Dictionary Pair Classifier Driven Convolutional Neural Networks for Object Detection

Keze Wang[1,3], Liang Lin[1,*], Wangmeng Zuo[2], Shuhang Gu[3], Lei Zhang[3]

[1]School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China

[2]School of Computer Science and Technology, Harbin Institute of Technology, China

[3]Department of Computing, The Hong Kong Polytechnic University

linliang@ieee.org; {kezewang, cswmzuo}@gmail.com; {cssgu, cslzhang}@comp.polyu.edu.hk.

Abstract

Feature representation and object category classification are two key components of most object detection methods. While significant improvements have been achieved for deep feature representation learning, traditional SVM/softmax classifiers remain the dominant methods for the final object category classification. However, SVM/softmax classifiers lack the capacity of explicitly exploiting the complex structure of deep features, as they are purely discriminative methods. The recently proposed discriminative dictionary pair learning (DPL) model involves a fidelity term to minimize the reconstruction loss and a discrimination term to enhance the discriminative capability of the learned dictionary pair, and thus is appropriate for balancing the representation and discrimination to boost object detection performance. In this paper, we propose a novel object detection system by unifying DPL with the convolutional feature learning. Specifically, we incorporate DPL as a Dictionary Pair Classifier Layer (DPCL) into the deep architecture, and develop an end-to-end learning algorithm for optimizing the dictionary pairs and the neural networks simultaneously. Moreover, we design a multi-task loss for guiding our model to accomplish the three correlated tasks: objectness estimation, categoryness computation, and bounding box regression. From the extensive experiments on PASCAL VOC 2007/2012 benchmarks, our approach demonstrates the effectiveness to substantially improve the performances over the popular existing object detection frameworks (e.g., R-CNN [13] and FRCN [12]), and achieves new state-of-the-arts.

1. Introduction

Aiming at finding instances of real-world objects from images or video sequences, object detection has been attracting great interests in computer vision community. Although its performance has been improved substantially in the past decade [31, 34, 21, 8, 30, 33, 13, 16], object detection remains a challenge problem under complex and unconstrained environments.

Recently, ground breaking progress on object detection has been made due to the advances in deep convolutional neural networks (CNNs) [19, 28] and the increasing size of training dataset [5]. The state-of-the-art object detection methods generally adopt the region-based CNN framework which includes three components: region proposal, feature extraction and object category classification. By far, many region proposal methods [31, 3, 24] and deep CNN architectures [13, 26, 28, 27, 16, 12] have been proposed, but not too many methods have been proposed for object category classification, where the SVM/softmax classifiers are dominantly used. Several complex classifiers, such as network on convolutional feature maps (NoCs) [25] and structured SVM [38], have been developed to improve the accuracy and robustness of object detection. These classifiers, however, are fully discriminative methods which directly learn an optimal mapping from the CNN features to the desired classification output.

Combining of discriminative learning with representation or generative modeling is beneficial to exploit the complex structure of CNN features for improving object detection. As an extension of the reconstructive dictionary learning proposed in image and signal modeling, discriminative dictionary learning (DDL) has achieved great success in the last decade [22, 9, 36, 18]. DDL aims to learn a dictionary by considering both its representation accuracy and discriminative capability, and thus it is more suitable to act as a classifier for object category classification. However, the existing DDL methods cannot achieve state-of-the-art performance for large scale image classification and object detection, partially due to that the DDL models have only been evaluated with conventional handcrafted features (e.g.,

*Corresponding author is Liang Lin (Email: linliang@ieee.org).

SIFT and HOG). Therefore, it is interesting to investigate whether we can significantly boost the object detection performance of DDL by utilizing more powerful deep CNN features.

Computational burden is another obstacle which restricts the application of DDL to large scale scenarios. Most DDL models involve costly ℓ_0- or ℓ_1-norm regularization to generate sparse coding vectors, limiting their use to the scenario with high feature dimension and large volumes of data. Fortunately, Gu *et al*. [15] suggested a projective dictionary pair learning (DPL) method, which improves greatly the computational efficiency. To avoid costly sparse coding, DPL adopts an analysis dictionary to generate coding vector via linear projection and a synthesis dictionary for class-specific discriminative reconstruction, respectively. In this work, we propose to design a dictionary pair classifier layer (DPCL) at the end of the CNN for object detection. For readability, some main abbreviations of this paper are listed in Tab. 1.

Rather than learning the CNN and the dictionary pair separately, we adopt a joint training mechanism for simultaneous feature learning and classifier learning. A dictionary pair back propagation (DPBP) algorithm is proposed to jointly update the parameters of CNN and DPCL in an end-to-end learning manner. With DPBP, we can fine-tune the trained CNN to extract discriminative features specialized to DPCL. Meanwhile, DPCL is tailored to the learned CNN features and better detection results can be expected.

Furthermore, we present a sample weighting scheme in DPCL to improve the localization accuracy. As analyzed in [13], poor localization remains the major type of detection errors. One major cause of inaccurate localization is that the objective of classifier is to correctly predict the category label of the object, while the objective of detection is to accurately estimate the location. To make classification conformable with localization, careful selection of thresholds of the intersection-over-union (IoU) with the ground truth is important to define positive and negative samples [13]. To alleviate the inaccurate localization, Zhang *et al*. [38] adopted the structured SVM classifier to simultaneously predict category and location, while Girshick [12] suggested a multi-task loss to balance between classification and localization. Different from [38, 12], we introduce a predefined weight to each training sample based on its IoU overlapping with the ground truth bounding box, encouraging the samples with higher IoUs (*i.e*., better localization) to have lower reconstruction residual (*i.e*., higher score). Experimental results show that the weighting scheme in DPCLs can further improve the detection performance.

Motivated by the success of multi-task learning [4] in object detection [12], we present a novel multi-task loss for joint training of the DPCL and bounding-box regressor. In [12], Girshick considered two learning tasks, where the classification task loss is on the probability distribution over $K+1$ categories (K object categories and one background category), and the location task loss is on the bounding box regression offsets. In this work, we divide the classification task into two related ones, i.e., an objectness task to distinguish object from background and a categoryness task to recognize the category of the object. Although the objectness [2] can be used as a pre-filtering process in object detection [13, 16], its potential remains untapped and not fully released. First, most objectness measures are based on hand-crafted features, while the learned objectness on deep CNN features can further benefit object detection. Second, the incorporation of objectness and categoryness allows us to use the coarse-to-fine strategy for object category classification. Third, our objectness detection task is not aimed to learn a general objectness measure but to learn a classifier to distinguish background from objects of interest. To this end, we employ two separate DPCLs to accomplish the two correlated tasks, *i.e*., objectness learning and categoryness learning, and our multi-task loss includes three tasks: objectness, categoryness, and localization. Compared with [12], we adopt a hybrid fusion strategy, where the product rule is used to fuse objectness score and categoryness score into classification score, and the sum rule is then utilized to combine classification score and localization loss. Moreover, DPBP can also be extended to minimize the multi-task loss in an end-to-end manner.

By integrating DPCL classifier training with CNN feature learning, the proposed method achieves about 3% / 2% mAP gain over the popular existing object detection frameworks (e.g., R-CNN [13], FRCN [12]) on PASCAL VOC 2007/2012 benchmarks, respectively. This establishes the significance of the joint learning framework as well as the proposed multi-task loss. In summary, the contributions of this work are three-fold. i) A novel deep architecture is developed by integrating DPCL with CNN for objection detection, and a DPBP algorithm is suggested for the end-to-end learning of CNN and DPCL parameters. ii) Based on the R-CNN [13]/FRCN [12] framework, we propose a novel multi-task loss by combining objectness estimation, categoryness computation and bounding box regression to improve the detection performance. iii) A sample weighting scheme is introduced to assign larger weight to the training samples with higher IoU with the ground truth, which can further improve the location accuracy of object detection.

DPL	Dictionary Pair Learning
DPCL	Dictionary Pair Classifier Layer
DPBP	Dictionary Pair Back Propagation
ODP	Objectness Dictionary Pair
CDP	Categoryness Dictionary Pair

Table 1. Some main abbreviations are used in this paper.

2. Related Work

Deep Convolutional Neural Networks. By directly learning features from raw images, deep convolutional neural networks (CNNs) have made impressive progresses on image classification, object detection, semantic segmentation and many other recognition tasks [19, 28, 13, 1, 17]. Motivated by the success of CNNs [19] on the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [5], a variety of CNN-based object detection methods have been proposed. Szegedy *et al*. [30, 6] treated object detection as a regression problem, and trained CNNs to predict object bounding boxes (MultiBox) or bounding box masks (DetectorNet). Overfeat [26] suggested by Sermanet *et al*. adopts the sliding window scheme, and uses two CNNs to predict the objectness and the true bounding box location, respectively. Deformable parts models (DPMs) can also be explained from the CNN perspective, and the integration of DPMs and CNNs has been investigated in [32, 14].

Most recent object detection methods are based on the R-CNN framework [13], which includes three main components: region proposal, feature extraction and object category classification. To improve the efficiency of region proposal generation, Szegedy *et al*. [29] improved MultiBox by using the Inception-style network, contextual features and robust loss, while Ren *et al*. [24] suggested a region proposal network (RPN). To improve the efficiency of detection network and avoid region proposal resizing, spatial pyramid pooling networks (SPPnets) [16] and fast R-CNN [12] proposed to introduce a RoI-pooling layer to extract fixed-size proposal features from shared convolutional feature maps of the entire image. For better classification and localization, Girshick adopted a multi-task loss, and Zhang *et al*. [38] used a fine-grained Bayesian search algorithm for region proposal refining and a structured SVM classifier for simultaneous classification and localization. Besides, contextual information, *e.g*., background, parts, and segmentation, can also be utilized to improve the detection performance [11, 39].

Discriminative Dictionary Learning. Discriminative dictionary learning (DDL) plays an important role in sparse representation or collaborative representation based classifier [35, 37], and has been intensively studied in computer vision community. Generally, there are two approaches to enhance the discriminative capability of the learned dictionary. First, the discrimination can be imposed on the coding vectors to have a better classification performance. Jiang *et al*. [18] introduced a binary class label sparse code matrix to encourage samples from the same class to have similar sparse codes. Mairal *et al*. [22] proposed a task driven dictionary learning (TDDL) framework, which minimizes different risk functions of the coding coefficients for different tasks. Yang *et al*. [36] proposed a Fisher discrimination dictionary learning (FDDL) method which applies the Fisher criterion to representation coefficients.

Second, the discrimination can also be obtained by learning structured dictionary, *i.e*., learning a sub-dictionary for each class and minimizing the class-specific residual [36]. Ramirez *et al*. [23] used a structured incoherence term to enforce the independence of the sub-dictionaries. Besides the sub-dictionaries, Gao *et al*. [10] learned an extra shared dictionary to encode common features shared by all classes. To improve the efficiency of DDL, Gu *et al*. [15] proposed a projective projective dictionary pair learning (DPL) model by utilizing an analytic dictionary to estimate the representation coefficients efficiently.

3. Integration of DPCL and CNN

3.1. The Dictionary Pair Classifier Layer

3.1.1 Layer Description

Let $\mathbf{X} = [\mathbf{X}_0, ..., \mathbf{X}_k, ..., \mathbf{X}_K]$ ($\mathbf{X}_k \in \mathcal{R}^{d \times n_k}$, n_k is the number of training samples for the k-th category) denote a set of previous layer's d-dimensional outputs for the input image regions $\mathbf{I}$ from $K+1$ categories. The DPCL aims to find a class-specific analysis dictionary $\mathbf{P} = [\mathbf{P}_0, ..., \mathbf{P}_k, ..., \mathbf{P}_K] \in \mathcal{R}^{m(K+1) \times d}$ ($\mathbf{P}_k \in \mathcal{R}^{m \times d}$) and a class-specific synthesis dictionary $\mathbf{D} = [\mathbf{D}_0, ..., \mathbf{D}_k, ..., \mathbf{D}_K] \in \mathcal{R}^{d \times m(K+1)}$ ($\mathbf{D}_k \in \mathcal{R}^{d \times m}$) to analytically encode and reconstruct the feature $\mathbf{X}$, where m is the number of dictionary atoms. The sub-dictionaries $\mathbf{P}_k$ and $\mathbf{D}_k$ form a dictionary pair for the k-th category. Given $\mathbf{P}_k$ and $\mathbf{D}_k$, the encoding coefficients $\mathbf{A}_k$ of the k-th category training samples $\mathbf{X}_k$ over synthesis $\mathbf{D}_k$ can be analytically obtained as $\mathbf{A}_k = \mathbf{P}_k\ \mathbf{X}_k$. Compared to the costly l_0-norm or l_1-norm non-linear sparse coding operation in most of the existing DDL methods, it is quite efficient to resolve the code $\mathbf{A}_k$ for the representation of $\mathbf{X}_k$ in DPL. To learn such an analysis dictionary $\mathbf{P}$ together with the synthesis dictionary $\mathbf{D}$, the DPL model [15] is formulated as:

$$\{\mathbf{P}^*, \mathbf{D}^*\} = \arg\min_{\mathbf{P},\mathbf{D}} \sum_{k=0}^{K} \|\mathbf{X}_k - \mathbf{D}_k\mathbf{P}_k\mathbf{X}_k\|_F^2 + \Phi\{\mathbf{P}, \mathbf{D}, \mathbf{X}, \mathbf{Y}\}, \quad (1)$$

where $\mathbf{Y}$ represents the category label matrix of samples in $\mathbf{X}$, and $\Phi\{\mathbf{P}, \mathbf{D}, \mathbf{X}, \mathbf{Y}\}$ is some discrimination term to promote the discriminative power of $\mathbf{D}$ and $\mathbf{P}$.

In the original DPL [15], the sub-dictionary $\mathbf{P}_k$ is enforced to project the samples $\overline{\mathbf{X}}_k$ from another category i, $i \neq k$, to a nearly null space, *i.e*., $\mathbf{P}_k\mathbf{X}_i \approx \mathbf{0}, \forall k \neq i$. With this constraint, the coefficient matrix $\mathbf{A}_k$ is nearly block diagonal. However, the original DPL does not consider the fact that different training samples may play different importance in training a discriminative model. In this work, we introduce a diagonal importance weight matrix $\mathbf{W}_k$ to the k-th category of training samples, and the proposed DPCL is then defined as:

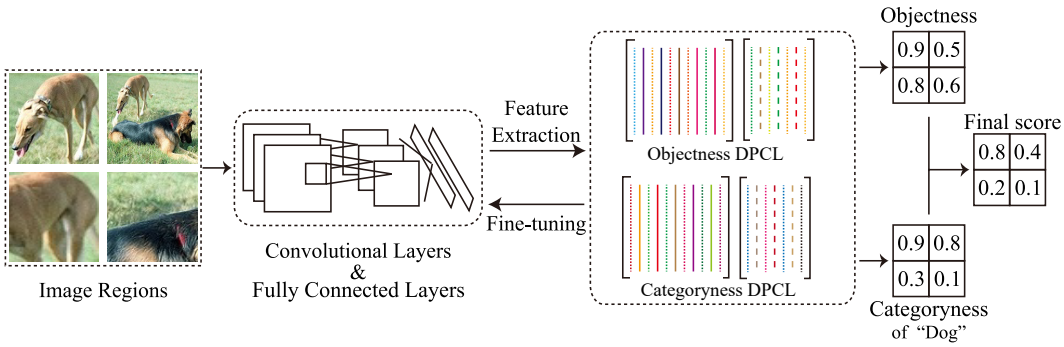

Figure 1. The flowchart of our multi-loss CNN+DPCL model. Our model is stacked up by convolutional layers, fully connected layers, objectness DPCL and categoryness DPCL. The four values inside 2×2 grids are corresponding to the four input regions. The final score for each image region is the combination of objectness and categoryness.

$$\{\mathbf{P}^*, \mathbf{D}^*\} = \arg\min_{\mathbf{P},\mathbf{D}} \sum_{k=0}^{K} \|(\mathbf{X}_k - \mathbf{D}_k\mathbf{P}_k\mathbf{X}_k)\mathbf{W_k}\|_F^2 + \lambda\|\mathbf{P}_k\overline{\mathbf{X}}_k\|_F^2 + \kappa\|\mathbf{D}_k\|_F^2. \quad (2)$$

where $\lambda > 0$ and $\kappa > 0$ are scalar constants, $\overline{\mathbf{X}}_k$ denotes the complementary data matrix of $\mathbf{X}_k$ from the whole training samples $\mathbf{X}$. To avoid the trivial solution of $\mathbf{P}_k = \mathbf{0}$, an extra constraint term $\|\mathbf{D}_k\|_F^2$ is added.

The introduction of $\mathbf{W}_k$ is to improve the localization performance. To this end, we assign higher weights to the samples with better localization. By this way, lower reconstruction residual will be expected for sample with higher weight, and thus the reconstruction residual can be adopted to find the proposal with better localization. Therefore, we use the IoU with the ground truth bounding box of the k-th object category to define $\mathbf{W}_k$.

3.1.2 Learning of the Dictionary Pair

To exploit the alternating minimization algorithm for dictionary pair learning, we relax Eqn. (2) by introducing a coding coefficient matrix $\mathbf{A}$:

$$\{\mathbf{P}^*, \mathbf{A}^*, \mathbf{D}^*\} = \arg\min_{\mathbf{P},\mathbf{A},\mathbf{D}} \sum_{k=0}^{K} \|(\mathbf{X}_k - \mathbf{D}_k\mathbf{A}_k)\mathbf{W_k}\|_F^2 + \tau\|(\mathbf{P}_k\mathbf{X}_k - \mathbf{A}_k)\mathbf{W_k}\|_F^2 + \lambda\|\mathbf{P}_k\overline{\mathbf{X}}_k\|_F^2 + \kappa\|\mathbf{D}_k\|_F^2, \quad (3)$$

where τ is a scalar constant. All terms in the above objective function are characterized by the squared Frobenius norm, and thus Eqn. (3) can be efficiently solved by alternating minimization.

By initializing $\mathbf{P}$ and $\mathbf{D}$ with random matrices with unit Frobenius norm, the minimization for Eqn. (3) can be performed by alternating between the following three steps:

(i) Fix $\{\mathbf{D}, \mathbf{P}, \mathbf{X}\}$, and update $\mathbf{A}$ via:

$$\mathbf{A}_k^* = (\mathbf{D}_k^T\mathbf{D}_k + \tau\mathbf{I})^{-1}(\tau\mathbf{P}_k\mathbf{X}_k + \mathbf{D}_k^T\mathbf{X}_k). \quad (4)$$

(ii) Fix $\{\mathbf{D}, \mathbf{A}, \mathbf{X}\}$, and update $\mathbf{P}$ via:

$$\mathbf{P}_k^* = \tau\mathbf{A}_k\mathbf{W}_k\mathbf{W}_k^T\mathbf{X}_k^T(\tau\mathbf{X}_k\mathbf{W}_k\mathbf{W}_k^T\mathbf{X}_k^T + \lambda\overline{\mathbf{X}}_k\overline{\mathbf{X}}_k^T + \gamma\mathbf{I})^{-1}, \quad (5)$$

where the constant γ is empirically set as 0.0001 according to the validation set.

(iii) Fix $\{\mathbf{A}, \mathbf{P}, \mathbf{X}\}$, update $\mathbf{D}$ via:

$$\mathbf{D}_k^* = \mathbf{X}_k\mathbf{W}_k\mathbf{W}_k^T\mathbf{A}_k^T(\mathbf{A}_k\mathbf{W}_k\mathbf{W}_k^T\mathbf{A}_k^T + \kappa\mathbf{I})^{-1}. \quad (6)$$

Since all steps have closed-form solutions for $\{\mathbf{A}, \mathbf{P}, \mathbf{D}\}$, the 3-step minimization is quite efficient. We stop the iteration when the difference between the energy in two adjacent iterations is less than a threshold (e.g., 0.01).

3.2. Dictionary Pair Back Propagation

In this section, we propose a dictionary pair back propagation (DPBP) algorithm for joint learning of DPCL and CNN parameters in an end-to-end manner.

The dictionary pair $(\mathbf{D}_k, \mathbf{P}_k)$ of the DPCL model can be optimized separately, and thus Eqn. (2) can be decomposed into the following $K+1$ sub-problems:

$$\arg\min_{\mathbf{P}_k,\mathbf{D}_k} L_k(\mathbf{P}_k, \mathbf{D}_k) = \arg\min_{\mathbf{P}_k,\mathbf{D}_k} \|(\mathbf{X}_k - \mathbf{D}_k\mathbf{P}_k\mathbf{X}_k)\mathbf{W_k}\|_F^2 + \lambda\|\mathbf{P}_k\overline{\mathbf{X}}_k\|_F^2 + \kappa\|\mathbf{D}_k\|_F^2. \quad (7)$$

In DPBP, the partial derivatives with respect to $\{\mathbf{P}_k, \mathbf{D}_k\}$ are defined as:

$$\frac{\partial L_k(\mathbf{P}_k, \mathbf{D}_k)}{\partial \mathbf{P}_k} = -2\mathbf{D}_k(\mathbf{I} - \mathbf{D}_k\mathbf{P}_k)\mathbf{X}_k\mathbf{W}_k\mathbf{W}_k^T\mathbf{X}_k^T + 2\lambda\mathbf{P}_k\overline{\mathbf{X}}_k\overline{\mathbf{X}}_k^T$$
$$\frac{\partial L_k(\mathbf{P}_k, \mathbf{D}_k)}{\partial \mathbf{D}_k} = -2\mathbf{X}_k\mathbf{W}_k(I - \mathbf{D}_k\mathbf{P}_k)\mathbf{W}_k^T\mathbf{X}_k^T\mathbf{P}_k^T + 2\kappa\mathbf{D}_k \quad (8)$$

With $L = \sum_{k=0}^{K} L_k$, the partial derivatives with respect to $\mathbf{X}_k$ is then defined as:

$$\frac{\partial L}{\partial \mathbf{X}_k} = 2(\mathbf{I} - \mathbf{P}_k^T \mathbf{D}_k^T)(\mathbf{X}_k - \mathbf{D}_k \mathbf{P}_k \mathbf{X}_k)\mathbf{W}_k \mathbf{W}_k^T + \sum_{k' \neq k} 2\lambda \mathbf{P}_{k'}^T \mathbf{P}_{k'} \mathbf{X}_k \tag{9}$$

Once all $\frac{\partial L}{\partial \mathbf{X}_k}$ are obtained, we can perform the standard back propagation [20] to update the CNN parameters.

3.3. Object Detection on Test Image

Given a proposal I from the test image, we first extract the CNN feature $\mathbf{x}$ from I, and then define the reconstruction residual for the k-th category as:

$$\mathcal{L}(\mathbf{x}; \mathbf{D}_k, \mathbf{P}_k) = \|\mathbf{x} - \mathbf{D}_k \mathbf{P}_k \mathbf{x}\|_F^2. \tag{10}$$

The classification rule of the DPCL is

$$y = \arg\min_i \mathcal{L}(\mathbf{x}; \mathbf{D}_i, \mathbf{P}_i). \tag{11}$$

When $y \neq 0$, we further use bounding box regression to refine the location of the object location.

4. Optimization with Multi-Task Loss

4.1. Multi-Task Loss

The proposed DPCL is a category classification method and is not conformable with localization task. To improve localization, Girshick [12] adopted a multi-task loss to balance classification and localization. In this method, each proposal is classified into either background or one of the object categories, which may not work well in distinguishing background from object categories. To address this issue, we further decompose the classification task into two related ones. As illustrated in Fig. 1, the feature extracted by the CNN layers is duplicated and simultaneously fed into two DPCLs: the objectness DPCL layer and the categoryness DPCL layer. The former estimates the score for being an object, while the latter computes the scores for being a specific object category.

Objectness Dictionary Pair Classifier Layer. The objectness usually is defined as the score of covering objects of any category. For the purpose of measuring the objectness of the input region, the proposed Objectness Dictionary Pair (ODP) layer applies two dictionary pairs $\{\mathbf{D}_o, \mathbf{P}_o\}$ and $\{\mathbf{D}_b, \mathbf{P}_b\}$ to represent objects of any category and background, respectively. If a region feature $\mathbf{x}$ can be better represented by the background dictionary pairs $\{\mathbf{D}_b, \mathbf{P}_b\}$, it is very unlikely to have objects inside. Rather than directly identify the background according to Eqn. (11), ODP calculates objectness of the input feature $\mathbf{x}$ for further object detection in a soft way: a threshold $\mathcal{T}$ is used to distinguish the region with large background. With the reconstruction residual defined in Eqn. (10), the objectness score $Q(x)$ for the feature $\mathbf{x}$ of the input region is defined as:

$$Q(\mathbf{x}) = \begin{cases} 1 - \frac{\mathcal{L}(\mathbf{x}; \mathbf{D}_o, \mathbf{P}_o)}{\sum_{i \in \{o,b\}} \mathcal{L}(\mathbf{x}; \mathbf{D}_i, \mathbf{P}_i)}; & \frac{\mathcal{L}(\mathbf{x}; \mathbf{D}_o, \mathbf{P}_o)}{\mathcal{L}(\mathbf{x}, \mathbf{D}_b, \mathbf{P}_b)} < \mathcal{T}; \\ 0, & otherwise, \end{cases} \tag{12}$$

where $\mathcal{T}$ controls the precision and recall rate of detecting background (larger $\mathcal{T}$ results in higher precision and lower recall rate), and is empirically set as 0.5 according to the validation set. Thus, our model is able to identify the background based on whether $Q(\mathbf{x})$ is 0 or not.

Categoryness Dictionary Pair Classifier Layer. The categoryness score $S(\mathbf{x}, k)$ denotes the likeliness that the feature $\mathbf{x}$ belongs to the k-th category. In order to compute the categoryness for object detection, the proposed Categoryness Dictionary Pair (CDP) layer consists of K dictionary pairs, where K is the number of object categories. Once the feature $\mathbf{x}$ of the input region is fed, CDP will encode $\mathbf{x}$ over the K category-specific dictionary pairs $\{\mathbf{D}_k, \mathbf{P}_k\}$ and output the reconstruction residual for each dictionary pair. We define the categoryness $S(\mathbf{x}, k)$ using the reconstruction residual as:

$$S(\mathbf{x}, k) = 1 - \frac{\mathcal{L}(\mathbf{x}; \mathbf{D}_k, \mathbf{P}_k) \cdot e^{\beta \mathcal{L}(\mathbf{x}; \mathbf{D}_k, \mathbf{P}_k)}}{\sum_{i=1}^{K} \mathcal{L}(\mathbf{x}; \mathbf{D}_i, \mathbf{P}_i) \cdot e^{\beta \mathcal{L}(\mathbf{x}; \mathbf{D}_i, \mathbf{P}_i)}}, \tag{13}$$

where the constant β is empirically set as 0.003 according to the validation set.

Then, the product rule is used to fuse objectness score and categoryness score, and the classification score $\mathcal{F}_k$ that $\mathbf{x}$ belongs to the k-th category is defined as:

$$\mathcal{F}_k(\mathbf{x}) = S(\mathbf{x}, k) * Q(\mathbf{x}). \tag{14}$$

Let ϕ denote the function of the CNN layers and I_i denote the input region with the category label y_i, we have the feature $\mathbf{x} = \phi(I, \omega)$. With the classification score $\mathcal{F}_k$, the final classification loss is defined as:

$$\begin{aligned} L_{cls}(I) = \sum_{k=0}^{K} \mathbf{1}(y = k) \log \mathcal{F}_k(\phi(I, \omega)) \\ +(1 - \mathbf{1}(y = k)) \log(1 - \mathcal{F}_k(\phi(I, \omega))) \\ +R\{\omega, \mathbf{D}, \mathbf{P}\}, \end{aligned} \tag{15}$$

where $\mathbf{1} \in \{0, 1\}$ is the indicator function, and $R\{\omega, \mathbf{D}, \mathbf{P}\}$ denotes the regularization term on the parameters of CNN and two DPCLs.

Bounding Box Regression Loss. Our defined multi-task loss can easily append other correlated loss, e.g., the robust loss in [12]. Let $t^k(I) = (t_x^k, t_y^k, t_w^k, t_h^k)$ and $t^*(I) = (t_x^*, t_y^*, t_w^*, t_h^*)$ be the predicted and ground truth bounding boxes of the proposal I, where k denotes that the proposal I belongs to the k-th object category. Then, the bounding box regression loss is defined as:

$$L_{loc}(t^k(I), t^*(I)) = \sum_{i \in x,y,w,h} H_1(t_i^k - t_i^*), \tag{16}$$

Algorithm 1 Multi-Task CNN+DPCL Learning

Input:

Training samples $\mathbf{I} = [\mathbf{I}_1, \mathbf{I}_2, ..., \mathbf{I}_K, \mathbf{I}_b]$ for $K+1$ classes ($\mathbf{I}_b$ denotes background), pre-trained CNN layers' parameters.

Output:

Dictionary pairs $\{\mathbf{D}, \mathbf{P}\} = \{\{\mathbf{D}_1, \mathbf{P}_1\}, \{\mathbf{D}_2, \mathbf{P}_2\}, ..., \{\mathbf{D}_K, \mathbf{P}_K\}, \{\mathbf{D}_o, \mathbf{P}_o\}, \{\mathbf{D}_b, \mathbf{P}_b\}\}$, fine-tuned CNN parameter ω, bounding box regressors ω_r.

Initialization:

1. Initialize CNN parameters ω with pre-trained network;
2. Obtain output features $\mathbf{x}_i = \phi(I_i; \omega)$ for all training samples;
3. Regard $\phi(I_b; \omega)$ as background samples and the other $\phi(I_o; \omega) = [\phi(I_1; \omega), \phi(I_2; \omega), ..., \phi(I_K; \omega)]$ as object samples;
5. Optimize dictionary pair $\{\mathbf{D}, \mathbf{P}\}$ as described in Sect. 3.1.2:
   i. Given $\phi(I_b; \omega)$ and $\phi(I_o; \omega)$, train $\{\mathbf{D}_o, \mathbf{P}_o\}, \{\mathbf{D}_b, \mathbf{P}_b\}$;
   ii. Given $\phi(I_k; \omega)$, train $\{\mathbf{D}_k, \mathbf{P}_k\}, k = 1, ..., K$.

repeat

6. Fine-tune $\{\mathbf{D}, \mathbf{P}, \omega, \omega_r\}$ via mini-batch based back propagation on L_{mt};

until Eqn.(18) converges.

where $H_1(z)$ is the Huber loss

$$H_1(z) = \begin{cases} 0.5x^2, & \text{if } |z| < 1 \\ |z| - 0.5, & \text{otherwise} \end{cases}, \tag{17}$$

which is robust to outliers.

We can adopt the sum rule in [12] to combine L_{cls} and L_{loc}, and the multi-task loss is defined as:

$$L_{mt} = -\frac{1}{N}\Big(\sum_{i=1}^{N} L_{cls}(I_i) + p_i^* L_{loc}(t^k(I_i), t^*(I_i))\Big), \tag{18}$$

where p_i^* is an indicator to denote whether the proposal I_i is an object.

4.2. Optimization

After obtaining the partial derivatives of L_{mt} with respect to $\mathbf{D}_b$, $\mathbf{P}_b$, $\mathbf{D}_o$, $\mathbf{P}_o$, $\mathbf{D}_k$, $\mathbf{P}_k$, $\mathbf{X}_k$, we can extend DPBP to fine-tune CNN+DPCL to update the dictionary pairs, CNN parameters and bounding box regressors. To optimize L_{mt}, we initialize the CNN parameters with some pre-trained network, *e.g.*, AlexNet [19] or VGG [27], and initialize the dictionary pairs using the dictionary pair learning algorithm in Sect. 3.1.2. Then the DPBP algorithm is adopted to further optimize CNN+DPCL in an end-to-end manner. We summarize the whole learning procedure as Alg. 1.

4.3. Inference

The inference task is to predict the detection scores and bounding box to a given image region I. Formally, we perform forward propagation to output the CNN feature $\phi(I, \omega)$ of the region, and then feed it into the ODP layer and the CDP layer, simultaneously. With the learned dictionary pairs, we calculate reconstruction residuals of the feature via Eqn. (10), and obtain the objectness $Q(\phi(I, \omega))$ via Eqn. (12) as well as the categoryness $S(\phi(I, \omega))$ for each category via Eqn. (13). Finally, our model outputs the final object detection score via Eqn. (14) for each object category. If $Q(\phi(I, \omega)) > 0$, we further use the bounding box regressors to predict the object location.

5. Experiments

We demonstrate the performance of our proposed joint feature and DPCL learning framework on serveral object detection benchmarks. The experiments are conducted on the commonly used Pascal VOC 2007/2012 datasets [7]. During evaluation, we adopt the PASCAL Challenge protocol: a correct detection should has more than 0.5 IoU with the ground truth bounding-box. The performance is evaluated by mean Average Precision (mAP).

5.1. Parameter Setting

In all experiments, we set $\{\tau, \lambda, \kappa, \beta, \gamma, \mathcal{T}, m\}$ as $\{0.01, 0.01, 0.001, 0.003, 0.0001, 0.5, 64\}$. We consider R-CNN [13] with AlexNet [19] / VGG [27] and FRCN with VGG [27] as the baseline model. Following the same experiment settings in [13], the employed CNN parameters are firstly pretrained on ImageNet, and then fine-tuned on the corresponding VOC training and validation sets by stochastic gradient descent (SGD) with a 21-way softmax loss (20 object categories plus one background). Then we replace the softmax classification layer with our proposed model, and fine-tune the network via DPBP with learning rate starting at 0.00001 and momentum beginning at 0.9. During the fine-tuning, all regions with < 0.5 IoU overlap with a ground-truth bounding box are treated as background, while those with ≥ 0.5 IoU are considered as positives for the corresponding object category. The weight of these positives is defined as the IoU with the ground truth bounding box. For instance, if a region has 0.6 IoU with the ground-truth bounding box from the cat category, then it is a positive sample with the weight 0.6 for the further dictionary pair learning of the cat category.

5.2. Results and Comparisons

We denote by R-CNN(Alex/VGG) [13] and FRCN [12] the used CNN frameworks, by ODP and CDP the proposed objectiveness and categoryness dictionary pair layers, and by BB the Bounding Box regression in the R-CNN framework. From Tab. 2, BB regression can consistently achieve $3\% \sim 4\%$ performance gain by mAP. Therefore, we have included BB regression for all the results listed in the Tab. 3$\sim$5, and the comparison is fair. Our full implemented model with the proposed DPBP in AlexNet [19] is then denoted as "R-CNN(Alex) + CDP + ODP". Other variants of our implementation are represented similarly.

In Tab. 2, we report in detail the accuracy on all object categories of VOC 2007, compared with the meth-

Method	data	mAP	areo	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
DPM [8]	07	33.7	33.2	60.3	10.2	16.1	27.3	54.3	58.2	23.0	20.0	24.1	26.7	12.7	58.1	48.2	43.2	12.0	21.1	36.1	46.0	43.5
SS [31]	07	33.7	43.5	46.5	10.4	12.0	9.3	49.4	53.7	39.4	12.5	36.9	42.2	26.4	47.0	52.4	23.5	12.1	29.9	36.3	42.2	48.8
Regionlet [34]	07	41.7	54.2	52.0	20.3	24.0	20.1	55.5	68.7	42.6	19.2	44.2	49.1	26.6	57.0	54.5	43.4	16.4	36.6	37.7	59.4	52.3
DetNet [30]	07	30.5	29.2	35.2	19.4	16.7	3.7	53.2	50.2	27.2	10.2	34.8	30.2	28.2	46.6	41.7	26.2	10.3	32.8	26.8	39.8	47.0
R-CNN(Alex) [13]	07	54.2	64.2	69.7	50.0	41.9	32.0	62.6	71.0	60.7	32.7	58.5	46.5	56.1	60.6	66.8	54.2	31.5	52.8	48.9	57.9	64.7
SPP(Alex) [16]	07	55.2	**65.5**	65.9	51.7	38.4	32.7	62.6	68.6	**69.7**	33.1	**66.6**	**53.1**	**58.2**	63.6	**68.8**	50.4	27.4	53.7	**48.2**	**61.7**	64.7
Best approach of [32]	07	46.9	49.3	69.5	31.9	28.7	40.4	61.5	61.5	41.5	25.5	44.5	47.8	32.0	**67.5**	61.8	46.7	25.9	40.5	46.0	57.1	58.2
R-CNN(Alex)+ODP+CDP	07	**57.5**	64.8	**71.5**	**54.6**	**46.1**	**50.7**	**68.9**	**78.2**	56.9	**36.2**	58.3	47.1	51.2	**67.5**	67.8	**66.0**	**34.7**	**61.5**	42.8	58.7	**66.7**
R-CNN(VGG) [13]	07	62.2	71.6	73.5	58.1	42.2	39.4	70.7	76.0	74.5	38.7	71.0	56.9	74.5	67.9	69.6	59.3	35.7	62.1	64.0	66.5	71.2
R-CNN(VGG)+BB [13]	07	66.0	73.4	77.0	63.4	45.4	44.6	75.1	78.1	**79.8**	40.5	73.7	62.2	**79.4**	78.1	73.1	64.2	35.6	66.8	**67.2**	70.4	71.1
Best approach of [38] with BB	07	68.5	74.1	**83.2**	**67.0**	50.8	51.6	76.2	81.4	77.2	**48.1**	**78.9**	**65.6**	77.3	78.4	**75.1**	70.1	41.4	**69.6**	60.8	70.2	**73.7**
R-CNN(VGG)+ODP+CDP	07	65.7	71.0	76.5	62.8	49.5	58.3	76.9	81.1	74.0	43.5	72.5	58.1	71.8	75.2	74.0	70.2	**42.2**	65.2	56.9	64.9	70.0
R-CNN(VGG)+ODP+CDP+BB	07	**68.6**	**75.0**	79.3	65.3	**52.8**	**60.9**	**80.2**	**81.7**	77.0	45.2	75.5	62.5	76.1	**80.3**	74.8	**71.7**	42.1	68.1	59.4	**72.3**	71.7

Table 2. Test set mAP for VOC 2007. The entries with the best APs for each object category are bold-faced. Training data key: "07": VOC07 trainval.

Method	data	mAP	areo	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
FRCN	07	66.9	74.5	78.3	69.2	53.2	36.6	77.3	78.2	82.0	40.7	72.7	67.9	79.6	79.2	73.0	69.0	30.1	65.4	70.2	75.8	65.8
FRCN+CDP+ODP	07	71.1	78.6	78.9	70.4	57.8	47.7	83.1	82.5	84.2	51.8	75.7	69.1	80.4	82.1	77.7	76.3	42.4	69.0	68.6	77.6	67.3
FRCN	07+12	70.0	77.0	78.1	69.3	59.4	38.3	81.6	78.6	86.7	42.8	78.8	68.9	84.7	82.0	76.6	69.9	31.8	70.1	74.8	80.4	70.4
FRCN+CDP+ODP	07+12	73.4	79.6	80.0	70.6	65.1	50.0	86.1	85.4	84.1	54.2	79.5	71.5	82.0	83.9	79.3	77.1	44.6	69.2	74.1	83.3	69.2

Table 3. Test set mAP for VOC 2007. The entries with the best APs for each object category are bold-faced. Training data key: "07": VOC07 trainval, "07+12": VOC07 trainval union with VOC12 trainval, "07++12": VOC07 trainval and test with VOC12 trainval.

Method	data	mAP	areo	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
FRCN	12	65.7	80.3	74.7	66.9	46.9	37.7	73.9	68.6	87.7	41.7	71.1	51.1	86.0	77.8	79.8	69.8	32.1	65.5	63.8	76.4	61.7
FRCN+CDP+ODP	12	66.9	81.0	75.1	69.8	50.0	43.6	73.4	71.0	87.7	44.4	69.6	54.4	85.5	77.2	77.8	72.0	37.1	66.4	58.8	77.0	65.5
FRCN	07++12	68.4	82.3	**78.4**	70.8	52.3	38.7	**77.8**	71.6	89.3	44.2	73.0	55.0	**87.5**	**80.5**	**80.8**	72.0	35.1	**68.3**	65.7	**80.4**	64.2
FRCN+CDP+ODP	07++12	**69.7**	**82.4**	78.1	**72.1**	**55.7**	**44.3**	77.6	**73.3**	**89.4**	**49.1**	**73.4**	**56.3**	87.0	80.2	79.5	**74.4**	**40.9**	67.0	**66.9**	78.9	**67.3**

Table 4. Test set mAP for VOC 2012. The entries with the best APs for each object category are bold-faced. Training data key: "12": VOC07 trainval, "07++12": VOC07 trainval and test with VOC12 trainval.

ods based on hand-crafted feature engineering [8, 34, 31] and deep CNNs [30, 13, 16, 38, 32]. Given the same CNN and region proposals (AlexNet), our proposed model, R-CNN(Alex)+ODP+CDP, achieves the mAP of 57.5%, distinctly superior to the other six competing methods, i.e., 33.7% [8], 33.7% [31], 41.7% [34], 46.9% [32], 30.5% [30], 54.2% [13] and 55.2% [16]. Given another network architecture VGG, our R-CNN(VGG)+ODP+CDP model obtains 65.7%/68.6% mAP with/without bounding box regression, and is comparable with the recently published state-of-the-art method [38].

Tab. 3 and Tab. 4 demonstrate that our proposed model FRCN+CDP+ODP can consistently achieve 3% and 2% performance gain over FRCN on VOC 2007 and VOC 2012 datasets, and also justify that our model is robust to different CNN frameworks.

Fig. 2 demonstrates some object detection examples obtained by the proposed method and FRCN. Thanks to the use of reconstruction residual in both objectness estimation and categoryness calculation, the selected bounding box with optimal detection score by our method has fewer background, as shown in Fig. 2 (a). When one bounding box covers more background, its reconstruction residual over the objectness dictionary pairs will be larger, resulting in a lower detection score. Thanks to the divide-and-conquer manner brought by ODP and CDP, our model can recognize more objects (Fig. 2 (b)) in the image with better accuracy (Fig. 2 (c)).

(a) (b) (c)

Figure 2. Detection examples from PASCAL VOC 2007 dataset. The boxes and category labels obtained by baseline FRCN is in green, and those obtained by the proposed structure model is in red.

5.3. Analysis

For further evaluation, we conduct the following three empirical analysis under different settings. To show the advantages of the proposed model, we directly employ the pre-

Method	data	mAP	areo	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
FRCN	07+12	70.0	77.0	78.1	69.3	59.4	38.3	81.6	78.6	86.7	42.8	78.8	68.9	84.7	82.0	76.6	69.9	31.8	70.1	**74.8**	80.4	70.4
FRCN+softmax+ODP	07+12	72.3	81.3	79.7	71.6	**65.4**	47.9	**86.6**	84.0	85.6	48.4	78.6	70.2	80.4	82.9	77.6	70.8	43.7	69.8	71.3	81.8	69.6
FRCN+CDP (w/o FT)	07+12	70.2	74.7	76.1	68.3	60.2	43.6	79.8	79.1	82.7	50.5	77.3	69.9	83.9	81.0	72.0	68.8	37.4	**73.3**	71.8	77.2	**76.7**
FRCN+CDP	07+12	70.9	78.1	78.8	70.1	57.8	47.8	84.1	82.9	84.1	51.7	75.2	67.5	79.7	82.3	77.0	76.2	42.5	68.2	68.4	77.6	67.7
FRCN+CDP+ODP (w/o FT)	07+12	72.4	**83.0**	**83.4**	**77.1**	56.1	42.7	83.5	72.8	**90.5**	52.5	73.3	62.0	**90.0**	81.8	**85.1**	69.1	44.0	71.2	73.6	**85.1**	72.0
FRCN+CDP+ODP (w/o weights)	07+12	71.1	78.9	79.0	70.5	59.0	47.2	83.7	82.7	84.7	51.5	75.3	69.1	80.3	82.4	77.6	76.2	41.8	67.8	69.4	77.3	67.7
FRCN+CDP+ODP	07+12	**73.4**	79.6	80.0	70.6	65.1	**50.0**	86.1	**85.4**	84.1	**54.2**	**79.5**	**71.5**	82.0	**83.9**	79.3	**77.1**	**44.6**	69.2	74.1	83.3	69.2

Table 5. Test set mAP for VOC 2007. The entries with the best APs for each object category are bold-faced. "07+12": VOC07 trainval union with VOC12 trainval.

trained CNN models (VGG [27]) under the FRCN framework, and perform component analysis on the VOC 2007 dataset.

(I) We demonstrate the effectiveness of incorporating objectness estimation into our model for object detection. That is, we discard the ODP in our model, and train it only with the CDP via DPBP. Note that, the number of dictionary pairs is now 21 (20 object categories plus background). We denote the model without ODP as "FRCN+CDP". Similarly, we introduce ODP into FRCN and keep its softmax layer and denote this scheme as "FRCN+softmax+ODP". In FRCN+softmax+ODP, we directly adopt the FRCN model fine-tuned on PASCAL VOC 07+12, which is provided by the authors. Based on its feature representation, we train an extra ODP classifier, and use the original softmax classifier to replace CDP for object detection. As Tab. 5 reports, FRCN+softmax+ODP achieves 2.3% performance gain. "FRCN+CDP" drops by 1.0% the performance. This is because there are too many background samples to achieve fine level representation of objects. Hence, owe to detecting objects in a divide-and-conquer strategy, ODP makes great contributions to improve the detection accuracy.

(II) To clarify the significance of the proposed DPBP for network fine-tuning, we directly replace the softmax layer of FRCN by the proposed ODP and CDP layers. We denote these models as "FRCN+ODP+CDP (w/o FT)" and "FRCN+CDP (w/o FT)". The results demonstrate that fine-tuning can obtain about 1.0% performance gain.

(III) To demonstrate the effectiveness of predefined weights for training samples, we set all weights to 1 in our model. That is, the training samples have the same weights during the dictionary pair training inside ODP and CDP. We denote this model as "Ours (w/o weights)", and compare it with the original version "Ours (full)". As illustrated in Fig. 3, the test error rate of "Ours (w/o weights)" is much higher than "Ours (full)" after 10,000 iterations. The reason is that the category-specific dictionary pair is introduced to represent the parts of other category or background inside the training examples. By means of regarding the IoU as the predefined weights of training samples, this phenomenon can be suppressed. From Fig. 3, one can see that the introduced weights can make the training phase stable and achieve a lower error rate. Tab. 5 also demonstrates that

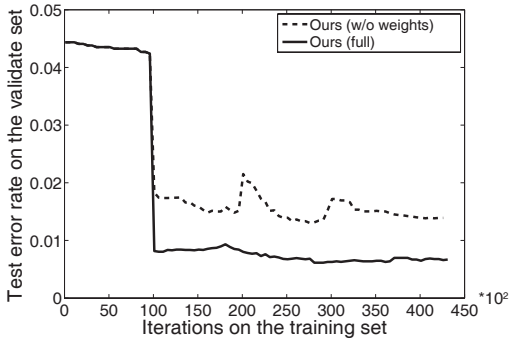

Figure 3. Test error rates with/without weighted training samples in the deep model. The solid curve represents our full model, and the dashed curve represents our model without using weights.

"FRCN+ODP+CDP" with weights can improve about 2% mAP, compared with "FRCN+ODP+CDP (w/o weights)".

Acknowledgment

This work was supported in part by the Hong Kong Polyutechnic University's Joint Supervision Scheme with the Chinese Mainland, Taiwan and Macao Universities (Grant no. G-SB20). This work was also supported in part by the Guangdong Natural Science Foundation under Grant S2013050014548 and 2014A030313201, in part by Special Program for Applied Research on Super Computation of the NSFC-Guangdong Joint Fund (the second phase), and in part by the Fundamental Research Funds for the Central Universities.

6. Conclusion

In this paper, we presented dictionary pair classifier-driven CNNs for object detection, where dictionary pair back propagation (DPBP) is proposed for the end-to-end learning of dictionary pair classifiers and CNN representation, and sample weighting is adopted to improve the localization performance. Furthermore, a multi-task loss is suggested for joint training of the DPCLs and bounding-box regressor. Experiments demonstrated the superiority of the proposed framework. In the future, we will apply our model with other powerful CNNs to improve detection accuracy.

References

[1] P. Agrawal, R. Girshick, and J. Malik. Analyzing the performance of multilayer neural networks for object recognition. In *ECCV*, 2014.

[2] B. Alexe, T. Deselaers, and V. Ferrari. Measuring the objectness of image windows. *TPAMI*, 2010.

[3] P. Arbeláez, J. Pont-Tuset, J. Barron, F. Marques, and J. Malik. Multiscale combinatorial grouping. In *CVPR*, 2014.

[4] R. Caruana. Multitask learning. *Machine learning*, 28:41–75, 1997.

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

[6] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov. Scalable object detection using deep neural networks. In *CVPR*, June 2014.

[7] M. Everingham, L. V. Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge 2007 (voc2007) results, 2007.

[8] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *TPAMI*, 2010.

[9] Z. Feng, M. Yang, L. Zhang, Y. Liu, and D. Zhang. Joint discriminative dimensionality reduction and dictionary learning for face recognition. *Pattern Recognition*, 46:2134–2143, 2013.

[10] S. Gao, I.-H. Tsang, and Y. Ma. Learning category-specific dictionary and shared dictionary for fine-grained image categorization. *Image Processing, IEEE Transactions on*, 23(2):623–634, Feb 2014.

[11] S. Gidaris and N. Komodakis. Object detection via a multi-region & semantic segmentation-aware cnn model. In *ICCV*, 2015.

[12] R. Girshick. Fast r-cnn. In *ICCV*, 2015.

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

[14] R. Girshick, F. Iandola, T. Darrell, and J. Malik. Deformable part models are convolutional neural networks. In *CVPR*, 2015.

[15] S. Gu, L. Zhang, W. Zuo, and X. Feng. Projective dictionary pair learning for pattern classification. In *Advances in neural information processing systems*, 2014.

[16] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *TPAMI*, 2015.

[17] J. Hoffman, S. Guadarrama, E. S. Tzeng, R. Hu, J. Donahue, R. Girshick, T. Darrell, and K. Saenko. Lsda: Large scale detection through adaptation. In *Advances in Neural Information Processing Systems*, pages 3536–3544, 2014.

[18] Z. Jiang, Z. Lin, and L. Davis. Label consistent k-svd: learning a discriminative dictionary for recognition. *TPAMI*, 34, 2013.

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, 2012.

[20] Y. LeCun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard, and L. Jackel. Handwritten digit recognition with a back-propagation network. In *Advances in neural information processing systems*, 1990.

[21] L. Lin, P. Luo, X. Chen, and K. Zeng. Representing and recognizing objects with massive local image patches. *Pattern Recognition*, 45:231–240, 2012.

[22] J. Mairal, F. Bach, and J. Ponce. Task-driven dictionary learning. *TPAMI*, 35, 2012.

[23] I. Ramirez, P. Sprechmann, and G. Sapiro. Classification and clustering via dictionary learning with structured incoherence and shared features. In *CVPR*, 2010.

[24] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 2015.

[25] S. Ren, K. He, R. Girshick, X. Zhang, and J. Sun. Object detection networks on convolutional feature maps. In *arXiv:1504.06066 [cs.CV]*, 2015.

[26] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. In *ICLR 2014*, April 2014.

[27] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

[28] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*, 2014.

[29] C. Szegedy, S. Reed, D. Erhan, and D. Anguelov. Scalable, high-quality object detection. *CoRR*, abs/1412.1441, 2014.

[30] C. Szegedy, A. Toshev, and D. Erhan. Deep neural networks for object detection. In *Advances in neural information processing systems*, 2013.

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

[32] L. Wan, D. Eigen, and R. Fergus. End-to-end integration of a convolution network, deformable parts model and non-maximum suppression. In *CVPR*, 2015.

[33] X. Wang, L. Lin, L. Huang, and S. Yan. Incorporating structural alternatives and sharing into hierarchy for multiclass object recognition and detection. In *CVPR*, 2013.

[34] X. Wang, M. Yang, S. Zhu, and Y. Lin. Regionlets for generic object detection. In *ICCV*, 2013.

[35] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *TPAMI*, 31, 2008.

[36] M. Yang, L. Zhang, X. Feng, and D. Zhang. Sparse representation based fisher discrimination dictionary learning for image classification. *IJCV*, 109, 2014.

[37] L. Zhang, M. Yang, and X. Feng. Sparse representation or collaborative representation: Which helps face recognition? In *ICCV*, 2011.

[38] Y. Zhang, K. Sohn, R. Villegas, G. Pan, and H. Lee. Improving object detection with deep convolutional networks via bayesian optimization and structured prediction. In *CVPR*, 2015.

[39] Y. Zhu, R. Urtasun, R. Salakhutdinov, and S. Fidler. segdeepm: Exploiting segmentation and context in deep neural networks for object detection. In *CVPR*, June 2015.