WELDON: Weakly Supervised Learning of Deep Convolutional Neural Networks

Thibaut Durand, Nicolas Thome, Matthieu Cord

WELDON: Weakly Supervised Learning of Deep Convolutional Neural Networks

Thibaut Durand, Nicolas Thome, Matthieu Cord
Sorbonne Universités, UPMC Univ Paris 06, CNRS, LIP6 UMR 7606, 4 place Jussieu, 75005 Paris
{thibaut.durand, nicolas.thome, matthieu.cord}@lip6.fr

Abstract

In this paper, we introduce a novel framework for WEakly supervised Learning of Deep cOnvolutional neural Networks (WELDON). Our method is dedicated to automatically selecting relevant image regions from weak annotations, e.g. *global image labels, and encompasses the following contributions. Firstly, WELDON leverages recent improvements on the Multiple Instance Learning paradigm,* i.e. *negative evidence scoring and top instance selection. Secondly, the deep CNN is trained to optimize Average Precision, and fine-tuned on the target dataset with efficient computations due to convolutional feature sharing. A thorough experimental validation shows that WELDON outperforms state-of-the-art results on six different datasets.*

1. Introduction

Over the last few years, deep learning and Convolutional Neural Networks (CNN) [22] have become state-of-the-art methods for various visual recognition tasks, *e.g.* image classification or object detection. To overcome the limited invariance capacity of CNN, bounding box annotations are often used [33, 16]. However, these rich annotations rapidly become costly to obtain [6], making the development of Weakly Supervised Learning (WSL) models appealing.

Recently, there have been some attempts for WSL training of deep CNNs [34, 36]. In this context, image annotations consist in global labels, and the training objective is to localize image regions which are the most relevant for classification. In computer vision, the dominant approach for WSL is the Multiple Instance Learning (MIL) paradigm [9]: an image is considered as a bag of regions, and the model seeks the `max` scoring instance in each bag [35, 41, 37, 5, 47, 44, 11]. Recently, relaxations of standard MIL assumptions have been introduced in the context of Latent SVM models and shallow architectures [27, 38, 10], showing improved recognition performances on various object and scene datasets.

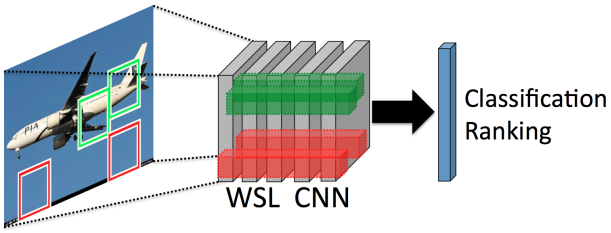

Figure 1. The WELDON model is a deep CNN trained in a weakly supervised manner. To perform image prediction, *e.g.* classification or ranking, WELDON automatically selects multiple positive (green) + negative (red) evidences on several regions in the image.

In this paper, we propose a new model for WEakly supervised Learning of Deep cOnvolutional neural Networks (WELDON), which is illustrated in Figure 1. WELDON is trained to automatically select relevant regions from images annotated with a global label, and to perform end-to-end learning of a deep CNN from the selected regions. The ultimate goal is image classification (or ranking). We call this setting weakly-supervised, because the localization step only exploits global labels.

Regarding WSL, WELDON is dedicated to selecting two types of regions, adapted from [27, 10, 38] to deep networks: green regions in Figure 1 correspond to areas with top scores, *i.e.* regions which best support the presence of the global label. On the contrary, red regions incorporate negative evidence for the class, *i.e.* are the lowest scoring areas. Our deep WSL model is detailed in section 3.

Regarding training, the model parameters are optimized using back-propagation with standard classification losses, but we also adapt the learning to structured output ranking. We design a network architecture which enables fast region feature computation by convolutional sharing. The network is initialized from deep features trained on ImageNet, and the parameters are fine-tuned on the target dataset.

2. Related Works & Contributions

The computer vision community is currently witnessing a revolutionary change, essentially caused by Convolutional Neural Networks (CNN) and deep learning. Beyond the

outstanding success reached in the context of large scale classification (ImageNet) [22], deep features also prove to be very effective for transfer learning: state-of-the-art results on standard benchmarks are nowadays obtained with deep features as input. Recent studies reveal that performances can further be improved by collecting large datasets that are semantically closer to the target domain [54], or by fine-tuning the network with data augmentation [7].

Despite their excellent performances, current CNN architectures only carry limited invariance properties: although a small amount of shift invariance is built into the models through subsampling (pooling) layers, strong invariance is generally not dealt with [53]. Recently, attempts have been made to overcome this limitation. Some methods revisit the BoW model with deep features as local region activations [19, 18] or designed BoW layers [2]. The drawback of these models is that background regions are encoded into the final representation, decreasing its discriminative power. Another option to gain strong invariance is to explicitly align image regions, *e.g.* by using Weakly Supervised Learning (WSL) models.

In the computer vision community, WSL has been predominantly addressed through the Multiple Instance Learning (MIL) paradigm [9]. In standard MIL modeling, an image is regarded as a bag of instances (regions), and there is an asymmetric relationship between bag and instance labels: a bag is positive if it contains at least one positive instance, and negative if all its instances are negative - *i.e.* the Negative instances in Negative bags (NiN) hypothesis. MIL models thus perform image prediction through its `max` scoring region. The Deformable Part Model (DPM) [14] is an instantiation of the MI-SVM model [1] for MIL, which is extremely popular for WSL due to its excellent performances for object detection. Extensive works have therefore used DPM and its generalization to structured output prediction, LSSVM [50], for weakly supervised scene recognition and object localization [23, 35, 41, 5, 42, 37, 21, 47]. Contrarily to these methods built upon handcrafted features, *e.g.* BoW models [46, 39, 3, 17] or biologically-inspired models [43, 49, 48], recent approaches tackle the problem of WSL training of deep CNNs, *e.g.* [34, 36], incorporating a `max` CNN layer accounting for the MIL hypothesis.

Recently, interesting MIL extensions have been introduced in [51, 24, 27, 38, 10]. All these methods use a bag prediction strategy which departs from the standard `max` scoring function in MIL, especially due to the relaxation of the common Negative instances in Negative bags (NiN) MIL assumption. In the Learning with Label Proportion (LLP) framework [51], only label ratios between $\oplus/\ominus$ instances in bags are provided during training. In [24], the LLP method of [51] is explicitly applied to MIL problems, in the context of video event detection. LLP is shown to outperform baseline methods (mi/MI-SVM [1]), especially by its capacity to relax the NiN assumption. In [27], the authors question the NiN assumption by claiming that it is often violated in practice during image annotation: human rather label images based on their dominant concept than on the actual presence of the concept in each subregion. To support the dominant concept annotation, the authors in [27] introduce a prediction function selecting the top scoring instances in each bag. Other approaches departs from the NiN assumption by tracking negative evidence of a class with regions [38, 10]: for example, a cow detector should strongly penalize the prediction of the bedroom class. In [38], the authors introduce a WSL learning formulation specific to multi-class classification, where negative evidence is explicitly encoded by augmenting model parameters to represent the positive/negative contribution of a part to a class. In [10], the idea of negative evidence is formalized by the introduction of a generic structured output latent variable, where the prediction function is extended from `max` to `max+min` region scores. The `min` scoring region accounts for the concept of negative evidence, and is capitalized on for learning a more robust model.

Many computer vision tasks are evaluated with ranking metrics, *e.g.* Average Precision (AP). In the WSL setting, this is, however, a very challenging problem: for example, no algorithm exists for solving the loss-augmented inference problem with Latent Structural SVM [50]. In [4], LAPSVM is introduced, enabling a tractable optimization by defining an *ad-hoc* prediction rule dedicated to ranking. In [10], the proposed ranking model offers the ability to solve loss-augmented inference with an elegant symmetrization due to the `max+min` prediction function.

In this paper, we introduce a new model for WSL training of deep CNNs, which takes advantage of recent MIL extensions. The approach the most closely connected to ours is [34], which we extend at several levels. Our submission therefore encompasses the following contributions:

- We improve the deep WSL modeling in [34] by incorporating top instance [27] and negative evidence [38, 10] insights into our deep prediction function. Contrarily to [38, 10, 27], we propose an end-to-end training of deep CNNs.
- We improve deep WSL training in [34] by introducing a specific architecture design which enable an easy and effective transfer learning and fine-tuning. In addition, we adapt our training scheme to explicitly optimize over ranking metrics, *e.g.* AP.
- We report excellent performances, outperforming state-of-the-art results on six challenging datasets. A systematic evaluation of our modeling and training contributions highlights their importance for training deep CNN models from weak annotations.

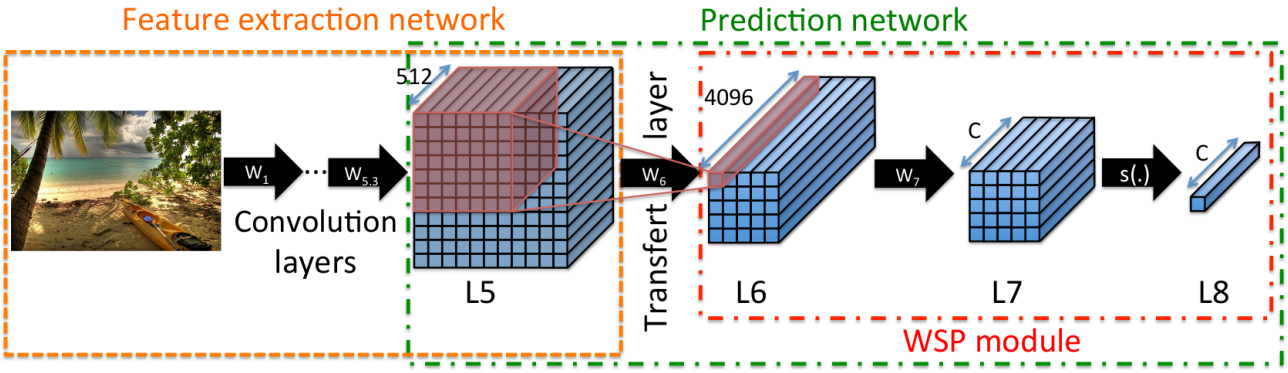

Figure 2. WELDON deep architecture: our model is composed of 2 sub-networks. The feature extraction net outputs a fixed-size vector for any region in the image, using a multi-scale sliding window mechanism. The prediction net is composed of a transfer layer with weights $\mathbf{W_6}$, which enables using networks pre-trained on large-scale datasets for model initialization, and a Weakly-Supervised Prediction (WSP) module, which is the main point studied in this submission. In the proposed WSP module, the spatial aggregation function s combines improvements on the MIL modeling, *i.e.* top-instance scoring and negative evidence into the training of the full deep CNN.

3. WELDON Model

The proposed WELDON model is decomposed into two sub-networks: a deep feature extraction net and a prediction net, as illustrated in Figure 2. The feature extraction net purpose is to extract a fixed-size deep descriptor for each region in the image, while the prediction net outputs a structured output for the whole image. We firstly detail the prediction network, since the main paper contributions are incorporated at this level, mainly by the introduction of novel methods for weakly supervised learning of deep CNNs.

3.1. Prediction network design

The prediction net acts on the **L5** layer, which is a set of $d(=512)$ feature maps with $n \times n$ ($n \geq 7$) spatial neurons. **L5** is computed by the feature extraction net (Section 3.2).

a) Transfer layer The first layer of the prediction network transforms the **L5** layer into a layer **L6** of size $n' \times n' \times d'$[1] ($d' = 4096$), as illustrated in Figure 2. This convolutional layer is composed of filters $\mathbf{W_6}$, each of size $7 \times 7 \times d$. Note that each 7×7 area in **L5** in thus mapped to a fixed-size d'-dimensional vector, so that this transfer layer is equivalent to applying the whole CNN on each of the 7×7 region. This architecture design serves two purposes: fast feature computation in regions (see Section 3.2), and transferring $\mathbf{W_6}$ weights from large scale datasets (see Section 4).

b) Weakly-Supervised Prediction (WSP) module This is the heart of the proposed method, and is dedicated to selecting relevant regions for properly predicting the global (structured) label associated to each training image.

The WSP module consists in a succession of two layers. The first layer is a linear prediction model $\mathbf{W_7}$, which is dedicated to providing a (structured output) prediction for each of the $n' \times n'$ spatial cell in **L6**. This corresponds to a fully connected layer applied to each spatial cell in **L6**, which we implement using 1×1 convolutions, as in [29]. The **L7** layer is thus of size $n' \times n' \times C$, where C is the size of the structured prediction map, *e.g.* C is the number of classes for multi-class classification (we detail our structured output instantiations in Section 4).

The second layer of the WSP module is a spatial pooling layer **s**, which aggregates, for each output $c \in \{1; C\}$, the score over the $n' \times n'$ regions into a single scalar value. This give the final prediction layer **L8**. As mentioned in Section 2, the standard approach for WSL inherited from MIL is to select the `max` scoring region. We propose to improve this strategy in two complementary directions.

i) Top instances Based on recent MIL insights on learning with top instances [27], we propose to extend the selection of a single region to multiple high scoring regions.

Formally, let us denote as $h_i \in \{0, 1\}$ the binary variable denoting the selection of the i^{th} region from layer **L7**, and $l^7_{i,c}$ the value of the i^{th} region score for output (*e.g.* class) c.We propose the following aggregation strategy s_{top}, which selects the k highest scoring regions as follows:

$$s_{top}\left(\mathbf{L7}\right) = \max_{\mathbf{h}} \sum_{i=1}^{n'^2} h_i \cdot \mathbf{l_i^7}, \text{ s.t. } \sum_{i=1}^{n'^2} h_i = k \qquad (1)$$

where $\mathbf{h} = \{h_i\}, i \in \left\{1; n'^2\right\}$, and $\mathbf{l_i^7} = \left\{l^7_{i,c}\right\}, c \in \{1; C\}$. Beyond the relaxation of the NiN assumption, which is sometimes inappropriate (see Section 2), the intuition behind s_{top} is to provide a more robust region selection strategy. Indeed, using a single area for training the model necessarily increases the risk of selecting outliers, guiding the training of the deep CNN towards bad local minima.

[1] $n' = n - 6$ because of the $\mathbf{W_6}$ filter padding.

ii) MinMax layer When using top instances in Eq (1) for classifying images, we make use of the most informative regions. Recent studies show that this information can be effectively combined with negative evidence for a class, *e.g.* using regions which best support the absence of the class [38, 10]. In this submission, we propose to incorporate this negative evidence in our prediction layer using multiple instances, in the same way as for top instances. Therefore, we augment our aggregation strategy with the term s_{low}, which selects the m lowest-scoring regions in an image:

$$s_{low}\left(\mathbf{L7}\right) = \min_{\mathbf{h}} \sum_{i=1}^{n'^2} h_i \cdot \mathbf{l7_i}, \text{ s.t. } \sum_{i=1}^{n'^2} h_i = m \qquad (2)$$

The final prediction of the network, that we denote as $\mathbf{L8}$, simply consists in summing s_{top} and s_{low}. If we denote as t_c^* (resp. l_c^*) the k top (resp. m lowest) instances selected for output c, the c^{th} output feature $\mathbf{L8}(c)$ is:

$$\mathbf{L8}(c) = s_{top}\left(\mathbf{L7}(c)\right) + s_{low}\left(\mathbf{L7}(c)\right) = \sum_{t_c^*=1}^{k} \mathbf{l7}_{t_c^*} + \sum_{l_c^*=1}^{m} \mathbf{l7}_{l_c^*} \qquad (3)$$

The proposed WSP aggregation scheme in Eq. (3) thus generalizes the `max+min` prediction function in [10] in the case of multiple top positive/negative instances.

3.2. Feature extraction network design

The feature extraction network is dedicated to computing a fixed-size representation for any region of the input image. When using CNNs as feature extractors, the most naive option is to process input regions independently, *i.e.* to resize each region to match the size of a full image for CNN architectures trained on large scale databases such as ImageNet (*e.g.* 224×224). This is the approach followed in R-CNN [16], or in MANTRA [10]. This is, however, highly inefficient since feature computation in (close) neighbor regions is not shared. Recent improvements in SPP nets [19] or fast R-CNN [15] process images of any size by using only convolutional/pooling layers of CNNs trained on ImageNet, subsequently applying max pooling to map each region into a fixed-size vector. Fully-convolutional networks are also used for semantic segmentation [8, 31].

We propose here a different strategy, which is based on a multi-scale sliding window scheme. In the proposed architecture, input images at a given scale are rescaled to a constant size IxI, with $I \geq 224$. For all I, we consider regions of size 224×224 pixels, so that the region scale is $\alpha = 224/I$ (see details in Table 1 of supplementary 1). Input images are processed with the fully convolutional/pooling layers of CNNs trained on ImageNet, leading to $\mathbf{L5}$ layers of different sizes.

Our multi-scale strategy is close to that of [34], but the region size is designed to fit a 224×224 pixel area (*i.e.* 7×7 in $\mathbf{L5}$ layer), which is not the case in [34]. This is a crucial difference, which enables the weights $\mathbf{W6}$ to the first prediction layer $\mathbf{L6}$ in Figure 2 to be transferred from ImageNet, which is capitalized on for defining a training strategy robust to over-fitting, see Section 4.2. We now detail the training of our deep WSL architecture.

4. Training the WELDON Model

As shown in Figure 2, the WELDON model outputs $\mathbf{L8} \in \mathbb{R}^C$. This vector represents a structured output, which can be used in a multi-class or multi-label classification framework, but also in a ranking problem formulation.

4.1. Training formulation

In this paper, we consider three different structured prediction for WELDON, and their associated loss functions during training.

Multi-class classification In this simple case, C is the number of classes. We use the usual soft-max activation function on top of $\mathbf{L8}$: $P(\mathbf{L8}(c)) = e^{\mathbf{L8}(c)} / \sum_{c'} e^{\mathbf{L8}(c')}$, with its corresponding log loss during training.

Multi-label classification In the case of multiple labels, we use a one-against-all strategy, as [34]. For C different classes, we train the C binary classifiers jointly, using logistic regression for prediction $P(\mathbf{L8}(c)) = \left(1 + e^{-\mathbf{L8}(c)}\right)^{-1}$, with its associated log loss[2].

Ranking: Average Precision We also tackle the problem of optimizing ranking metrics, and especially Average Precision (AP) with our WELDON model. We use a latent structured output ranking formulation, following [52]: our input is a set of N training images $\mathbf{x} = \{x_i\}$, $i \in \{1; N\}$, with their binary labels y_i, and our goal is to predict a ranking matrix $\mathbf{c} \in \mathcal{C}$ of size $N \times N$ providing an ordering of the training examples (our ranking feature map is detailed supplementary 2.1, Eq (1)). Here, we explicitly denote the output $\mathbf{L8}(\mathbf{x}, \mathbf{c})$ to highlight the dependence on $\mathbf{x}$.

During training, we aim at minimizing the following loss: $\Delta_{ap}(\mathbf{c}^*, \mathbf{c}) = 1 - AP(\mathbf{c}^*, \mathbf{c})$, where $\mathbf{c}^*$ is the ground-truth ranking. Since AP is non-smooth, we define the following surrogate (upper-bound) loss:

$$\ell_{\mathbf{W}}(\mathbf{x}, \mathbf{c}^*) = \max_{\mathbf{c} \in \mathcal{C}} \left[\Delta_{ap}(\mathbf{c}^*, \mathbf{c}) + \mathbf{L8}(\mathbf{x}, \mathbf{c}) - \mathbf{L8}(\mathbf{x}, \mathbf{c}^*)\right] \qquad (4)$$

The maximization in Eq (4) is generally referred to as Loss-Augmented Inference (LAI), while inference consists computing $\hat{\mathbf{c}}(\mathbf{x}) = \arg\max_{\mathbf{c} \in \mathcal{C}} \mathbf{L8}(\mathbf{x}, \mathbf{c})$. Exhaustive maximization is intractable due to the huge size of the structured

[2] Experimentally, hinge loss with linear prediction performs similarly.

output space. The problem is even exacerbated in the WSL setting, see [4, 10]. We exhibit here the following result for WELDON (proof in supplementary 2.2):

Proposition 1 *For each training example, let us denote* $s(i) = s_{top}(\mathbf{W_7 L6^i}) + s_{low}(\mathbf{W_7 L6^i})$ *in Eq (3). Inference and LAI for the WELDON ranking model can be solved exactly by sorting examples in descending order of score* $s(i)$.

Proposition 1 shows that the optimization over regions, *i.e.* score $s(i)$, decouples from the maximization over output variables c. This reduces inference and LAI optimization to fully supervised problems. Inference solution directly corresponds to $s(i)$ sorting. For solving LAI with AP loss Δ_{ap} in Eq (4), we use the exact greedy algorithm of [52][3].

4.2. Optimization

Given the loss functions given in Section 4.1, WELDON parameters are adjusted using gradient-based methods.

For multi-class and multi-label predictions, error gradients in $\mathbf{L8}$ are well-known. For the ranking instantiation, we have (details in supplementary 3):

$$\frac{\partial \ell}{\partial \mathbf{W_7}} = \frac{\partial \mathbf{L8}(\mathbf{x}, \tilde{\mathbf{c}})}{\partial \mathbf{W_7}} - \frac{\partial \mathbf{L8}(\mathbf{x}, \mathbf{c}^*)}{\partial \mathbf{W_7}}$$

where $\tilde{\mathbf{c}}$ is the LAI solution. In all cases, error gradient is back-propagated in the deep CNN through chain rule.

Transfer learning & fine-tuning Similarly to other deep WSL models, our whole CNN contains a lot of parameters. The vast majority of weights is located in $\mathbf{W_6}$ (Figure 2), which contains $\sim 10^8$ parameters. Training such huge models on medium-size datasets as those studied in this paper (with $[10^3$-$10^5]$ examples) is highly prone to over-fitting.

With a network even bigger than ours, the authors in [34] address this issue by extensively using regularization during training with dropout and data-augmentation. We propose here to couple these regularization strategies with a two-step learning procedure to limit over-fitting.

In a first training phase, all parameters except those of the WSP prediction module, *i.e.* $\mathbf{W_7}$, are frozen. All other parameters, *i.e.* convolutional layers and $\mathbf{W_6}$ are transferred from CNNs trained on large-scale datasets (ImageNet). Note that the transfer for $\mathbf{W_6}$ is fully effective thanks to the carefully designed architecture of our feature extraction network (Section 3.2) and the transfer layer (Section 3.1a)). It is, for example, not possible as it with the architecture in [34]. Note that $\mathbf{W_7}$ only contains $\sim 10^4$ parameters, and can therefore robustly be optimized in the considered medium-size datasets.

In a second training phase, starting with $\mathbf{W_7}$ initialized from the first phase, a fine-tuning of all other CNN parameters is achieved. We use dropmap as regularization strategy, consisting in randomly freezing maps in $\mathbf{L_6}$.

5. Experiments

Our deep CNN architecture is based on VGG16 [45]. We implement our model using Torch7 (http://torch.ch/)[4].

We evaluate our WELDON strategy on several Computer Vision benchmarks corresponding to various visual recognition tasks. While some choose pre-trained deep features according to the target task (like Places features for Scene recognition [54]), we knowingly decide with WELDON to use only deep features pre-trained on ImageNet whatever the visual recognition task. This is to put to the proof our claim about genericity of our deep architecture.

Absolute comparison with state-of-the-art methods is provided in Section 5.1, while Section 5.2 analyzes the impact of the different improvements introduced in Section 3 and 4 for training deep WSL CNNs.

Experimental Setup In order to get results in very different recognition contexts, 6 datasets are used: object recognition (Pascal VOC 2007 [12], Pascal VOC 2012 [13]), scene categorization (MIT67 [40] and 15 Scene [26]), and visual recognition, where context plays an important role (COCO [30], Pascal VOC 2012 Action [13]).

For MIT67, 15 Scene and VOC 2007, performances are evaluated following the standard protocol. For VOC 2012, evaluation is carried out on the *val* set (which does not require server evaluation). On COCO dataset, we follow the protocol in [34], and perform classification experiments. On Pascal VOC 2012 Action, we use the same weakly supervised protocol as in [10], with evaluation on the *val* set.

5.1. Overall comparison

Firstly, we compare the proposed WELDON model to state-of-the-art methods. We use the multi-scale WSL model described in Section 3.2, and scale combination is performed using an Object-Bank [28] strategy. For the selection of top/low instances, we use here the default setting of $k = m = 3$ (Eq (1) and Eq (2) in Section 3.1), for scale $\alpha \leq 70\%$ (Table 1 of supplementary 1). This parameter is analyzed in Section 5.2, showing further improvements by careful tuning. Results for object (resp. scene and context) datasets are gathered in Table 1 (resp. Table 2 and Table 3).

For object datasets, we can show in Table 1 that WELDON outperforms all recent methods based on deep features by a large margin. More specifically, the improvement compared to deep features computed on the whole image [7, 45] is significant: there is an improvement over

[3] Faster (approximate) methods, *e.g.* [32], could also be used.

[4] We will make our code publicly available if accepted.

	VOC 2007	VOC 2012
Return Devil [7]	82.4	
VGG16 (online code) [45]	84.5	82.8
SPP net [19]	82.4	
Deep WSL MIL [34]		81.8
MANTRA [10]	85.8	
WELDON	**90.2**	**88.5**

Table 1. mAP results on object recognition datasets. WELDON and state-of-the-art methods results are reported.

the best method [45] of $\sim$ 6 pt on both datasets. Note that since we use deep features VGG16 from [45], the performance gain directly measures the relevance of using a WSL method, which selects localized evidence for performing prediction, rather than relying on the whole image information. Compared to SPP net [19], the improvement of $\sim$ 8 pt on VOC 2007 highlights the superiority of region selection based on supervised information, rather than using handcrafted aggregation with spatial-pooling BoW models. The most important comparison is the improvement over other recent WSL methods on deep features [34, 10]. Compared to [10], the improvement of 4.4 pt on VOC 2007 essentially shows the importance of using multiple instances, and the relevance of an end-to-end training of a deep CNN in the target dataset. We also outperform the deep WSL CNN in [34], the approach which is the most closely connected to ours, by 6.7 pt on VOC 2012. This big improvement illustrates the positive impact of incorporating MIL relaxations for WSL training of deep CNNs, *i.e*. negative evidence scoring and top-instance selection. Finally, we can point out the outstanding score reached by WELDON on VOC 2007, exceeding the nominal score of 90%.

	15 Scene	MIT67
CaffeNet ImageNet [20]	84.2	56.8
CaffeNet Places [54]	90.2	68.2
VGG16 (online code) [45]	91.2	69.9
MOP CNN [18]		68.9
MANTRA [10]	93.3	76.6
Negative parts [38]		77.1
WELDON (OB)	**94.3**	**78.0**

Table 2. Multiclass accuracy results on scene categorization datasets. WELDON and state-of-the-art methods results are reported.

The results shown in Table 2 for scene recognition also illustrate the big improvement of WELDON compared to deep features computed on the whole image [20, 54, 45] and MOP CNN [18], a BoW method pooling deep features with VLAD. It is worth noticing that WELDON also outperforms recent part-based methods including negative evidence during training [10, 38]. This shows the improvement brought out by the end-to-end deep WSL CNN training with WELDON. Note that in these scene datasets, deep features trained on Places [54] reach much better results than those trained on ImageNet. Therefore, we can expect further performance improvement with WELDON by using stronger feature as input for transfer, before fine-tuning the network to the target dataset.

In Table 3, we show the results in datasets where contextual information is important for performing prediction. On VOC 2012 action and COCO, selecting the regions corresponding to objects or parts directly related to the class is important, but contextual features are also strongly related to the decision. WELDON outperforms VGG16 [45] by $\sim$ 8 pt on both datasets, again validating our WSL deep method in this context. On COCO, the improvement is from 62.8% [34] to 68.8% for WELDON. This shows the importance of the negative evidence and top-instance scoring in our WSP module, which better help to capture contextual information than the standard MIL `max` function used in [34]. Finally, note that the very good results in COCO also illustrate the efficiency of the proposed WSL training of deep CNN with WELDON, which is able to deal with this large datasets (80 classes and $\sim$ 80000 training examples).

	VOC 2012 action	COCO
VGG16 (online code) [45]	67.1	59.7
Deep WSL MIL [34]		62.8
WELDON	**75.0**	**68.8**

Table 3. WELDON results and comparison to state-of- the-art methods on context datasets.

5.2. WELDON Analysis

In this section, we analyze the impact on prediction performances of the different contributions of WELDON given in Section 3 and 4. Our baseline model a) is the WSL CNN model using an aggregation function s=`max` at the WSP module stage (Figure 2), evaluated at scale $\alpha = 30\%$. It gives a network similar to [34], trained at a single scale. To measure the importance of the difference between WELDON and a), we perform a systematic evaluation on the performance when the following variations are incorporated:

b) Use of k top instances instead of the `max`. We use $k = 3$.

c) Incorporation of negative evidence through `max+min` aggregation function. When b)+c) are combined, we use m lowest-instances instead of the `min`, with $m = 3$.

d) Learning the deep WSL with ranking loss, *e.g*. AP, in the concerned datasets (PASCAL VOC).

e) Fine-tuning the network on the target dataset, *i.e*. using the second training phase in Section 4.2.

The results are reported in Table 4 for object and context datasets with AP evaluation (VOC 2007 and VOC 2012 action), and in Table 5 for scene datasets.

a) `max`	b) +top	c) +`min`	d) +AP	VOC07	VOC act
✓				83.6	53.5
✓	✓			86.3	62.6
✓		✓		87.5	68.4
✓		✓	✓	88.4	71.7
✓	✓	✓		87.8	69.8
✓	✓	✓	✓	88.9	72.6

Table 4. Systematic evaluation of our WSL deep CNN contributions. Object and Context databases with AP evaluation.

a) `max`	b) +top	c) +`min`	d) +FT	MIT67	15-Scene
✓				42.3	72.0
✓	✓			69.5	85.9
✓		✓		72.1	89.7
✓	✓	✓		74.5	90.9
✓	✓	✓	✓	75.1	91.5

Table 5. Systematic evaluation of our WSL deep CNN contributions. Scene databases with multi-class classification evaluation. FT: fine-tuning.

From this systematic evaluation, we can draw the following conclusions:

- Both b) and c) improvements result in a very large performance gain on all datasets, with a comparable impact on performances: $\sim$ +30 pt on MIT67, $\sim$ +15 pt on 15-Scene, $\sim$ +15 pt on VOC 2012 Action and$\sim$ +4 pt on VOC 2007. When looking more accurately, we can notice that `max+min` leads always to a larger improvement, *e.g.* is 4 pt above on 15-Scene or VOC 2012 Action and 3 pt on MIT67.

- Combining b) and c) improvements further boost performances: +3 pt on MIT67 and VOC 2012 Action, +2 pt on 15-Scene, +1pt on VOC 2007. This shows the complementarity of these two extensions at the aggregation level. We perform an additional experiment for comparing b)+c) and c), by setting the same number of regions (*e.g.* 6 for k-`max` and 3-3 for k-m `max+min`). It turns out that k-m `max+min` is the best method for various k/m values, showing that negative evidence contains significant information for visual prediction.

- Minimizing an AP loss enables to further improve performances. Interestingly, the same level of improvement is observed when AP optimizing is added to the c) configuration than to the more powerful b)+c) configuration: +3pt on VOC 2012 Action, +1 pt on VOC 2007. This shows that b) and c) are conditionally independent from the AP optimization.

- Fine-tuning favorably impacts performances, with +0.6 pt gain on MIT67 and 15-Scene. Note that the performance level is already high at the b)+c) configuration, making further improvements challenging. These results are obtained with the two-step fine-tuning proposed in section 4.2. We compare this strategy to a parallel optimization, consisting in jointly updating all network parameters. Performances drop with this parallel procedure, *e.g.* 73.5% on MIT67.

To further evaluate the impact of the number k top and m low instances, we show in Figure 3 the performance variation ($k = m$) on MIT67 and 15 Scene. We can see that performances can still be significantly improved on these datasets when k and m increase, although performances decrease for $k \geq 8$ on MIT67 (see results in other datasets on supplementary 4).

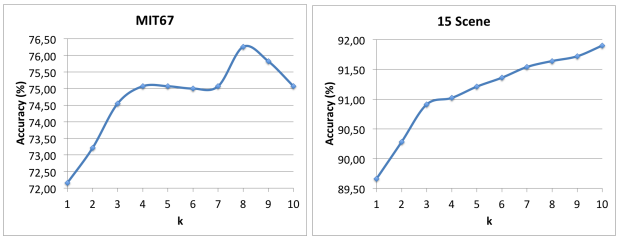

Figure 3. Multi-class accuracy with respect to the number of top/low instances for MIT67 and 15 Scene at scale $\alpha = 30\%$.

Finally, we show in Figure 4 the performance in different configurations, corresponding to sequentially adding the previous improvements in the following order: a), a)+b), b)+c), and b)+c)+d) for VOC 2007 / VOC 2012 / VOC 2012 Action and c)+c)+e) for MIT67 and 15 Scene. On all dataset, we can see the very large improvement from configuration a) to configuration b)+c)+d)/e). The behavior can, however, be different among datasets: for example, the performance boost is sharp from a) to a)+b) on MIT67 (the following improvements being less pronounced), whereas there is a linear increase from a) b)+c)+d) on VOC 2007 and VOC 2012.

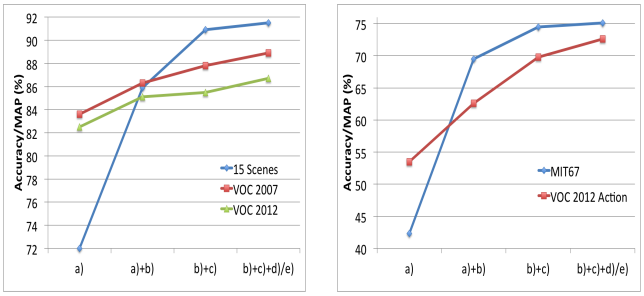

Figure 4. Performance variations when the different improvements are incorporated: from the baseline model a) to b), a)+b), b)+c), and b)+c)+d)/e).

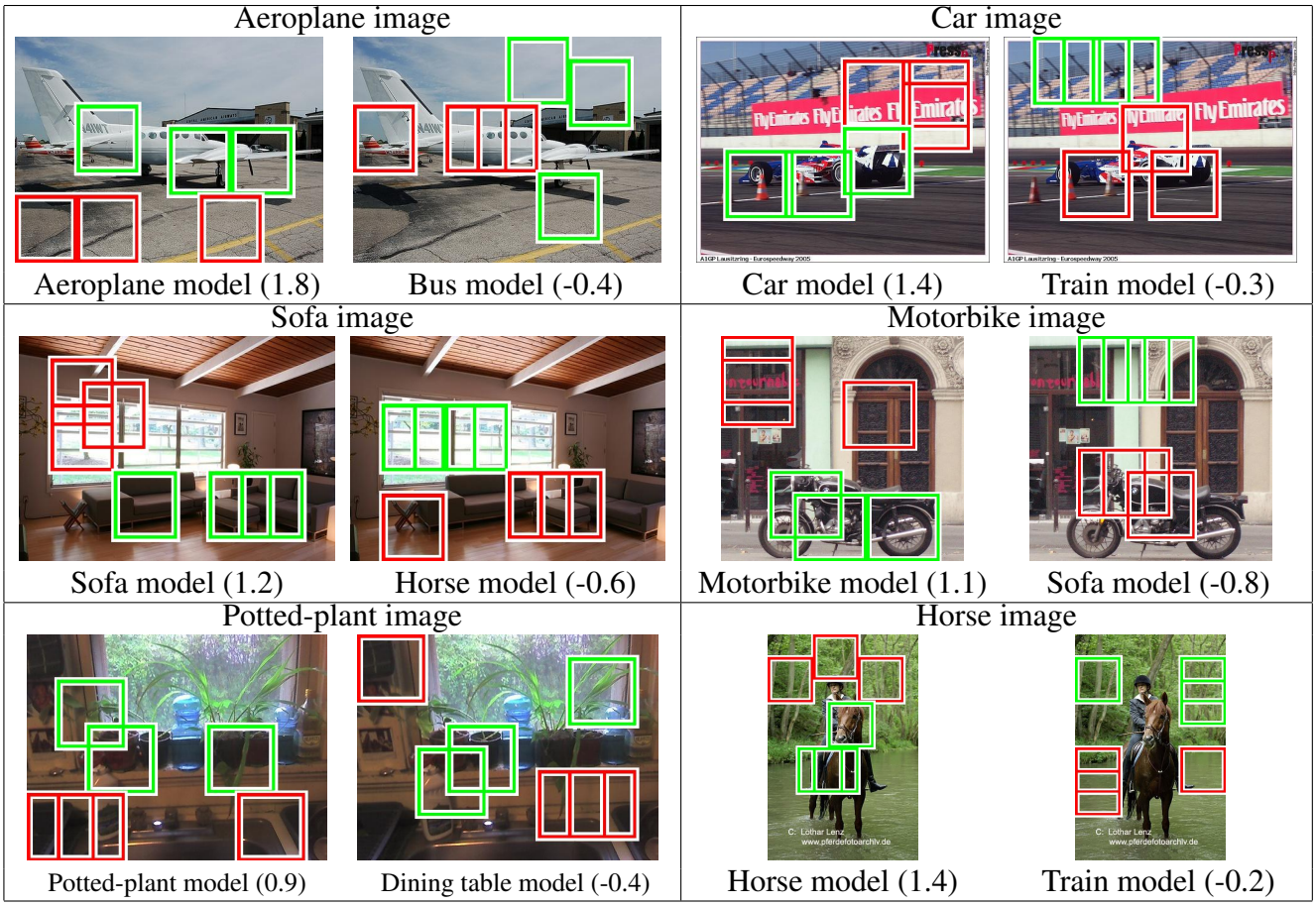

Figure 5. Visual results of WELDON on VOC 2007 with $k = m = 3$ instances. The green (resp. red) boxes are the 3 top (resp. 3 low) instances. For each image, the first column represents WELDON prediction for the ground truth classifier (with its corresponding score), and the second column shows prediction and score for an incorrect classifier.

Qualititative analysis of region selection To illustrate the region selection policy performed by WELDON, we show in Figure 5 the top 3 positive (resp. top 3 negative) regions selected by the model in green (resp. red), on the VOC 2007 dataset. We show the results for the ground truth classification model in the first column, with its associated prediction score. We can notice that top positive green regions detect several discriminant parts related to the object class, potentially capturing several instances or modalities (*e.g.* wheels or airfoil for the car model), whereas negative evidence on red regions, which should remain small, encode contextual information (*e.g.* road or sky for airplane, or trees for horse). The region selection results are shown for incorrect classification models in the second column, again with the prediction score. We can notice that red regions correspond to multiple negative evidence for the class, *e.g.* parts of coach strongly penalizes the prediction of the class horse, or seat or handlebar negatively supports the prediction of the sofa category.

6. Conclusion

In this paper, we introduce WELDON, a new method for training deep CNNs in a weakly supervised manner. Our method exploits to the full extend deep CNN strategy in multiple instance learning framework to efficiently deal with weak supervision. The whole architecture is carefully designed for fast processing by sharing region feature computations, and robust training.

We show the excellent performances of WELDON for WSL prediction on very different visual recognition tasks: object class recognition, scene classification, and images with a strong context, outperforming state-of-the-art results on six challenging datasets. Future works include adapting WELDON for other structured visual applications, *e.g.* metric learning [25], semantic segmentation.

Acknowledgments This research was supported by a DGA-MRIS scholarship.

References

[1] S. Andrews, I. Tsochantaridis, and T. Hofmann. Support vector machines for multiple-instance learning. In *NIPS*, 2003. 2

[2] R. Arandjelović, P. Gronat, A. Torii, T. Pajdla, and J. Sivic. NetVLAD: CNN architecture for weakly supervised place recognition. In *CVPR*, 2016. 2

[3] S. Avila, N. Thome, M. Cord, E. Valle, and A. Araujo. Pooling in image representation: the visual codeword point of view. *Computer Vision and Image Understanding*, 2012. 2

[4] A. Behl, C. V. Jawahar, and M. P. Kumar. Optimizing average precision using weakly supervised data. In *CVPR*, 2014. 2, 5

[5] H. Bilen, V. Namboodiri, and L. Van Gool. Object classification with latent window parameters. In *IJCV*, 2013. 1, 2

[6] M. Blaschko, P. Kumar, and B. Taskar. Tutorial: Visual learning with weak supervision, CVPR 2013. 1

[7] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. In *BMVC*, 2014. 2, 5, 6

[8] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. 2015. 4

[9] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artif. Intell.*, 1997. 1, 2

[10] T. Durand, N. Thome, and M. Cord. MANTRA: Minimum Maximum Latent Structural SVM for Image Classification and Ranking. In *ICCV*, 2015. 1, 2, 4, 5, 6

[11] T. Durand, N. Thome, M. Cord, and D. Picard. Incremental learning of latent structural svm for weakly supervised image classification. In *ICIP*, 2014. 1

[12] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html. 5

[13] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html. 5

[14] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *PAMI*, 2010. 2

[15] R. Girshick. Fast R-CNN. In *ICCV*, 2015. 4

[16] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. 1, 4

[17] H. Goh, N. Thome, M. Cord, and J.-H. Lim. Learning Deep Hierarchical Visual Feature Coding. *IEEE Transactions on Neural Networks and Learning Systems*, 2014. 2

[18] Y. Gong, L. Wang, R. Guo, and S. Lazebnik. Multi-scale orderless pooling of deep convolutional activation features. In *ECCV*, 2014. 2, 6

[19] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In *ECCV*, 2014. 2, 4, 6

[20] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *ACM International Conference on Multimedia*, 2014. 6

[21] M. Juneja, A. Vedaldi, C. V. Jawahar, and A. Zisserman. Blocks that shout: Distinctive parts for scene classification. In *CVPR*, 2013. 2

[22] A. Krizhevsky, I. Sutskever, and G. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*. 2012. 1, 2

[23] P. Kumar, B. Packer, and D. Koller. Self-paced learning for latent variable models. In *NIPS*, 2010. 2

[24] K.-T. Lai, F. X. Yu, M.-S. Chen, and S.-F. Chang. Video event detection by inferring temporal instance labels. In *CVPR*, 2014. 2

[25] M. T. Law, N. Thome, and M. Cord. Fantope regularization in metric learning. In *CVPR*, 2014. 8

[26] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, 2006. 5

[27] W. Li and N. Vasconcelos. Multiple instance learning for soft bags via top instances. In *CVPR*, 2015. 1, 2, 3

[28] E. P. X. Li-Jia Li, Hao Su and L. Fei-Fei. Object bank: A high-level image representation for scene classification & semantic feature sparsification. In *NIPS*, 2010. 5

[29] M. Lin, Q. Chen, and S. Yan. Network in network. In *ICLR*, 2014. 3

[30] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *ECCV*, Zürich, September 2014. 5

[31] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. *CVPR*, 2015. 4

[32] P. Mohapatra, C. Jawahar, and M. P. Kumar. Efficient optimization for average precision svm. In *NIPS*. 2014. 5

[33] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *CVPR*, 2014. 1

[34] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Is object localization for free? weakly-supervised learning with convolutional neural networks. In *CVPR*, 2015. 1, 2, 4, 5, 6

[35] M. Pandey and S. Lazebnik. Scene recognition and weakly supervised object localization with deformable part-based models. In *ICCV*, 2011. 1, 2

[36] G. Papandreou, I. Kokkinos, and P.-A. Savalle. Modeling local and global deformations in deep learning: Epitomic convolution, multiple instance learning, and sliding window detection. In *CVPR*, 2015. 1, 2

[37] S. N. Parizi, J. G. Oberlin, and P. F. Felzenszwalb. Reconfigurable models for scene recognition. In *CVPR*, 2012. 1, 2

[38] S. N. Parizi, A. Vedaldi, A. Zisserman, and P. F. Felzenszwalb. Automatic discovery and optimization of parts for image classification. In *ICLR*, 2015. 1, 2, 4, 6

[39] F. Perronnin and C. R. Dance. Fisher kernels on visual vocabularies for image categorization. In *CVPR*, 2007. 2

[40] A. Quattoni and A. Torralba. Recognizing indoor scenes. In *CVPR*, 2009. 5

[41] O. Russakovsky, Y. Lin, K. Yu, and L. Fei-Fei. Object-centric spatial pooling for image classification. In *ECCV*, 2012. 1, 2

[42] F. Sadeghi and M. F. Tappen. Latent pyramidal regions for recognizing scenes. In *ECCV*, 2012. 2

[43] T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, and T. Poggio. Robust object recognition with cortex-like mechanisms. *PAMI*, 2007. 2

[44] G. Sharma, F. Jurie, and C. Schmid. Discriminative spatial saliency for image classification. In *CVPR*, 2012. 1

[45] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 5, 6

[46] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *ICCV*, 2003. 2

[47] J. Sun and J. Ponce. Learning discriminative part detectors for image classification and cosegmentation. In *ICCV*, 2013. 1, 2

[48] C. Thériault, N. Thome, and M. Cord. Dynamic scene classification: Learning motion descriptors with slow features analysis. In *CVPR*, 2013. 2

[49] C. Thériault, N. Thome, and M. Cord. Extended Coding and Pooling in the HMAX Model. *IEEE Transactions on Image Processing (TIP)*, 2013. 2

[50] C.-N. Yu and T. Joachims. Learning structural svms with latent variables. In *ICML*, 2009. 2

[51] F. X. Yu, D. Liu, S. Kumar, T. Jebara, and S.-F. Chang. $\propto$svm for learning with label proportions. In *ICML*, 2013. 2

[52] Y. Yue, T. Finley, F. Radlinski, and T. Joachims. A support vector method for optimizing average precision. In *SIGIR*, 2007. 4, 5

[53] N. Zhang, M. Paluri, M. Ranzato, T. Darrell, and L. Bourdev. PANDA: Pose Aligned Networks for Deep Attribute Modeling. In *ECCV*, 2014. 2

[54] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning Deep Features for Scene Recognition using Places Database. *NIPS*, 2014. 2, 5, 6