No Spare Parts: Sharing Part Detectors for Image Categorization

Mettes, P.; van Gemert, J.C.; Snoek, C.G.M.

DOI
10.1016/j.cviu.2016.07.008

Publication date
2016

Document Version
Final published version

Published in
Computer Vision and Image Understanding

License
Article 25fa Dutch Copyright Act (https://www.openaccess.nl/en/in-the-netherlands/you-share-we-take-care)

Link to publication

Citation for published version (APA):

General rights
It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations
If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: https://uba.uva.nl/en/contact, or a letter to: Library of the University of Amsterdam, Secretariat, Singel 426, 1012 WP Amsterdam, The Netherlands. You will be contacted as soon as possible.

UvA-DARE is a service provided by the library of the University of Amsterdam (https://dare.uva.nl)
No spare parts: Sharing part detectors for image categorization

Pascal Mettes\textsuperscript{a,∗}, Jan C. van Gemert\textsuperscript{a,b}, Cees G.M. Snoek\textsuperscript{a,c}

\textsuperscript{a}University of Amsterdam, Science Park 904, Amsterdam, The Netherlands
\textsuperscript{b}Delft University of Technology, Meelkerweg 4, Delft, The Netherlands
\textsuperscript{c}Qualcomm Research Netherlands, Science Park 400, Amsterdam, The Netherlands

A R T I C L E   I N F O

Article history:
Received 9 October 2015
Revised 8 July 2016
Accepted 26 July 2016
Available online 10 August 2016

Keywords:
Image categorization
Discriminative parts
Part sharing

A B S T R A C T

This work aims for image categorization by learning a representation of discriminative parts. Different from most existing part-based methods, we argue that parts are naturally shared between image categories and should be modeled as such. We motivate our approach with a quantitative and qualitative analysis by backtracking where selected parts come from. Our analysis shows that in addition to the category parts defining the category, the parts coming from the background context and parts from other image categories improve categorization performance. Part selection should not be done separately for each category, but instead be shared and optimized over all categories. To incorporate part sharing between categories, we present an algorithm based on AdaBoost to optimize part sharing and selection, as well as fusion with the global image representation. With a single algorithm and without the need for task-specific optimization, we achieve results competitive to the state-of-the-art on object, scene, and action categories, further improving over deep convolutional neural networks and alternative part representations.

© 2016 Elsevier Inc. All rights reserved.

1. Introduction

In this work, we aim to categorize images into their object, scene, and action category. Image categorization has been studied for decades and tremendous progress has been made (Lazebnik et al., 2006; Sánchez et al., 2013; Sivic and Zisserman, 2003; van de Sande et al., 2010; van Gemert et al., 2010; Zhang et al., 2007), most recently by the introduction of very deep convolutional neural networks (Krizhevsky et al., 2012; Simonyan and Zisserman, 2015; Szegedy et al., 2015; Zeiler and Fergus, 2014). These networks learn to categorize images from examples by backpropagating errors through stacked layers of convolutional filters pooled over image regions. Despite their implicit local nature, deep nets result in a global scene representation only. Thereby, ignoring known benefits of explicitly encoding local image blocks, i.e. discriminative power (Doersch et al., 2013; Zuo et al., 2014), image interpretation (Doersch et al., 2012; Freytag et al., 2014), and complementarity (Juneja et al., 2013; Zuo et al., 2014). In this paper we make a case for sharing parts for image categorization, studying what parts to consider, which parts to select and how to share them between categories.

The notion of sharing has been well studied in data mining (Rematas et al., 2015; Yuan et al., 2008). These works repeatedly show that sharing boosts classification performance and provides connections between co-occurring elements. Motivated by these examples, we investigate sharing of parts when learning part-representations for image categorization.

Consider Fig. 1, where examples of image categories sofa and horse utilize parts from their own category as well as parts from other categories and the background context. When a classifier is trained exclusively using parts from its own category, relevant information is missed (Azizpour et al., 2015; Parizi et al., 2015). As illustrated in Fig. 1, for object categories such as sofa, it is informative to use parts from cat and dog categories as well as parts from sofa itself (Azizpour et al., 2015; Parizi et al., 2015). By giving the sofa classifier access to dog and cat training images, the recognition of sofa is improved even though these images may not contain a sofa at all. Even when global image categories differ, their representation can share similar parts and should thus be modeled as such (Juneja et al., 2013; Parizi et al., 2015).

To obtain insight in part sharing, we track where a part comes from. We define three types of part-origin: Own: parts coming from the defining image category; Other: parts coming from other labeled categories; Context: parts not overlapping with labeled categories. Our analysis shows that the global image representation captures the scene; that Own parts benefit out-of-context and small images (e.g. sofa in the grass); Other parts aid for co-occurring categories (e.g. person on a horse) and Context parts...
help to recognize supporting structures (e.g. fence for sheep or cows).

Several recent works have focused on local parts as representation for image classification (Bossard et al., 2014; Doersch et al., 2013; 2012; Juneja et al., 2013) by performing part selection for each category separately. These works follow a common four-step pipeline. First, a set of potentially interesting parts is proposed. Second, a selection algorithm discovers the most discriminative parts. Third, classification is performed on the part-based representations. And fourth, fusion with a global image representation. These four steps have shown to yield robust and complementary image representations (Bossard et al., 2014; Doersch et al., 2013; Zuo et al., 2015). However, by separating the part selection, category classification, and fusion steps into disjoint steps these works leave room for a number of improvements. Most notably, (1) the models do not share parts and perform part selection for each category independently, as also observed by Parizi et al. (2015), (2) the models use different objective functions for the part selection step and the final classification step, and (3) the models perform part selection independently of the global representation with which they are fused. As we will show, these drawbacks result in sub-optimal image categorization results.

We make three contributions in this paper: (i) we establish that three part types are relevant for image categorization, which are all naturally shared between categories when learning a part representation for image categorization; (ii) we present an algorithm for part selection, sharing, and image categorization based on boosting. It embeds all three part types, without the need for explicit part type definition; (iii) we introduce a fusion technique for combining part-based with global image representations. We report results competitive to the state-of-the-art on object, scene, action, and fine-grained categorization challenges, further improving over the very deep convolutional neural networks of Simonyan and Zisserman (2015); Szegedy et al. (2015).

The rest of the paper is outlined as follows. In Section 2, we describe related work, while Section 3 outlines our proposed method. This is followed by the experimental evaluation in Section 4. We draw conclusions in Section 5.

2. Related work

The state-of-the-art in image categorization relies on deep convolutional neural networks (ConvNets) (Krizhevsky et al., 2012; Simonyan and Zisserman, 2015; Szegedy et al., 2015; Zeiler and Fergus, 2014). These networks learn image feature representations in the form of convolution filters. Locality is implicitly incorporated by stacking and pooling local filter responses, resulting in increasingly larger filter responses, cumulating in a single global image representation. Two recent network architectures, the VGG network of Simonyan and Zisserman (2015) and the GoogleNet network of Szegedy et al. (2015), have shown that further increasing the network depth results in state-of-the-art performance on a wide range of image categorization datasets (Simonyan and Zisserman, 2015; Szegedy et al., 2015). A global representation from such networks benefits from augmenting it by aggregating local ConvNets features of densely sampled image parts in a VLAD (Gong et al., 2014) or Fisher Vector (Cimpoi et al., 2015; Yoo et al., 2015) representation. We follow such approaches and augment the global representation by using ConvNets to represent discriminative local parts.

An excellent source of part-based information for classifying object images is the response of a detector for the object at hand. In the pioneering work of Harzallah et al. (2009), it is shown that an object detector improves image classification, and vice versa (Divvala et al., 2009; Sadeghi and Farhadi, 2011). Others improved upon this idea by adding context (Song et al., 2011), and deep learning (Oquab et al., 2014). However, when no bounding box annotations are available, a supervised detector cannot be trained. In our method, we do not use any bounding box annotations and rely exclusively on the global image label.

In the absence of bounding box annotations, one may focus on automatically discovering discriminative parts in images. The work of Singh et al. (2012) proposes an unsupervised method for finding parts, by iterative sampling and clustering large collections of HOG features, SVM training on the clusters, and assigning new top members to each cluster. Other part-based methods follow a supervised approach, e.g. using discriminative mode seeking (Doersch et al., 2013), random forests (Bossard et al., 2014), average max precision gains (Endres et al., 2013), or group sparsity (Sun and Ponce, 2013) for discovering the best parts for each category separately from image-level labels. Recent work moves away from HOG features in favor of local ConvNet activations for part selection (Li et al., 2015), based on association rule mining per category. In this work, we also leverage image-level labels for part selection. In contrast to Bossard et al. (2014); Doersch et al. (2013); 2012), we also perform part selection by sharing over all
categories, rather than performing selection per category. Furthermore, we optimize the part selection with the global image representation during fusion.

The work by Juneja et al. (2013) shows that using image-level labels leads to better categorization performance using fewer parts than the unsupervised method of Singh et al. (2012). Their method selects parts that have the lowest entropy among categories. In effect, this limits the sharing over a few categories only. We strive to share parts as much as possible. Rather than relying on entropy for the selection, we prefer a boosting objective that optimizes sharing for all categories simultaneously.

The seminal work of Torralba et al. (2007) has previously explored the idea of sharing among categories. We follow this line of work in a part-based setting. Similar to Torralba et al. (2007), we opt for a boosting approach, as we can exploit the inherent feature selection and sampling methods in AdaBoost (Freund and Schapire, 1995) for jointly finding the best parts while training the category classifiers. However, where Torralba et al. (2007) learn which parts distinguish multiple categories simultaneously from a common background, our objective is to learn what parts to share to distinguish categories from each other. We extend boosting to explicitly focus on part sharing, and propose a bootstrapping method for fusion with the global image representation.

The work of Azizpour et al. (2015) generalizes latent variable models and shows how explicitly incorporating both Own (referred to as foreground) and Other (referred to as background) parts benefits objects in a binary setting. We find empirically that Own and Other parts are indeed important, in addition to parts from Context. Furthermore, we extend the scope from a binary setting to a multi-class and multi-label setting. And finally, we introduce a method for fusion that exploits the strength of both global and part-based image representations.

Recent work of Parizi et al. (2015) opts for a joint optimization of part detection and image categorization, where the part detectors and image classifiers are optimized alternatively. They show that such a joint optimization over all categories simultaneously improves categorization performance over independently optimized part-based methods, be it that their approach requires significant feature dimension reduction to be feasible. Similar to Azizpour et al. (2015), Parizi et al. (2015) consider Own and Other part types, where we establish the relevancy of three part types (Own, Other, and Context) for sharing among image categories and we demonstrate their effectiveness. Moreover, we use state-of-the-art ConvNet features without the need for dimension reduction. Finally, our algorithm naturally incorporates fusion with the global image representation.

In Song et al. (2013), learn tree classifiers on image parts for object detection. We also rely on trees, be it that we prefer to have many of them (order of $10^3$), and next to objects, we also consider scenes, actions, and fine-grained birds.

3. Part sharing for image categorization

In Fig. 2, we provide an overview of our approach. First, we perform part representation, where parts with their features and learned detectors are extracted from training images. Second, we perform our part-based optimization based on boosting to yield a part-based classifier. Third, we perform our bootstrap fusion between the part-based classifier and a classifier trained on global image representations.

3.1. Part representation

We decompose each image into part proposals which offer a modest set of bounding boxes with a high likelihood to contain any part or object. More formally, let the function $B(X)$ map an image $X$ to $k$ part proposals, $B(X) \rightarrow \{p_1, p_2, \ldots, p_k\}$. For $n$ training images, $T = \{X_1, X_2, \ldots, X_n\}$, we extract and combine all parts as $P_{train} = \cup_{X \in T} B(X)$. In our work, we opt for selective search (Uijlings et al., 2013), but our method applies to any part partition (Cheng et al., 2014; Krähenbühl and Koltun, 2014; Zitnick and Dollár, 2014).

For each part $p_i$, we learn a part detector $d(\psi(p_i))$, where $\psi(p_i) \in \mathbb{R}^d$ denotes the feature representation of $p_i$. For the detector $d()$, we follow Juneja et al. (2013) and transform the part representations into linear classifiers. Each linear classifier serves as an exemplar SVM for the corresponding part representation. We use the fast exemplar SVM approach of Hariharan et al. (2012) to compute the classifiers. Let $\mu_i$ and $\Sigma_i$ denote the mean and covariance estimated from sampled part representation, ignoring their category label. Then the detector function is defined as: $d(p) = \Sigma^{-1}(p - \mu)$. Given a detector for each part, we represent an image $X$ by a feature vector $\mathbf{v}_X \in \mathbb{R}^{|P|}$, where each dimension corresponds to the response of a part detector on $X$. Each dimension $j$ in our image representation vector $\mathbf{v}_X(j)$ is the best-part-detector response for a training part $p_i$ in image $X$.

$$\mathbf{v}_X(j) = \max_{p \in P} d(\psi(p_i)) \cdot \psi(p).$$  \hspace{1cm} (1)

Our image representation has $s = |P|$ dimensions where each dimension is a part from the set $P \subseteq P_{train}$. Our goal is to select the set of shared parts $P$ to use as detectors over all categories.

3.2. Part boosting

We unify part selection $P \subseteq P_{train}$ and image categorization by extending Adaboost (Freund and Schapire, 1995). Boosting minimizes the training error in an iterative manner, converging asymptotically to the global minimum exponential loss (Freund and Schapire, 1995). At each iteration $t$, a weak learner $f_t(\cdot; \Phi_t)$ selects the parts $\Phi_t$ and weights $\alpha_t$ that maximally decrease the current
weighted image classification error while at the same time optimizing the part selection $P$. The classification of a test image $\mathbf{v}_i$ is a weighted sum of the $T$ weak learners under the global constraint of using at most $s$ parts.

$$h^t(\mathbf{X}) = \sum_{t=1}^{T} \alpha_t \cdot f_t(\mathbf{v}_X; \Phi_t), \quad \text{s.t. } |P| \leq s,$$

where $P = \bigcup_{t=1}^{T} \Phi_t$ is the set of all selected parts. Here, the set $\Phi_t$ denotes the parts used to construct a weak learner at iteration $t$. Our formulation extends standard boosting with a global constraint $|P| \leq s$ on the number of parts. By limiting the number of parts, we aim to arrive at a discriminative part-based representation (Bos2 et al., 2014; Doersch et al., 2013; Juneja et al., 2013). The convergence is dependent on the value of $s$; the higher the value, the more training iterations are required.

In the objective, the weak learners $f_t(\mathbf{X})$ take the form of orthogonal decision trees (Dieterich, 2000). The decision trees are constructed by selecting the part with the lowest weighted classification error at each binary split of the tree (Freund and Schapire, 1995). The weights of each weak learner $\alpha_t$ are computed as $\alpha_t = \frac{1}{2} \ln(1 - \epsilon_t)/\epsilon_t$, where $\epsilon_t$ denotes the error rate of the weak learner (Freund and Schapire, 1995).

For all $L$ categories we have $L$ corresponding binary classifiers of Eq. (2). In a multi-label setting each classifier is independent; yet the set of parts $P$ are shared over all categories. In a multi-class setting we have to make a single choice per image:

$$h(\mathbf{X}) = \arg \max_{i \in L} \sum_{t=1}^{T} \alpha_t \cdot f_t(\mathbf{v}_X; \Phi_t), \quad \text{s.t. } |P| \leq s.$$

with now $P = \bigcup_{t=1}^{T} \bigcup_{l=1}^{L} \Phi_t^l$. Note that our added constraint $|P| \leq s$ in Eq. (3) states that at most $s$ unique parts over all categories are used, enforcing the sharing of parts between categories. By limiting the number of parts over all categories simultaneously, each part selected during representation learning should be discriminative for as many categories as possible, i.e., should be shared between categories. We opt for a one-vs-rest objective instead of a multi-class objective (Eibl and Pfeiffer, 2005; Saberian and Vasconcelos, 2011) to allow for both multi-class and multi-label image categorization.

**Maximum exploiting sampling.** We introduce a part sampling algorithm for re-using parts and sharing parts between categories, called maximum exploiting. The main idea of maximum exploiting is to gradually increase the cardinality of $P$ by balancing exploitation: selection of new parts from $P_{\text{train}}$, and exploitation: reusing and sharing of previously selected parts $P$.

The idea behind Maximum Exploit sampling builds upon the work by Freund and Schapire (1995), which states that the training error on a strong learner reduces exponentially with the number of non-random weak learners. Here, we use this result for joint part selection and image categorization. As long as we can train non-random weak learners on the same set of parts, we are guaranteed to reduce the training error rate. We exploit this maximally in our sampling approach.

At train iteration $t$, we force the decision tree $f_t(i; \Phi_t)$ to select parts $\Phi_t$ exclusively from $P$, exploiting the shared pool. When the classification performance saturates and $P$ is maximally exploited, we allow a single exploration step, selecting new parts $\Phi_t$ from $P_{\text{train}} \backslash P$. In Fig. 3 we illustrate our algorithm. Our sampling scheme minimizes the cardinality of the selected parts $P$, forcing the decision trees to reuse the shared parts from all categories as long as this positively affects the empirical loss over the training examples.

To determine if $P$ is maximally exploited, we measure the saturation of $P$ by the classification error within the last range of exploitation iterations. For a category $l$, let this range start at iteration $u$ and let the current iteration be $v$. Then the error is defined as:

$$e_{uv} = \sum_{l=1}^{L} \left| \sum_{i=1}^{v} \alpha_t \cdot f_t(\mathbf{v}_X; \Phi_t) \right| \neq Y_l^i,$$

where $Y_l^i \in \{-1, +1\}$ states whether image $i$ is of category $l$. The error of Eq. (4) states the number of mis-classifications using the weak classifiers created in the current exploitation iterations. Let $e_{uv}$ denote the average error over all categories within range $[u, v]$, i.e.:

$$e_{uv} = \frac{1}{L} \sum_{l=1}^{L} e_{uv}.$$  

We keep exploiting if $e_{uv} < e_{uv-1}$, otherwise we perform a single exploration step and restart our exploitation evaluation, setting $u = v$.

### 3.3. Bootstrap fusion

We extend our shared part selection and sampling to a fusion with the global image representation. Instead of early or late fusion that would combine the part-based representation independently with the global image representation (Doersch et al., 2013; Juneja et al., 2013; Snoek et al., 2005), we fuse the representations from the start. We propose bootstrap fusion, which jointly optimizes part-selection and image classification by selecting those parts that gain the most from fusing with the global image representation.

In our bootstrap fusion, we start by applying our maximum exploitation on the global image representation $F$. Afterwards, our idea is to bootstrap the AdaBoost weights of the training images as initialization for the part representations. This bootstrapping provides information about the difficulty per training image for categorization according to the global representation. By transferring the weights to the part-based model, the part-based representation...
learning is forced to focus on the mistakes caused by the global representation. This fusion procedure enhances the complementary nature between the part and global representations. In our fusion approach, the dimensionality is equal to the dimensionality of the global representation and the number of selected parts $s$ in the part representation.

When transferring the weights, certain training examples have weight values $w_i$ up to several orders of magnitude higher or lower than the average weight value. The reason for this is the presence of the exponent in the weight update in AdaBoost (Freund and Schapire, 1995). Training examples that are consistently correctly or incorrectly classified in the initial boosted classifier will yield extreme weight values. Although we want to focus on the hard examples, we are not interested in focusing solely on these examples. To combat the high variance of the weights, we perform a power normalization step (Sánchez et al., 2013), followed by an $\epsilon_1$ normalization to make the weights a distribution again:

$$w_i = \frac{w_i^\alpha}{\sum_{j=1}^{N} w_j^\alpha}.$$  

(6)

Throughout our experiments, we set $\alpha = \frac{1}{2}$. The use of power normalization results in a more balanced weight distribution, which in turn results in better fusion performance. Fig. 4 highlights the effect of power normalization on the weight values.

During testing, we apply the same part and feature extraction as during training. We apply our boosted classifier with the $s$ selected parts to the parts of the test image. Idem, the boosted global classifier is applied to the global image representation of the test image. The confidence scores of both classifiers are summed and the category with the highest confidence is selected as the target class.

4. Experiments

4.1. Datasets

In our experiments, we consider 4 datasets.

**Pascal VOC 2007** (Everingham et al., 2010) consists of 9963 images and 20 object categories, where we report on the provided train/validation split using the Average Precision score (Everingham et al., 2010).

**MIT 67 Indoor Scenes** (Quattoni and Torralba, 2009) consists of 6700 images and 67 scene categories, where we use the provided 80/20 split for each scene category (Quattoni and Torralba, 2009), reporting results using multi-class classification accuracy.

**Willow Action** (Delaitre et al., 2010) consists of 911 images and 7 action categories, where we use the train/test split of Delaitre et al. (2010), reporting with the Average Precision score.

**Calltach-UCSD Birds 200–2011** (Wah et al., 2011) consists of 11,788 images and 200 fine-grained bird categories, where we use the train/test split of Wah et al. (2011), reporting with the multi-class classification accuracy.

4.2. Implementation details

**Part extraction.** During both training and testing, we use selective search as the function $B(x)$ which splits each image into roughly 2000 parts (Uijlings et al., 2013). Due to the hierarchical nature of selective search, we observe that parts vary with sizes, from superpixels to the complete image. Although selective search is intended for objects, we observe that it in fact produces parts suitable for our purpose. On Pascal VOC 2007, we derive there are over 22 parts for each labeled object for which the part of score from Vezhnevets and Ferrari (2015) is at least 0.5.

**Feature extraction.** As features, we employ a GoogLeNet convolutional neural network (Szegedy et al., 2015), pre-trained on 15 k ImageNet categories (Deng et al., 2009). For a given part, we rescale it to 256 × 256 pixels and feed it to the network, resulting in a 1,024-dimensional feature vector, which is subsequently $\epsilon_2$ normalized.

**Part detection.** We use the fast exemplar SVM of Hariharan et al. (2012) to transform the part representations into linear classifiers. The linear classifier of a part is used for the max-pooling operation of Eq. (1). For fair comparison, the max-pooling operation is only applied to the whole image. Both Juneka et al. (2013) and our approach will improve further when we incorporate pooling over the scene layout by Lazebnik et al. (2006). We sample roughly 400 k part features from train images to estimate the corresponding $\mu$ and $\Sigma$ parameters.

**Part boosting.** For the weak learners, we opt for orthogonal decision trees, as AdaBoost with decision trees have shown to yield excellent categorization performance (Dietterich, 2000). At each split, we select the part that minimizes the normalized weighted misclassification error of the examples in the corresponding node. As shown by Bühlmann and Yu (2003), AdaBoost is resistant to over-fitting when using a large number of training iterations. Therefore, we use 2000 iterations for boosting throughout our experiments for training the image classifiers.

4.3. Experimental evaluation

**Experiment 1: All part types matter.** We first motivate our main hypothesis that part-based methods benefit from three types of category parts. We rely on the Pascal VOC 2007 dataset for this experiment. As a surrogate for each type of part, we use the object bounding boxes as part locations. For each category, we use the features from the bounding boxes of the same category as its own parts. Similarly, the features from the bounding boxes of all other categories are used as other parts. For context parts, we use the features from selective search boxes without overlap to any ground truth bounding box. As a baseline, we use the performance of the global image representation. We add the features of each part type to the global representation to evaluate their effect.

**Results.** We show the image categorization results for the global representation and its combination with each part type in Fig. 5. As the figure indicates, adding each of the different part types boosts the categorization performance. Not surprisingly, the use of own parts for each category yields the best improvement (89.6% vs. 85.3% mean Average Precision (mAP)). We especially note the increase obtained for small objects such as bottle (from 55.1% to 76.3%) and potted plant (from 64.2% to 74.4%).

Surprisingly effective is the addition of other parts, with an mAP of 88.6%. For multiple categories, such as sofa, it is even
more favorable to add parts from other categories over parts from their own category. More specifically, using parts from cats and dogs yields a better performance than using parts from sofa itself. This result highlights the importance of exploiting and sharing the co-occurrence of categories. Lastly, the overall performance of context parts is marginally beneficial, compared to own and other parts (86.1% mAP). For categories such as car (94.0%), motorbike (91.7%), and tv (84.1%), the use of context parts may be good choice.

We note that experiment 1 serves to show that all part types matter for image categorization. Since we rely for this experiment on ground truth bounding boxes to approximate parts, the results should not directly be compared to other part-based methods, who all rely on the class labels only. A direct comparison will be discussed in experiment 2 and beyond.

**Qualitative analysis.** To understand why each part type matters for image categorization, we provide additional qualitative results. In Fig. 6, we show test examples that significantly improve or decrease in predictive performance when each of the part types is added.

For own parts, we observe that the improvement is most significant for small objects, objects out of context, and occluded objects, as shown in Fig. 6a–c for respectively bottle, sofa, and bus. Exploiting and sharing parts from other categories is beneficial when there is a high co-occurrence between categories in the images. This pattern is shown in Fig. 6e–g, for sofa / dog, chair / dining table, and horse / person. Parts from context are similarly beneficial in case of a co-occurrence between the categories and contextual elements. Notable examples are buildings for boat, cat basket for cat, and fence for sheep, as shown in Fig. 6i–k.

Fig. 6 furthermore shows where each part type fails. For own parts, performance decreases e.g. when the object cannot be located, as shown in Fig. 6d. Other category parts fail for inconsistent co-occurrence, such as car and tv in Fig. 6h. For context parts, performance decreases when context is either not present or consistent with other object categories, as shown in Fig. 6l.

Fig. 1 also shows that different part types focus on different elements in the scene, taking in the complete information from the image. Based on the quantitative and qualitative results of this experiment, we conclude that all part types matter for image categorization and should be exploited to improve image categorization.

**Experiment 2: Evaluating our part-based method**

We evaluate our joint part selection and image categorization without box information in three steps: (i) evaluate whether we capture all part types, (ii) compare to separate part optimization, and (iii) compare our maximum exploit sampling to baseline samplers.

**Do we capture all types of parts?** First, we validate that our method follows our hypothesis and we ask ourselves; do we capture all part types? We perform this evaluation on Pascal VOC 2007, as the object annotations of this dataset enables such an evaluation. To validate that our method is capable of incorporating all part types, we analyze the importance of each selected part as a function of the box agreement. We run our algorithm to select a total of 500 parts across all 20 categories in Pascal VOC 2007. We note that this setting yields an mAP of 89.1%, significantly higher than the global representation and on par with the representations from the strongly supervised bounding boxes used in the first experiment. For each selected part p, we compute its importance in a single decision tree as the normalized miss-classification reduction; this value is summed over all the decision tree where the part is used. For the selected part p and the best overlapping box b, the box agreement is computed here as $\frac{p}{b}$ (Vezhnevets and Ferrari, 2015). Intuitively, the box agreement states to what extend $p$ is part of $b$.

The relation between part importance and box agreement is shown in Fig. 7. The figure shows two clear peaks. The leftmost peak indicates that our method utilizes parts with no overlap to ground truth boxes, i.e. context parts. The other peak is at a box agreement of 1: when a part is contained in an object. The red and blue bars indicate that each category uses parts from its own object and from other objects. From the figure we conclude that our method uses own (blue), other (red), and context (green) parts.

We also provide part responses by our method in Fig. 8. The Figure shows how our method deals with occluded objects (8a), small objects (8b), and atypical object views (8c). The primary focus is on the visible portions of the objects and co-occurring other objects, but contextual information is also taken into account. Examples include car jack (car, 8a), steam (train, 8d), and horizontal bars (horse, 8g). Our method automatically discovers that people parts should be shared for dog (8f), but dog parts should not be shared for bike (8e).

**Joint vs. separate selection and categorization.** Second, we compare our method to the bag-of-parts of Juneja et al. (2013), which applies a separate optimization of part selection and image categorization. To ensure a fair comparison, we replicate the setup described in Juneja et al. (2013). This means that we use the same dataset (MIT 67 Indoor Scenes), the same part proposals, and the same part features. For the HOG features used in Juneja et al. (2013), we also employ the outlined iterative part re-
Fig. 6. Qualitative examples of when the different part types improve/decrease the categorization performance. Shown are improvements using own (a–d), other (e–h), and context (i–l) parts. The first three examples have increased performance, the last example has decreased performance. The icon denotes the category, while the numbers state the original and new ranking position.

Fig. 7. Histogram showing the importance of our selected parts, as a function of the overlap with annotated objects. On the x-axis, we show the distribution of the overlap between our parts and ground truth objects. On the y-axis, we show how important the parts have been during training. The plot shows that our parts focus on parts from its own category (blue), while also taking in other categories (red) and context (green). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Refinement. For full details, we refer to Juneja et al. (2013). We also provide a comparison using the GoogleNet features for the parts. For the bag-of-parts, we report the results with both a linear and $\chi^2$ SVM, as is proposed in Juneja et al. (2013). We only exclude image flipping and spatial pyramids, as these elements are not specific to either method and can therefore cloud the comparison.

In Fig. 9a, we show the classification accuracies using respectively the HOG and GoogleNet features. The scores the Figure are compared for four values of constraint s to evaluate the influence on the performance as a function of s for both our approach and the method of Juneja et al. (2013). We enforce four global part limits: 335, 670, 1675, and 3350 parts on MIT 67 Indoor Scenes and 100, 200, 500, 1000 on Pascal VOC 2007. For the baseline bag-of-parts approach, this corresponds to 5, 10, 25, and 50 parts selected per category.

For the HOG features, we observe that our joint part selection outperforms the separate optimization in the bag-of-parts of Juneja et al. (2013). The difference even grows as we use stricter part limits. At 5 parts per category, our method outperforms bag-of-parts by 7.6% ($\chi^2$) and 12.6% (linear). For the GoogLeNet features, we first observe that all results significantly outperform the HOG features (note the different ranges on the y-axis). We again outperform the bag-of-parts, albeit with a less pronounced difference. At five parts per category, we outperform the bag-of-parts by 3.6% ($\chi^2$) and 5.6% (linear). We attribute this to the strength of the features, resulting in saturation.

The run time of our approach is similar to other part-based methods, as they all compute part responses and apply learned classifiers. Naturally, the training time of our joint selection and classification is more involved than methods with separate selection and classification (Doersch et al., 2013; Juneja et al., 2013), yet fast enough to avoid dimensionality reduction, as in Parizi et al. (2015). On a single Intel Xeon E5-2609 core, it takes roughly 8 h to jointly optimize all 20 objects in Pascal VOC 2007. On a test image, it takes 2 s to max-pool the selected parts and apply the trees.

We conclude that our joint part selection and categorization is preferred over a separate optimization.

Evaluating maximum exploiting. Third, we evaluate the maximum exploiting itself within the joint AdaBoost optimization. We
reuse the HOG settings of Juneja et al. (2013) and draw a comparison to three baseline sampling strategies. The first baseline, Tieu-Viola, follows (Tieu and Viola, 2004). At each boosting iteration, a single new part for each category is selected until the specified part limit is reached. The second baseline, uniform, is a variant of LazyBoost (Escudero et al., 2000), where we randomly sample parts from the set of all parts (ignoring whether parts have been selected or not), until the limit is reached, after which we keep retraining on the selected parts. The third baseline follows the $\epsilon$-greedy strategy, where we have a static probability $\epsilon$ of exploring new parts and a static probability $1 - \epsilon$ of exploiting the current set of selected parts.

The comparison of maximum exploit sampling versus the baseline boosting sampling methods is shown in Fig. 9b. The maximum exploiting joint optimization is preferable across the whole range of part limits. These results indicate that a higher focus on exploitation is an important aspect in the joint optimization, since the baselines have a higher focus on exploration. It is also interesting to note that, with the exception of the Tieu-Viola baseline, all variants improve over Juneja et al. (2013), indicating the effectiveness of a joint optimization.

**Experiment 3: Evaluating bootstrap fusion.**

We evaluate our bootstrap fusion for combining our category parts with the global representation. We perform this evaluation on both Pascal VOC 2007 and MIT 67 Indoor Scenes and provide a comparison to two baseline fusions (Snoek et al., 2005). For the bootstrap procedure, we train a model on the global representations and then train our joint part selection and image categorization on the updated weights. The combination is then formed by concatenating the decision trees from both models. The first baseline, early fusion, combines the global representations and the representations from the parts selected by Juneja et al. (2013), after which a $\chi^2$ SVM is trained on the concatenated representations. The second baseline, late fusion, is similar to the bootstrap procedure, but without updating the weights. In the remaining experiments we always use an average of 50 parts per category for our approach.

**Results.** In Fig. 10 we show the performance of our bootstrap fusion against the two baselines. Across all settings and part limits, our fusion (blue bars) outperforms the baselines. For Pascal VOC 2007 our method outperforms the other combiners between 2–4% across all part limits, with an mAP up to 90.7% at 50 category parts per object. For MIT 67 Indoor Scenes, the difference in performance varies for both the GoogLeNet and HOG features. For the HOG features with the Fisher Vector as the global representation, the improvement is most significant, similar to the results of Fig. 9a. For the GoogLeNet features the improvement is less significant, but our bootstrap is still better than the baselines. We conclude that
our bootstrap procedure is preferred when combining the part and global representations.

Comparison to the state-of-the-art. We also compare our results for the fusion to related work. The comparison of our results to the state-of-the-art on Pascal VOC 2007 is provided in Table 1. Our results provide an improvement over the references. Our final result compares favorably to the convolutional neural network applied to the whole image (90.7% vs. 85.3%), a notable improvement in mAP. We note that recent concurrent work on (very deep) convolutional networks similar in spirit to Szegedy et al. (2015) has yielded an mAP of 89.3% (Simonyan and Zisserman, 2015), which is a 4% improvement over the network that we employ. Both our part-based and combined results still outperform Simonyan and Zisserman (2015) and we expect a further improvement when employing their network to represent the parts. In Fig. 12, we highlight 3 selected parts during training on MIT 67 Indoor Scenes, along with top responses in three test images.

The comparison for MIT 67 Indoor Scenes and Willow Actions is also provided in Table 2. For scenes, we see that our results improve upon existing combinations of global and part-based representations (Doersch et al., 2013; Juneja et al., 2013). Similar to Pascal VOC 2007, we note that the recently introduced scene-specific
Table 1
Average Precision (%) scores on Pascal VOC 2007, comparing our results to related work. Our part-based results improve upon existing global representations (from 85.3% for Szegedy et al. (2015) to 89.7%), while the combination with Szegedy et al. (2015) yields the best results.

<table>
<thead>
<tr>
<th>Method</th>
<th>Pascal VOC 2007</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oquab et al. (2014)</td>
<td>88.5 81.5 87.9</td>
</tr>
<tr>
<td>Chatfield et al. (2014)</td>
<td>95.3 90.4 92.5</td>
</tr>
<tr>
<td>Wei et al. (2014)</td>
<td>96.0 92.1 95.7</td>
</tr>
<tr>
<td>Cimpoi et al. (2015)</td>
<td>91.4 90.9 91.2</td>
</tr>
<tr>
<td>Szegedy et al. (2015)</td>
<td>97.7 93.1 94.4</td>
</tr>
<tr>
<td>Simonyan and Zisserman (2015)</td>
<td>89.3</td>
</tr>
<tr>
<td>Ours</td>
<td>98.0 96.8 97.9</td>
</tr>
<tr>
<td>Ours + (Szegedy et al., 2015)</td>
<td>98.7 97.0 97.9</td>
</tr>
</tbody>
</table>

Table 2
Comparison with state-of-the-art on MIT 67 Indoor Scenes, Willow Actions, and Caltech-UCSD Birds 200–2011. Here, ft denotes fine-tuning. Using a single feature only (Szegedy et al., 2015), our approach is competitive on all datasets and best for Pascal Objects and Willow Actions and Fine-grained Birds without fine-tuning.

<table>
<thead>
<tr>
<th>MIT 67 Indoor Scenes</th>
<th>Willow Actions</th>
<th>Caltech-UCSD Birds 200–2011</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>Acc.</td>
<td>Method</td>
</tr>
<tr>
<td>----------------------</td>
<td>----------------</td>
<td>-----------------------------</td>
</tr>
<tr>
<td>Juneja et al. (2013)</td>
<td>63.1%</td>
<td>Pina et al. (2014)</td>
</tr>
<tr>
<td>Doersch et al. (2012)</td>
<td>66.9%</td>
<td>Sire et al. (2014)</td>
</tr>
<tr>
<td>Szegedy et al. (2015)</td>
<td>69.3%</td>
<td>Delalire et al. (2010)</td>
</tr>
<tr>
<td>Zhou et al. (2014)</td>
<td>70.8%</td>
<td>Sharma et al. (2012)</td>
</tr>
<tr>
<td>Zou et al. (2014)</td>
<td>76.2%</td>
<td>Sharma et al. (2013)</td>
</tr>
<tr>
<td>Parizi et al. (2015)</td>
<td>77.1%</td>
<td>Khan et al. (2012)</td>
</tr>
<tr>
<td>Cimpoi et al. (2015)</td>
<td>81.0%</td>
<td>Szegedy et al. (2015)</td>
</tr>
<tr>
<td>Ours</td>
<td>77.4%</td>
<td>Ours</td>
</tr>
</tbody>
</table>

From this evaluation, we conclude that our method improves upon global ConvNet image representations and yields competitive results compared to the current state-of-the-art.

5. Conclusions

We have investigated image categorization using a representation of parts. We start from the intuition that parts are naturally shared between categories. We analyze the type of region where a part comes from, namely: parts from its own category, parts from other categories, and context parts. Experimental validation confirms that all the types of parts contain valuable and complementary information. To share parts between categories we extend Adaboost, profiting from joint part selection and image classification. Experimental evaluation on object, scene, action, and fine-grained categories shows that our method is capable of incorporating all three types of parts. Furthermore, our method provides a further boost in performance over a convolutional representation of the whole image for categorization. The strong performance of our and other state-of-the-art methods on Pascal VOC 2007, MIT 67 Indoor Scenes, and Caltech-UCSD Birds 200–2011 indicates the saturation on these datasets and the need for the community to explore bigger datasets for part-based image categorization in the near future. Our method opens up new possibilities for part-based methods, such as image captioning by sharing parts over similar words within the captions.

Acknowledgments

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

References


Doersch, C., Gupta, A., Efros, A.A., 2013. Mid-level visual element discovery as discriminative mode seeking. NIPS.


Eibl, G., Pfeiffer, K.P., 2005. Multiclass boosting for weak classifiers. MLJMR.

Endres, I., Shih, K.J., Jia, J., Heim, D., 2013. Learning collections of part models for object recognition. CVPR.


Freund, Y., Schapire, R.E., 1995. A desicion-theoretic generalization of on-line learning and an application to boosting. COLT.


Krähenbühl, P., Koltun, V., 2014. Geodesic object proposals. ECCV.


Lin, T.Y., RoyChowdhury, A., Maji, S., 2015. Bilinear cnn models for fine-grained visual recognition. ICCV.


