which we extract object-level attributes. In Section 4 we validate our models using three publicly available datasets, Caltech UCSD Birds [37], Leeds Butterflies [38], and a-Pascal [17]. We show experimental results in both a fully supervised classification setting as well as in a zero-shot classification setting. We conclude our paper in Section 5.

2 Related Work

Classification and Segmentation. Embedding locality in image representation has become increasingly popular over the recent years. Several works have shown that localizing the object of interest is beneficial, not only for providing a spatial support for the object [2, 33], but also for classifying it more accurately [11, 20, 32]. Certainly, semantic segmentation and object detection have received most of their attention from attempts at using localities for classification.

Both the state-of-the-art methodologies, for semantic segmentation and object detection, adopt a similar two-step approach. In the first step, object location proposals are extracted. For object detection, bounding box proposals [2, 27, 32] are usually computed on the basis of local region coherence. Moreover, the authors of [4, 9, 15] showed that to obtain accurate object proposals for semantic segmentation, one needs to incorporate richer local properties, as well as strong machine learning techniques, like efficient graph cuts [22]. For fine-grained recognition, in [40], a joint object detection and segmentation framework is introduced to localize objects. In [12], Chai *et al.* make use of region-level cues for discriminative co-segmentation on multiple images. Here, we opt for segmentation object proposals, as they allow for a more precise delineation of the object, thus enabling a better learning of attribute representations.

Once having object proposals, we need to classify them against a pre-defined set of object classes. For object detection [32] and semantic segmentation [3, 8], employing non-linear kernel machines [9] and state-of-the-art feature encodings, such as second-order poolers [8], Fisher vectors [26, 29] or deep learning features [21] has shown to yield excellent results. In this work our goal is to be able to perform (zero-shot) object classification *and* segmentation. Therefore, we depart from the above works and proceed with the classification and segmentation of objects solely on the basis of attributes, as the use of low-level features [3, 8, 9, 21, 26] would prohibit us from detecting the unseen classes.

Label and Attribute Embedding. As attributes were originally designed for describing objects, the learned attributes are not necessarily optimal for classifying (novel) objects. For this reason Akata *et al.* [1] propose to embed class label in the space of attributes and use the WSABIE [39] learning criterion, adapted for attribute learning. This method optimizes the attribute learning directly for object classification instead of attribute prediction. Our major difference with [1] is that we look for the best possible segment, while predicting the label of an unseen image. The segments, which we search, are integrated as latent variables in our empirical risk function. From a theoretical standpoint, instead of considering a fixed margin equal to 1, we minimize our max-margin empirical risk over both the class label as well as the segmentation quality. This learning methodology allows for learning of high quality segments.

Efficient Region Computations. As semantic segmentation and object detection entail a great number of free variables, direct optimization of attribute models on a per segment basis results in a severe computational bottleneck. For this reason there have been several methodologies proposed in the literature for efficiently computing classification scores from multiple image regions. In [24] Lampert *et al.* employ a branch and bound optimization scheme for visiting several thousands of bounding box locations efficiently. Relaxing the constraints for a bounding box geometry, Vijayanarasimhan and Grauman [35] propose a similar optimization scheme for arbitrary, free form regions in an image. Yet, both these methods do not consider any efficient normalization of the representations, as that would render their methods highly inefficient. For this reasons Li *et al.* [26] propose codemaps, which allows for efficient, accurate and normalized region-level representations by reordering the encoding, pooling and classification steps over superpixels. In this work we make use of the codemaps framework.

Structured SVMs and Latent SVMs. In the current work the main focus is multi-class classification, while using the best object segment proposal for each class. Since the segment proposals are not explicitly evaluated, we treat them as latent variables of our model, a formulation that resembles latent SVM [18]. Moreover, we use margin rescaling of structured SVMs [31] to include a penalty for segment proposals with a low-overlap with the ground-truth segment. The penalty function is based on the intersection over union criterion, also used in structured output regression [5, 9].

Different from structured output regression our final objective is (zero-shot) multi-class classification, and not a structured output containing the best segment for each class label. Hence, our structured loss is built around mid-level attribute representations, which also need to be optimized for. For this reason we follow [1, 39] and employ an embedding function with a ranking objective instead of a multi-class SVM objective function. In this model the latent segment variables help to learn better attributes and attributes help to improve segmentation.

3 Object-Level Attributes

Given an image x, our classification function f is defined as follows:

$$f(x) = \arg\max_{y \in \mathcal{Y}} \max_{z \in Z(x)} F(z, y), \tag{1}$$

where z is a latent variable, $Z(x)$ indicates a set of segment proposals for image x, and $F(z, y)$ a compatibility function between segment z and label y. Intuitively, this function first finds the best scoring segment $z \in Z(x)$, for each class y. Then, the class $y \in \mathcal{Y}$ with the highest score is returned as a prediction. Note that we do not assume the object bounding box or object segmentation is known at prediction time. Instead the object segmentation is inferred as a latent variable given the image.

We describe attribute embedding in Section 3.1 and our learning objective in Section 3.2. In Section 3.3 we discuss how segment proposals are obtained and how Eq. 1 can be evaluated efficiently, using codemaps [26].

3.1 Attribute Embedding

We follow the label and attribute embedding approaches from [1,39], where each class label y is embedded in the m-dimensional space of attributes by $\phi(y) \in \mathbb{R}^m$. While [1, 39] embed the full image features, we embed the visual features of a segment z only. Let $\theta(z) \in \mathbb{R}^d$ be the embedding of segment z yielding a d-dimensional feature vector. In this work we use the state-of-the-art Fisher vector framework [29] for this visual embedding.

In our model, $F(z, y)$ measures the compatibility between segment z and the embedding of class y. This compatibility function is defined as:

$$F(z, y; W, \phi) = \theta(z)' W \phi(y), \tag{2}$$

where $W \in \mathbb{R}^{d \times m}$ is the model parameter matrix, which we need to learn.

We stack the attribute embeddings of each class $\phi(y)$ into an embedding matrix Φ for all classes. In the fully supervised setting, where visual examples for each class are provided, we also learn the class-to-attribute embedding Φ, similar to WSABIE [39]. In the case of zero-shot classification, we use the fixed attribute-to-class mapping $\Phi = \Phi^{\mathcal{A}}$, which resembles ALE [1].

3.2 Learning

For training we assume a collection of images $\{(x_i, y_i, z_i)\}_{i=1}^{N}$, in which each image x_i has a ground truth label y_i and a ground truth object segment z_i. Furthermore, we assume that there exists a mapping from attributes to classes $\Phi^{\mathcal{A}}$, which defines the relevant attributes for each class. The goal is to learn the model parameters W and the mapping Φ to minimize the prediction errors, while selecting object segments of better quality. For learning we employ structured risk minimization, using a ranking objective built upon [1,5,39].

The loss function of a ground-truth image/label/segment triplet (x_i, y_i, z_i) for a prediction label y, is defined as:

$$\ell(y, z_i, y_i, x_i) = \max_{z \in Z(x_i)} \Delta(z, y, y_i, z_i) + F(z, y) - F(z_i, y_i). \tag{3}$$

The Δ function, which determines the margin, is defined as:

$$\Delta(z, y, z_i, y_i) = \begin{cases} 1 - O(z, z_i) & \text{if } y = y_i, \\ 1 & \text{otherwise,} \end{cases} \tag{4}$$

where $O(z, z_i)$ is the intersection over union between the selected segment z and the ground-truth segment z_i, similar to [5]. This margin re-scaling function enforces a margin of 1 if the label y and y_i do not match. When the labels do match, the margin is determined by the area of overlap between the segment z and the ground-truth segment z_i.

The following objective is used as the data term in the empirical risk:

$$R(W, \Phi) = \frac{1}{N} \sum_{i=1}^{N} \gamma(k_i) \sum_{y \in \mathcal{Y}} \left[\ell(y, z_i, y_i, x_i)\right]_+ , \tag{5}$$

where k_i is an upper-bound on the rank of the correct label, γ transforms this rank into a weight, and where $[\cdot]_+ = \max(0, \cdot)$. The upper-bound on the rank is computed as the number of loss-generating labels:

$$k_i = \sum_{y \in \mathcal{Y}} [\![\ell(y, z_i, y_i, x_i) > 0]\!] \tag{6}$$

where, we use Iversons bracket notation to denote $[\![\cdot]\!] = 1$ if the condition is true, and 0 otherwise. Following [34], we define the rank to weight function as $\gamma(k) = \frac{1}{k} \sum_{j=1}^{k} \alpha_j$, using $\alpha_j = \frac{1}{j}$.

When applied on the entire image, *i.e.,* when Eq. 3 is defined as $\ell(y, y_i, x_i) = \Delta(y, y_i) + F(x_i, y) - F(x_i, y_i)$, our objective function is identical to WSABIE/ALE.

Fully Supervised Learning. In the fully supervised case, where we have visual examples from all classes, we minimize the following regularized risk objective:

$$\min_{W, \Phi} \frac{\lambda}{2} ||W||^2 + \frac{\mu}{2} ||\Phi - \Phi^{\mathcal{A}}||^2 + R(W, \Phi), \tag{7}$$

where λ and μ are trade-off parameters between the data term and the regularization, which we set using cross-validation. Regularizing towards the pre-defined class-to-attribute encoding $(\Phi - \Phi^{\mathcal{A}})$ allows us to exploit this high-level semantic prior. This could be particularly beneficial in the case when just a few examples per class are available.

Zero-Shot Classification. In the setting of zero-shot classification, visual training examples are given only for a subset of the classes, while evaluation is performed on a disjoint set of the classes. In this case the attribute embedding is fixed to the existing mapping $\Phi = \Phi^{\mathcal{A}}$, and Eq. 7 reduces to:

$$\min_{W} \frac{\lambda}{2} ||W||^2 + R(W, \Phi^{\mathcal{A}}), \tag{8}$$

where λ is a trade-off parameter, which we set using cross-validation using a hold-out set of images from the known train classes.

3.3 Efficient Maximization

The main computational challenge of our method is efficiently solving Eq. 1 and Eq. 3 during training and evaluation. The number of possible segmentations in an image grows exponentially with the size of an image. To solve this problem efficiently we follow [26].

For each image, we start by extracting a set of superpixels S from an image, typically using $|S| \approx 500$. We then use a segment proposal algorithm, the off the shelve CPMC-algorithm [10], to obtain a set of approximately $1,000$ segments $Z(x)$. Even so, the maximization in Eq. 1 and Eq. 3 over such a large set of segments during training and evaluation remains expensive, especially for the high-dimensional visual embeddings we are using.

Since the visual embeddings are based on a sum-pooling operator of local image features, and since F (Eq. 2) is comprised of two linear components W and ϕ, we have:

$$\max_{z \in Z(x)} F(z, y) = \max_{z \in Z(x)} \sum_{s \in S(z)} F(s, y) = \max_{z \in Z(x)} \frac{1}{L_z} \sum_{s \in S(z)} \theta(s)' W \phi(y), \tag{9}$$

where $S(z)$ correspond to the set of superpixels in segment z and L_z is the ℓ_2-norm of the feature embedding of z. The decomposition over superpixels allows for on-the-fly calculations of the feature embeddings for all segments. For a given W and ϕ, the compatibility function $F(s, y)$ can be precomputed, and the maximization over segments boils down to just look-ups and summations.

At training time, the computational efficiency comes at the cost of higher memory requirements, as we need to maintain the feature embeddings for all superpixels per image. However, at test time both the computational and memory complexity are very low.

4 Experiments

4.1 Datasets and Experimental Setup

CUB-2011 Birds. We conduct our main experiments on the Caltech UCSD Birds 2011 dataset [37], as it fulfills three requirements. First, this dataset contains an extensive array of object categories that are visually difficult to distinguish. Second, the CUB-2011 dataset enjoys a detailed annotation of 312 human-understandable attributes, *e.g.,* whether a bird has a *striped wing* or a *curved beak.* Last, this dataset contains localization information in the form of segmentation masks. For the CUB-2011 dataset we use the standard training and test splits, without mirroring the training images. We use the provided segmentations only during training, unless stated otherwise. To obtain a mapping from attributes to classes we binarize the continuous attributes provided in the dataset; attributes are considered relevant for a class if their confidence is above the average confidence value of that attribute.

Butterflies. As a second dataset, we use a modification of the Leeds Butterfly Dataset [38]. This is a fine-grained multi-class dataset containing images and segmentation masks of ten butterfly species. We automatically transform the provided textual descriptions into a set of 20 attributes, mostly describing the color patterns of the butterflies, and automatically generate an attribute-to-class mapping. We also obtain the ground truth bounding boxes by automatically enclosing a bounding box around each segmentation mask. The dataset contains 620 images for training and 212 for testing.

a-Pascal++. As a third dataset, we use a modification of the a-Pascal [17] dataset, which we coin a-Pascal++. Our dataset combines the 64 attributes annotated for the a-Pascal dataset with the segmentation masks from the VOC Pascal Challenge [16]. Since the focus of our work is not segmentation inference, but segmentation-based classification, we select the images containing a single object. This makes a-Pascal++ a multi-class dataset with 20 classes, 64 attributes and segmentation masks, we use 1,429 images for training and 203 for testing. Note that the original a-Pascal dataset was used for describing attributes given objects [17], the bounding box object locations were available both during training and testing. In our work, we provide segmentation masks only at train time, at test time the object-level attributes are inferred.

Visual features. For the visual representation we follow the Fisher vector framework [29], computed on dense RGB-SIFT features [30] extracted every 2 pixels and at multiple scales, and projected to 80 dimensions using PCA. We experiment with different codebook sizes, and indicate the number of mixture components k used with each experiment. For the full image representation we use the Fisher vector with power-normalization and ℓ_2-normalization [29], while for the segment representation we use the ℓ_2-normalized Fisher codemaps [26]. For obtaining object segment proposals we use the off-the-shelf CPMC [10], which we approximate with the superpixels from [4].

Training. To train our models, we rely on stochastic gradient descent of the objective functions Eq. 7 and Eq. 8. We validate the regularization parameters and number of iterations on a subset of the train set, and re-train using these parameters on the whole train set. All experiments using the full image embedding are trained using the WSABIE/ALE objective, which equals to Eq. 7 and Eq. 8, when the full image is the only segment of an image.

Evaluation. We use three measures for evaluation. First, we use the *mean class accuracy* (MCA), where for each class the top-1 accuracy is computed and averaged over all classes. Second, we use the *mean class accuracy over correctly segmented objects* (MSO). MSO is computed similar to MCA, except that a prediction is considered correct only if both the label is correct and the overlap of the latent segment with the ground-truth segmentation meets the Pascal VOC criterion. This criterion requires that the intersection over the union (IoU) of the two segments is greater than 50%. In a similar vein, we use the *average overlap* (AO) to evaluate the quality of the inferred latent segments, disregarding the class label prediction.

4.2 Object-Level Attributes on Oracle Segments

In the first experiment we want to establish whether attributes on object segments are beneficial for object classification. To this end we design an oracle experiment, where we compare a full image feature embedding to using an embedding of visual features describing an oracle provided bounding box or segment. For all the three datasets, the oracle provides perfect object bounding boxes and segmentation masks, both during training and testing. The obtained accuracy on this experiment will not reflect reality, however it provides insight in the effectiveness of our proposed object-level attributes. We compare the Fisher vector embedding of the full image to the Fisher vector embedding of the oracle bounding box/segment, which we train with the ALE framework [1]. We present the aggregated results in Table 1.

As a preliminary, we note that our 40K Fisher vector (k=256) performs on par with the 64K Fisher vector used in [1], where 20.5% MCA is reported when evaluating ALE on the CUB-2011 Birds dataset. We observe that by using oracle object segments we obtain up to an absolute 31.5% accuracy increase. This improvement seems to be consistent over different datasets, each depicting different characteristics in the number of images, number of classes and visual relatedness among the classes. It is interesting to note that the improvement in accuracy is consistent across all categories (data not shown), and for various numbers of mixture components. Using oracle segments is also consistently better than using oracle bounding boxes, since bounding boxes inevitably include some background which may not depict the objects and thus the attributes. Having obtained evidence that extracting attributes on object-level helps, we proceed with the latent segments.

Table 1. Object-level attributes on oracle segments. We compare the performance of ALE [1] using full image embeddings with oracle bounding box/segment embeddings. By using oracle object-level attributes the classification accuracy increases substantially. Although these numbers are only theoretical, they serve as an upper bound of the classification accuracy that we may obtain.

Dataset	**Codebook**	**Entire image**	**Oracle bbox**	**Oracle segment**
		MCA	MCA	MCA
CUB-2011	$k = 16$	13.8	25.8	43.9
	$k = 256$	21.4	36.4	52.9
Butterflies	$k = 16$	83.8	96.9	99.1
a-Pascal++	$k = 256$	30.6	33.6	40.2

Table 2. Object-level attributes on latent segments. Object-level attributes optimized with our joint learning are up to around 4-21% more accurate than computing attributes on the full images. Note that for a larger number of mixture components accurate prediction also entails accurate segmentation.

Dataset	**Codebook**	**Entire image**	**Object-level attributes**		
		MCA	MCA	MSO	AO
CUB-2011	$k = 16$	13.8	35.2	29.9	60.5
	$k = 256$	21.4	39.2	35.5	66.3
Butterflies	$k = 16$	83.8	96.4	95.5	84.6
a-Pascal++	$k = 256$	30.6	35.0	24.7	48.2

4.3 Object-Level Attributes on Latent Segments

In the second experiment we evaluate the ability to infer the object segment as a latent variable in the model and to classify the segmented objects using attributes. During testing of a given image, our model is able to simultaneously predict the most likely label for the object and its respective segmentation mask.

We present the aggregated results in Table 2. We observe that our joint learning returns highly accurate results, as we improve over the full image results of [1] by around 4-21%. Note that the accuracy of the joint learning is reasonably close to the accuracy when using oracle segments, indicating that the returned segmentations are quite accurate. Moreover, we observe that for a larger codebook the discrepancy between accurate prediction and accurate prediction with accurate segmentation is smaller. Hence, a larger codebook is able to better suppress the background signal, thus returning simultaneously both accurate classifications and segmentations.

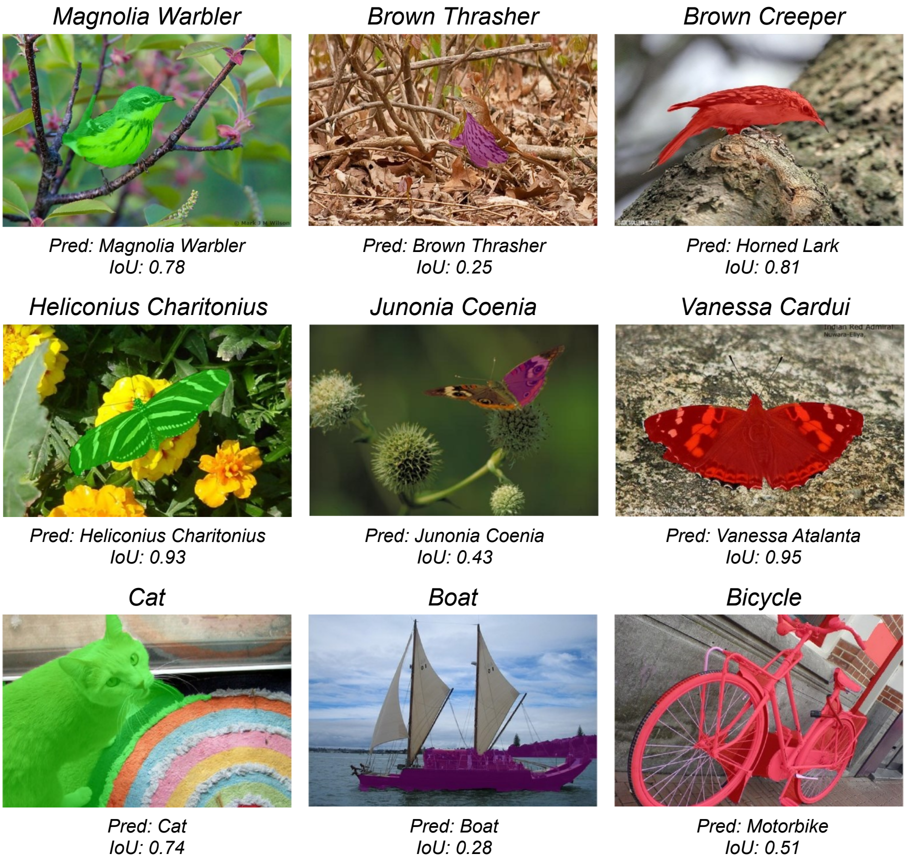
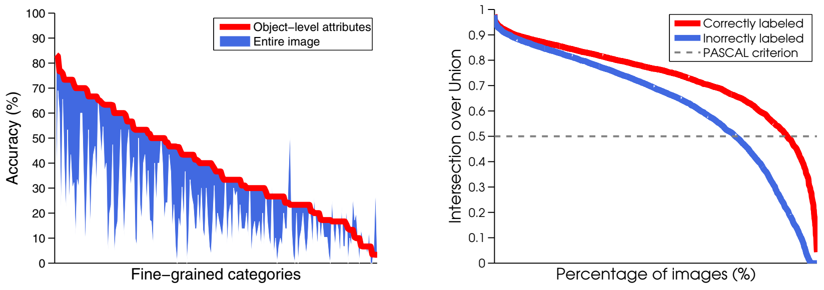


Fig. 2. Example classification and segmentation results using object-level attributes on latent segments. The segmentation masks are green-colored for correct label prediction and red-colored for wrong label prediction. The purple-colored ones indciate correct label predcition, but with low segmentation accuracy (IoU<0.5). It is noteworthy that even if the objects are labeled incorrectly, the selected segmentation masks are often very accurate.

In Fig. 2 we provide some illustrative examples of our classification and segmentation results. We make three observations. First, the predicted segments look in general of high quality. Second, even if the object segmentation does not meet the Pascal criterion, it often contains sufficient class specific information, for example the second column in Fig. 2. Third, even in the case when the predicted label is incorrect the segmentation is still focused on the object, see the third column in Fig. 2.

In Fig. 3a we present a comparison of the individual class accuracies between two methods. We observe that learning object-level attributes on latent segments brings a consistent improvement to almost all the classes. Last, we



(a)

(b)

Fig. 3. Object-level attributes on latent segments. (a) Object-level attributes are consistently better than learning attributes on the entire image for almost all the classes. (b) The joint learning discovers segments that meet the PASCAL criterion for about 91% of the correctly labeled images and for about 75% of the incorrectly labeled images. Results computed on *CUB-2011* with k=256.

illustrate in Fig. 3b the quality of segmentations for both the correctly and incorrectly predicted objects. We observe that for the vast majority the quality of segmentations is quite high. We discover segments that meet the PASCAL criterion for about 91% of the correctly labelled images and for about 75% of the incorrectly labelled images. Therefore, our joint learning allows for a precise localization of objects.

Comparison with part-localized attributes. To compare our approach with a recent part-localized attribute model, we also conduct an experiment on a subset of CUB-2011: five categories consisting of different species of warblers. We follow the same experimental protocol as [14]. Our model of learning object-level attributes on latent segments scores 65.8% accuracy using a codebook of mixture components k=16, using full image embedding scores 42.2%, while the localized attribute model [14] reports ~55%.

We conclude that the joint learning of object-level attributes with a segmentation model leads to accurate attribute-based classification.

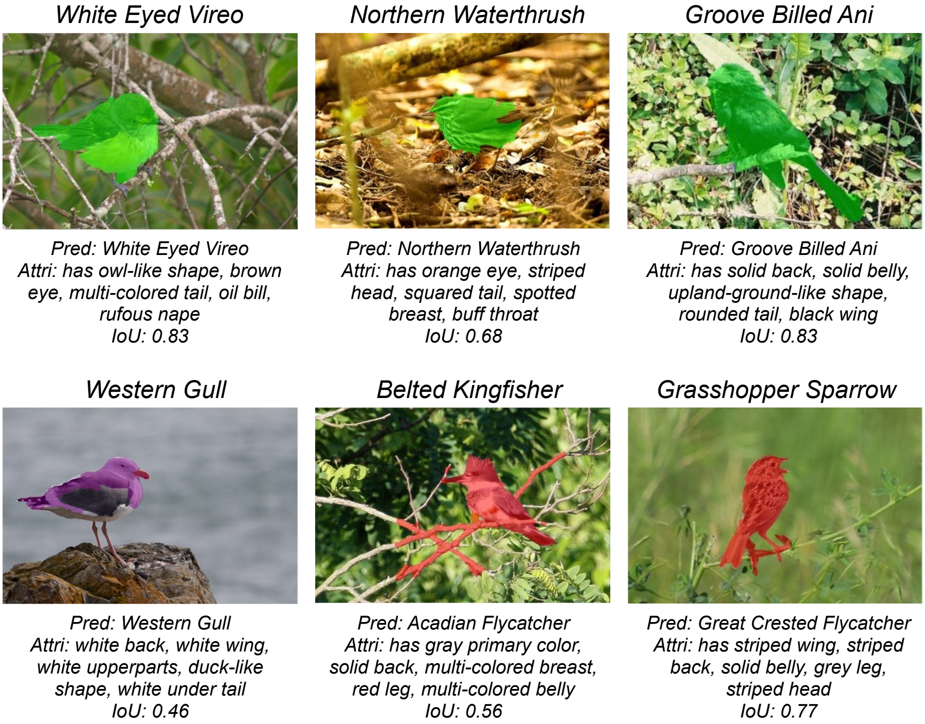
4.4 Object-Level Attributes for Zero-Shot

In the third experiment we take advantage of the fact that our object-level attributes can be shared among classes. As a result, assuming that one is provided with an attribute-to-class mapping, one can perform zero-shot classification, which allows for simultaneous classification and segmentation of the object of interest. We experiment on the CUB-2011 dataset, using the same 150 train classes and 50 test classes as in [1].

We present the numerical results in Table 3. We observe that our approach improves the zero-shot classification accuracy, while returning the location of objects that belong to classes we have not seen before. For fair comparison with

Table 3. Object-level attributes for zero-shot classification on CUB-2011. For fair comparison with the results of the supervised experiment, we report the accuracy of the supervised model also for the 50 classes that we used for testing the zero-shot model. The joint learning of attributes is able to not only improve the zero-shot classification accuracy, but also return the location of objects that belong to previously unseen classes.

Setting	Codebook	Entire image	Object-level attributes		
		MCA	MCA	MSO	AO
Supervised	$k = 16$	27.1	51.5	43.0	61.8
Zero-shot	$k = 16$	11.3	15.7	12.4	56.3



White Eyed Vireo

Pred: White Eyed Vireo
Attri: has owl-like shape, brown eye, multi-colored tail, oil bill, rufous nape
IoU: 0.83

Northern Waterthrush

Pred: Northern Waterthrush
Attri: has orange eye, striped head, squared tail, spotted breast, buff throat
IoU: 0.68

Groove Billed Ani

Pred: Groove Billed Ani
Attri: has solid back, solid belly, upland-ground-like shape, rounded tail, black wing
IoU: 0.83

Western Gull

Pred: Western Gull
Attri: white back, white wing, white upperparts, duck-like shape, white under tail
IoU: 0.46

Belted Kingfisher

Pred: Acadian Flycatcher
Attri: has gray primary color, solid back, multi-colored breast, red leg, multi-colored belly
IoU: 0.56

Grasshopper Sparrow

Pred: Great Crested Flycatcher
Attri: has striped wing, striped back, solid belly, grey leg, striped head
IoU: 0.77

Fig. 4. Example results using object-level attributes for zero-shot classification and segmentation. Note that although we have not seen examples of these classes, the label predictions and the segmentations are reasonable.

the results of the supervised experiment, we report the accuracy of the supervised model also for the 50 classes that we used for evaluating the zero-shot model.

In Fig. 4 we present some visual examples of zero-shot classification and segmentation, together with the highest scoring attributes for the respective images, found after computing the contribution of each attribute to the final classification score of each respective image. Although at training time there were no examples of the classes on which we test, we are able to obtain satisfactory classifications and segmentations. Interestingly, the most important attributes seem relevant, even for the misclassified birds. For example, although a *Grasshopper Sparrow* was wrongly labeled as a *Great Crested Flycatcher*, the most important attributes fit to the image, namely the bird has a *striped wing*, a *striped back* and a *solid belly*.

We conclude that object-level attributes also make sense for zero-shot.

5 Conclusions

In this paper we revisit attribute-based representations, approaching them from the perspective of locality. To this end we have introduced object-level attributes, which are trained on segmented images with attribute descriptions. At test time, the object segmentation is treated as a latent variable, which is inferred. As part of our first contribution, we make the observation that attributes usually refer to visual properties of object classes not of an object instance, *e.g.,* whether a bird has a *curved beak* or an airplane has a *jet engine*. Using oracle object-level attributes we have experimentally shown on three different datasets that, indeed, localizing attributes leads to an impressive increase in accuracy. As a second contribution, we have proposed a joint learning framework, learning attribute embeddings while improving object segmentations using a max-margin ranking objective. The experimental results show that our learning framework yields classification accuracies which are two- to three-fold better in attribute-based classification, compared to using full image features. Moreover, we can also return high quality segmentations. Finally, we have applied object-level attributes to the task of zero-shot classification on the CUB-2011 bird dataset. In this setting, we infer class predictions and segmentation masks from bird classes for which no training data was available. The experimental results show that object-level attributes, also in the zero-shot setting, improve accuracy significantly. We therefore conclude that attributes make sense on segmented objects.

Acknowledgments. This research is supported by the STW STORY project and the Dutch national program COMMIT.

References

1. Akata, Z., Perronnin, F., Harchaoui, Z., Schmid, C.: Label-embedding for attribute-based classification. In: CVPR (2013)
2. Alexe, B., Deselaers, T., Ferrari, V.: Measuring the objectness of image windows. TPAMI (2012)
3. Arbelaez, P., Hariharan, B., Gu, C., Gupta, S., Bourdev, L., Malik, J.: Semantic segmentation using regions and parts. In: CVPR (2012)
4. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: From contours to regions: An empirical evaluation. In: CVPR (2009)

5. Blaschko, M.B., Lampert, C.H.: Learning to localize objects with structured output regression. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 2–15. Springer, Heidelberg (2008)
6. Bourdev, L., Maji, S., Malik, J.: Describing people: A poselet-based approach to attribute classification. In: ICCV (2011)
7. Branson, S., Wah, C., Schroff, F., Babenko, B., Welinder, P., Perona, P., Belongie, S.: Visual recognition with humans in the loop. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part IV. LNCS, vol. 6314, pp. 438–451. Springer, Heidelberg (2010)
8. Carreira, J., Caseiro, R., Batista, J., Sminchisescu, C.: Semantic segmentation with second-order pooling. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part VII. LNCS, vol. 7578, pp. 430–443. Springer, Heidelberg (2012)
9. Carreira, J., Li, F., Sminchisescu, C.: Object recognition by sequential figure-ground ranking. IJCV (2012)
10. Carreira, J., Sminchisescu, C.: CPMC: Automatic object segmentation using constrained parametric min-cuts. TPAMI (2012)
11. Chai, Y., Lempitsky, V., Zisserman, A.: BiCoS: A bi-level co-segmentation method for image classification. In: ICCV (2011)
12. Chai, Y., Rahtu, E., Lempitsky, V., Van Gool, L., Zisserman, A.: TriCoS: A tri-level class-discriminative co-segmentation method for image classification. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part I. LNCS, vol. 7572, pp. 794–807. Springer, Heidelberg (2012)
13. Deng, J., Krause, J., Fei-Fei, L.: Fine-grained crowdsourcing for fine-grained recognition. In: CVPR (2013)
14. Duan, K., Parikh, D., Crandall, D., Grauman, K.: Discovering localized attributes for fine-grained recognition. In: CVPR (2012)
15. Endres, I., Hoiem, D.: Category-independent object proposals with diverse ranking. TPAMI (2014)
16. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. IJCV (2010)
17. Farhadi, A., Endres, I., Hoiem, D., Forsyth, D.: Describing objects by their attributes. In: CVPR (2009)
18. Felzenszwalb, P., Girshick, R., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part based models. TPAMI (2010)
19. Ferrari, V., Zisserman, A.: Learning visual attributes. In: NIPS (2007)
20. Gavves, E., Fernando, B., Snoek, C., Smeulders, A., Tuytelaars, T.: Fine-grained categorization by alignments. In: ICCV (2013)
21. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: CVPR (2014)
22. Kolmogorov, V., Zabin, R.: What energy functions can be minimized via graph cuts? TPAMI (2004)
23. Kovashka, A., Grauman, K.: Attribute adaptation for personalized image search. In: ICCV (2013)
24. Lampert, C., Blaschko, M., Hofmann, T.: Efficient subwindow search: A branch and bound framework for object localization. TPAMI (2009)
25. Lampert, C., Nickisch, H., Harmeling, S.: Attribute-based transfer learning for object categorization with zero/one training example. TPAMI (2013)
26. Li, Z., Gavves, E., van de Sande, K., Snoek, C., Smeulders, A.: Codemaps segment, classify and search objects locally. In: ICCV (2013)

27. Manen, S., Guillaumin, M., Van Gool, L.: Prime object proposals with randomized prim's algorithm. In: ICCV (2013)
28. Parikh, D., Grauman, K.: Relative attributes. In: ICCV (2011)
29. Sánchez, J., Perronnin, F., Mensink, T., Verbeek, J.: Image classification with the fisher vector: Theory and practice. IJCV (2013)
30. van de Sande, K., Gevers, T., Snoek, C.: Evaluating color descriptors for object and scene recognition. TPAMI (2010)
31. Tsochantaridis, I., Joachims, T., Hofmann, T., Altun, Y.: Large margin methods for structured and interdependent output variables (2005)
32. Uijlings, J., van de Sande, K., Gevers, T., Smeulders, A.: Selective search for object recognition. IJCV (2013)
33. Uijlings, J., Smeulders, A., Scha, R.: The visual extent of an object. IJCV (2012)
34. Usunier, N., Buffoni, D., Gallinar, P.: Ranking with ordered weighted pairwise classification. In: ICML (2009)
35. Vijayanarasimhan, S., Grauman, K.: Efficient region search for object detection. In: CVPR (2011)
36. Wah, C., Branson, S., Perona, P., Belongie, S.: Multiclass recognition and part localization with humans in the loop. In: ICCV (2011)
37. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The Caltech-UCSD Birds-200-2011 Dataset. Tech. rep. (2011)
38. Wang, J., Markert, K., Everingham, M.: Learning models for object recognition from natural language descriptions. In: BMVC (2009)
39. Weston, J., Bengio, S., Usunier, N.: WSABIE: Scaling up to large vocabulary image annotation. In: IJCAI (2011)
40. Zhu, S., Angelova, A.: Efficient object detection and segmentation for fine-grained recognition. In: CVPR (2013)