Deep Learning on Small Datasets using Online Image Search

Martin Kolář, Michal Hradiš, Pavel Zemčík
{kolarmartin,ihradis,zemcik}@fit.vutbr.cz
Brno University of Technology

Abstract

This paper tackles the important unsolved problem of training deep models with small amounts of annotated data. We propose a semi-supervised self-training bootstrap to deep learning which retrieves and utilizes additional images from internet image search.

We adapt the pseudolabel method proposed by Dong-Hyun Lee in 2013, previously used on the elementary MNIST handwritten digit classification task. We show that by suitable modifications to its example weighting and selection mechanisms it can be adapted to general image classification tasks supported by online image search.

The proposed approach does not require any human supervision, it is practical and efficient, and it actively avoids overtraining. The usefulness of the proposed method is demonstrated on the SUN 397 dataset with only 50 training images per category. When exploiting results of Google's Image Search, we achieve a significant improvement, with a classification accuracy of 51%, as opposed to 39% without our method.

Keywords: convolutional neural network, deep learning, image classification, reinforcement learning

1. Introduction

Image classification is an important and challenging problem of Computer Vision. Traditionally, visual categories could be learned by Support Vector Machines on histograms of local features [35]. Current approaches have shifted towards Convolutional Neural Networks [14, 28, 5], which require vast amounts of data and computational power to learn millions of parameters. Such approaches have achieved near-human performance in face recognition [33], in image segmentation [13], and have beaten previous approaches in classification of both very broad and very specific categories [25]. The motivation for our approach is to make it possible to use datasets with few examples, but it may also potentially be used to fine-tune Convolutional Neural Networks (CNNs) which already achieve high accuracy.

Deep Learning relies on large labelled datasets, with several hundred images for each category, but the creation of such datasets is demanding. Imperfect datasets can be created cheaply in an automated fashion, but near-perfect labelling, required by current approaches, relies on manual selection.

Creating datasets autonomously from the web has been demonstrated to work well when the requested data is in the form of text labels [26, 27]. In this paper, we focus on the problem where a few images are already known, and a label can also be retrieved, so that we can also learn classifiers when the label is ambiguous (such as "crane") or machine-generated.

One way to do Deep Learning on small datasets is to initialize network parameters from an existing network. Networks have been show to produce excellent embeddings, which generalize well to new categories [23, 5]. However, this approach is limited, and a larger dataset will further improve results.

The contribution of this paper, shown in Figure 1, has the

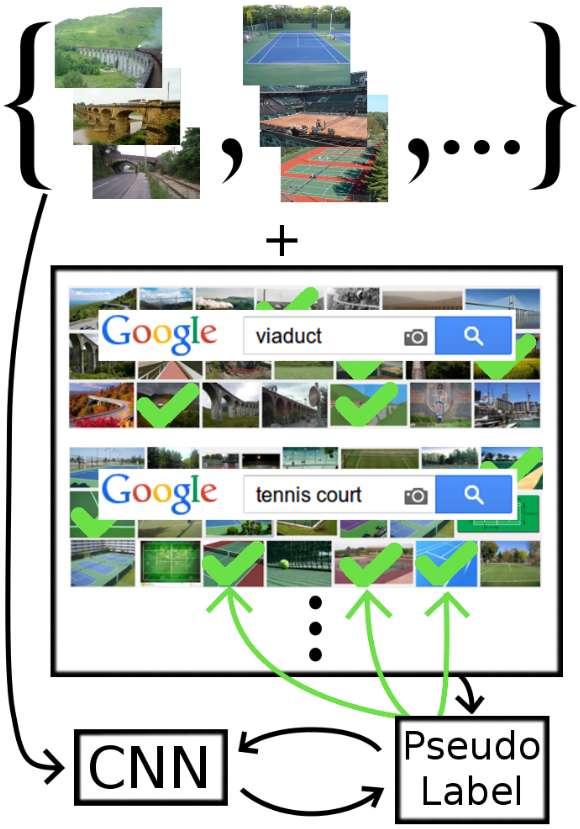


Figure 1: Pseudolabel selects useful additional images from an unreliable source, to help train a Deep Learning classifier

ability to learn visual categories from fewer images than previous approaches. We do this by modifying the pseudolabel [17] method which augments labelled training images with unlabelled images, to create a method capable of handling labelled training images as well as queried images, which are likely to belong to the desired class. This is achieved by modifying the weighting and selection processes.

The presented method adapts the pseudolabel approach to allow the use of web-scale datasets of millions of images. The results are demonstrated on a toy problem devised from the SUN 397 dataset, and on the full SUN 397 dataset expanded with images gathered from Google's image search without human intervention. The toy problem allows us to analyse the properties of the data selection progress during training. A significant improvement is achieved with our approach on the full dataset, from 39% without pseudolabels to 51% with them with the same CNN.

2. Previous Work

As discussed in Section 2.1, Convolutional Neural Networks produce state-of-the-art results, but train poorly on small datasets. Class complexity and variability are decisive for defining sufficient dataset size, so we consider any dataset with an insufficient number of examples to be "small". The pseudolabel method in Section 2.2 uses an unlabelled dataset to mitigate this. Any such approach needs to fully consider Dataset Bias and Limitations, Section 2.3. Semi-Supervised Learning offers a structured approach to utilize labelled data in conjunction with a dataset which is unlabelled, or labelled with known uncertainty, and this work is discussed in Section 2.4.

2.1. Convolutional Neural Networks

Convolutional Neural Networks [15] are the state-of-the-art approach for image classification, achieving the best accuracy for classification and detection [25]. These methods require large datasets [34], and this is usually handled by dataset augmentation with rotation, distortion, and other changes to the used images [14].

While much excellent work has been done to enhance the abilities of CNNs on large datasets [39, 32, 28, 29, 31, 22], it has generally been accepted that small datasets cannot be directly trained upon with random weight initialization. In this work, we focus on using the CNN structure to improve accuracy, rather than explicitly attempting to improve features, because features can be transferred from classifiers trained on other datasets [43].

Other approaches to train on small datasets without Neural Networks have been published, with limited success, such as a generative models [8] and a V1-like model [4].

2.2. Pseudolabel

Pseudolabel [17] introduced Semi-Supervised Learning to Convolutional Neural Networks. As shown in Algorithm 1, the CNN is trained in the usual way, but training images are supplemented by an unlabelled dataset. Low-density separation between classes justifies the use of entropy regularization on additional data.

```
Data: labelled images, unlabelled images
Result: trained classifier
initial training of CNN with labelled images only;
while CNN not converged do
    for each unlabelled image I do
        pick the class with max predicted probability
    end
    train CNN with labelled and weighted
    pseudolabelled images
end
```

Algorithm 1: Original pseudolabel algorithm [17]

In each iteration, the unlabelled set is classified with the current network, and these predictions are used as labels for the next iteration. Random selection from the unlabelled set, together with increasing weights for the selected subset, are meant to help convergence to a classifier principally influenced by the training set.

This approach is justified by the cluster assumption, which states that the decision boundary should lie in low-density regions to improve generalization performance [1]. Rather than explicitly searching for low-density regions, the pseudolabel approach implicitly finds these, because changes in classification are more likely to occur in regions where the consensus among examples can be perturbed by few label changes. These properties are desirable when using additional data, and the method proposed here maintains these advantages, but allows to use more informative and readily-available data. The pseudolabel approach helps with the MNIST dataset, divided artificially into a training set and a mixed set for which labels are unknown. An accuracy comparable to the one achieved by using the entire set was reached. However, this dataset is long considered solved [37], and similar results have not been demonstrated on a challenging problem.

2.3. Dataset Bias and Limitations

Datasets can have a variety of biases, which affect the trained classifier [34]. Since object classification should perform well across a broad spectrum of variances, such as lighting or deformation, datasets should exhibit these as well. Most datasets [25, 20, 7] are created semi-automatically: images are queried from a trusted automated source, and manually sifted through. Depending on the source, this leads to different forms of bias: ImageNet is known to contain mainly centered images, and SUN 397 is mostly composed of canonical ('archetypal') scenes.

By augmenting a biased dataset with additional data, the bias can be reduced and the resulting classifier may demonstrate less unwanted specificity, and thus a better generalization. This can be accomplished by extending the datasets manually, and image classifiers have greatly benefited from new, larger datasets (see Table 1). Similarly, human level performance on the Labelled Faces in the Wild dataset[1] [12] was achieved by

[1] 13 323 web photos of 5 749 celebrities

Dataset	# categories	# images containing instance	Top published classification accuracy
MNIST [16]	10	5 421 - 6 745 (mean 6 000)	99.79 [37]
PASCAL VOC 2012 [6]	20	303 - 4 087 (mean 834)	90.3 mAP [38]
MS COCO [20]	91	~300 - ~600 000 (mean 7 849)	59.0 mAP [11]
Caltech 101 [8]	101	31 - 800 (mean 90)	93.42 ± 0.5 [10]
Caltech 256 [9]	256	80 - 827 (mean 119)	82.2 [42]
SUN 397 [41]	397	100 - 2 361 (mean 274)	54.32 ± 0.14 [44]
ImageNet [25]	1 000	732 - 1 300 (mean 1 281)	68.4 (top-1), 92.3 (top-5) [40]

Table 1: Comparison of image classification datasets. Note that the top-1 metric is inherently inappropriate for ImageNet

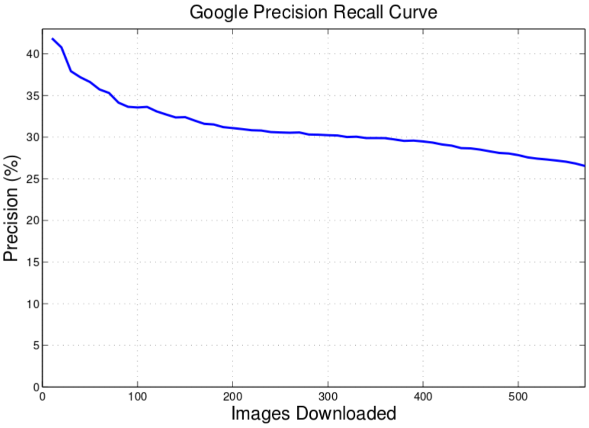


Figure 2: The portion of images returned by Google in 2007 rated *good* while constructing the Caltech 256 dataset [9].

pretraining on a private dataset of 800 to 1 200 faces for 4 030 people [33].

Table 1 lists the most popular image classification datasets. While a larger number of categories makes classification increasingly difficult, the top published classification accuracy is more correlated with the number of example images per category.

A specific type of database bias can even be seen in raw images from Google's image search: low accuracy, constructive error, and canonicity. For instance, a search for the SUN 397 category "marsh" will yield many images of people with the surname "Marsh", and a search for "mountain" will yield a disproportional number of visually pleasing photographs, such as images of the Matterhorn. Google's image search accuracy decreases as further images are queried; see Figure 2. However, the incentive behind using such data is that the sheer number of images guarantees that there will nevertheless be many representative ones.

2.4. Semi-Supervised Learning

Weakly Supervised Multiple Instance Learning (WSMIL) is a subproblem of Semi-Supervised Learning. By making the assumption that at least one of the queried images for each class is correctly labelled, training with online image search data becomes WSMIL [36]. This approach has been coupled with the traditional image classification approach of a dividing hyperplane in a feature histogram hyperspace [19, 18].

In contrast to approaches which have additional labelled data with a probability of being correct (such as in our method), correctly labelled data can be supplemented by unlabelled data belonging to the same classes (as is done in the original pseudolabel paper [17]).

CNNs have also been coupled with WSMIL [22, 3], but in the setting of searching through an image for the object instance, rather than searching through weakly labelled images. Semi-supervised learning approaches have also been proposed to mitigate CNN sensitivity to noisy labels [24, 30].

3. Method

The proposed algorithm (Algorithm 2) is composed of an initial pre-training step, a selection process, and a repeated weighted training step.

```
Data: labelled images, queried images for each class
Result: trained classifier
initial training of CNN with labelled images only;
while CNN not converged do
    for each queried image I do
        select whether to use I for training
    end
    train CNN with labelled and selected images
end
```

Algorithm 2: Proposed pseudolabel algorithm

In the original paper [17], pseudolabels are labels assigned during each epoch to any unlabelled images based on classifier responses. In our setting, pseudolabels are weighted labels of the class used to query each image in online image search.

The following conventions are adopted: $\mathbf{X}$ is a set of images $\{X_1, X_2, X_3, \dots\}$, $\mathbf{y}$ is a set of labels $\{y_1, y_2, y_3, \dots\}$ where $y_n \in \{1, 2, \dots C\}$. C denotes the number of categories. Images and their labels form fixed pairs, but may be denoted separately. Training examples have the form $(\mathbf{X}, \mathbf{y})$. Every i model update iterations are constitute one epoch, and a set of images and labels during the duration of epoch e is denoted $(\mathbf{X}_e, \mathbf{y}_e)$.

Labelled images are divided into a training set and testing set: $(\mathbf{X}^{\text{train}}, \mathbf{y}^{\text{train}})$, $(\mathbf{X}^{\text{test}}, \mathbf{y}^{\text{test}})$.

In addition to the training and testing sets, query images are retrieved from an online image search engine separately for each category. The queried images are denoted $(\mathbf{X}^{\text{query}}, \mathbf{y}^{\text{query}})$.

3.1. Training Convolutional Neural Networks

CNNs are trained by Stochastic Gradient Descent, where training images are propagated forward through the network in batches to produce outputs, for which error gradients are calculated. To complete an iteration, these are backpropagated to calculate loss gradients, which are used to update network weights. This process is repeated until convergence.

3.2. Pseudolabels with Query Images

The method published here relies on a different pseudolabel selection mechanism and a different pseudolabel weighting to the original approach [17]. When training with pseudolabel data, the CNN is trained as described in Section 3.1. However, $\mathbf{X}^{\text{query}}$ images are evaluated with the current network at each epoch, and some are selected with pseudolabels $\mathbf{X}^{\text{pl}}$, for training.

At the beginning of training, $\mathbf{X}_0^{\text{pl}}$ is empty.

$$\mathbf{X}_0^{\text{pl}} = \emptyset \tag{1}$$

For the first i iterations (during epoch 0), the CNN is trained only with $(\mathbf{X}^{\text{train}}, \mathbf{y}^{\text{train}})$. Then, $\mathbf{X}_0^{\text{query}}$ is propagated forward through the CNN, to produce a set of vectors of beliefs for all labels $\mathbf{b}_0$ for every query image. These beliefs correspond to the normalized outputs of the last fully connected layer, before applying the last softmax layer.

Then, a randomized selection process chooses which predicted labels $\mathbf{y}^{\text{query}}$ will be trusted. Pseudolabel examples $\mathbf{X}_e^{\text{pl}}$ from the previous epoch are excluded.

$$(\mathbf{X}_{e+1}^{\text{pl}}, \mathbf{y}_{e+1}^{\text{pl}}) = \mathit{selected}(\mathbf{X}^{\text{query}} \setminus \mathbf{X}_e^{\text{pl}}, \mathbf{y}^{\text{query}}, \mathbf{b}_e) \tag{2}$$

The selection method proposed in this paper is explained in Section 3.3. The rest of $\mathbf{X}^{\text{query}} \setminus \mathbf{X}_e^{\text{pl}}$ is unused in this epoch.

In each following epoch e, the CNN is trained with

$$\{(\mathbf{X}_e^{\text{pl}}, \mathbf{y}_e^{\text{pl}}), (\mathbf{X}^{\text{train}}, \mathbf{y}^{\text{train}})\} \tag{3}$$

Section 3.4 discusses how $\mathbf{y}_e^{\text{pl}}$ can be weighted against $\mathbf{y}^{\text{train}}$ for better convergence stability.

3.3. Pseudolabel Selection

Each example image is chosen with probability:

$$\frac{(1 - \lambda_c) * b_e}{2} \tag{4}$$

Where the accuracy λ_c for each class c on unlabelled data is the ratio of images classified as class c to the number of queried images in class c. By making the weak assumption that queried class accuracies across queried data are similar, class accuracies λ_c for the classifier are an indicator of training data and class complexity for each category.

The classifier belief b_e is the activation of the image for the queried class, as predicted by the network. By using the normalized belief in the $\mathbf{y}^{\text{query}}$ class, the selection favours images the classifier is more confident about, thus removing incorrect query images. This belief is normalized across network responses.

Classes with higher accuracy on the query dataset are given lower pseudolabel priority. This is accomplished with the $(1 - \lambda_c)$ factor.

A number of factors affect the quantitative benefit of using pseudolabelled images: dataset belief, accuracy of the selection method, difference between datasets, selection variability over epochs, and randomization. Our selection method balances these by selecting images in a randomized order, which depends on class accuracies and classifier belief for the correct class.

The last step is randomization. A portion of query images is randomly removed during selection. In our experiments, we chose to remove 50%, and found this beneficial. This is justified by a need to regularize across data when the CNN is trained.

3.4. Pseudolabel Weighting

Pseudolabels are likely to affect the classifier adversely when it hasn't yet reached a sufficient accuracy, just as the classifier would fail to train on raw query data. Self-training is prone to quickly converge to suboptimal solutions, because the classifier assigns high confidence to wrong examples. How this is mitigated in our approach is explained below.

In the original pseudolabel paper [17], images from the training set have constant weights and the pseudolabel losses are weighted by α, where α increases with time according to two hyperparameters.

Our experiments showed that this method is not more effective than setting $\alpha = 0$ until the network converges on training examples, and then setting $\alpha = 1$. This method crucially relies on the network's ability to create a weak classifier from the training data alone, and we found that this is the case with the previously published α tuning method as well. All shown results are achieved with this step function, thus demonstrating its usefulness.

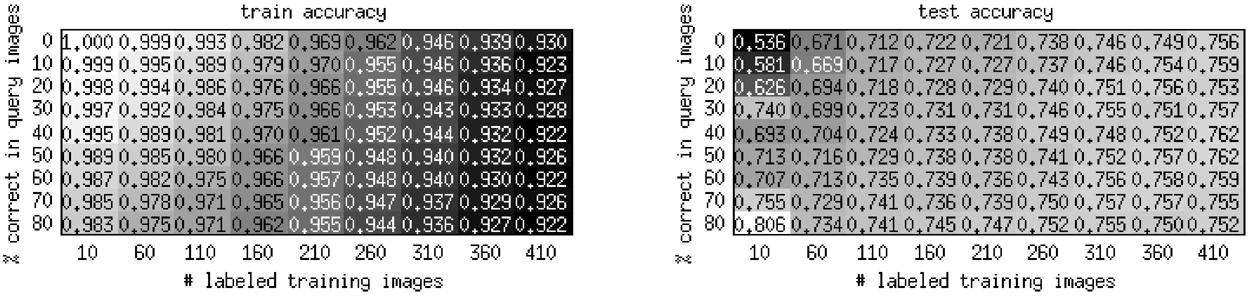
This weighting method, albeit crude, simplifies hyperparameter tuning, and at the cost of a few epochs, achieves the same accuracy.

3.5. Dataset Belief

For an automatically retrieved set of images, a crucial information for deciding whether to train using pseudolabels is the accuracy of the queried data. The unknown proportion of images which belong to the queried category is B, or query accuracy.

Query images can be wrong, misleading, and/or contain correctly and incorrectly labelled images from the training dataset, see Figure 3.

The proposed approach assures that an imperfect selection varies over epochs, in order to mitigate convergence to a non-median representation of the class.



train accuracy

% correct in query images	10	60	110	160	210	260	310	360	410
0	1.000	0.999	0.993	0.982	0.969	0.962	0.946	0.939	0.930
10	0.999	0.995	0.989	0.979	0.970	0.955	0.946	0.936	0.923
20	0.998	0.994	0.986	0.976	0.966	0.955	0.946	0.934	0.927
30	0.997	0.992	0.984	0.975	0.966	0.953	0.943	0.933	0.928
40	0.995	0.989	0.981	0.970	0.961	0.952	0.944	0.932	0.922
50	0.989	0.985	0.980	0.966	0.959	0.948	0.940	0.932	0.926
60	0.987	0.982	0.975	0.966	0.957	0.948	0.940	0.930	0.922
70	0.985	0.978	0.971	0.965	0.956	0.947	0.937	0.929	0.926
80	0.983	0.975	0.971	0.962	0.955	0.944	0.936	0.927	0.922

test accuracy

% correct in query images	10	60	110	160	210	260	310	360	410
0	0.536	0.671	0.712	0.722	0.721	0.738	0.746	0.749	0.756
10	0.581	0.669	0.717	0.727	0.727	0.737	0.746	0.754	0.759
20	0.626	0.694	0.718	0.728	0.729	0.740	0.751	0.756	0.753
30	0.740	0.699	0.723	0.731	0.731	0.746	0.755	0.751	0.757
40	0.693	0.704	0.724	0.733	0.738	0.749	0.748	0.752	0.762
50	0.713	0.716	0.729	0.738	0.738	0.741	0.752	0.757	0.762
60	0.707	0.713	0.735	0.739	0.736	0.743	0.756	0.758	0.759
70	0.755	0.729	0.741	0.736	0.739	0.750	0.757	0.757	0.755
80	0.806	0.734	0.741	0.745	0.747	0.752	0.755	0.750	0.752

Figure 4: Train and test accuracies with varying correct query images, and varying train set sizes for each class

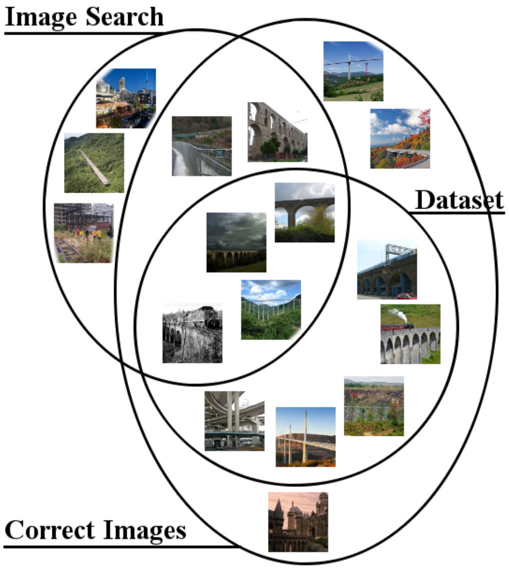


Figure 3: Example images of the viaduct class

3.6. *Difference Between Datasets*

If the training dataset and the images queried from online image search are the same, the method will not be of benefit. It is important that they are complementary, albeit with an overlap, and that they disagree to a degree. The disagreement creates jitter in the hyperspace between images where the classifier should not be divisive, and it supports convergence to a decision boundary elsewhere.

We found that selecting (X^{query}, y^{query}) which fully agrees with the current classifier does not boost classifier accuracy over not using pseudolabels at all. This is because despite bringing new information, the data doesn't create disagreement, and therefore no novelty. In our experiments, we found that a certain degree of wrong and randomly labelled images helped the classifier to converge to higher accuracy over the test set. Adding this form of noise achieves regularisation.

3.7. *Implementation*

All images $\mathbf{X}^{\text{train}}$, $\mathbf{X}^{\text{test}}$, $\mathbf{X}^{\text{query}}$ were resized so that the smaller dimension is 227 pixels, and a central crop of 227×227 pixels is extracted. This has been shown to work better than other cropping methods [2], and the value 227 was chosen because this is the input size of the AlexNet network [14]. Preprocessing details are discussed and evaluated in [2].

The AlexNet [14] architecture was used, and initialized with weights trained on the ImageNet dataset. The network was retrained by keeping all but the last fully connected layer locked, and by updating weights on the last layer.

The network was trained over 100 epochs of 500 iterations each with each combination of parameters. In our GPU-accelerated environment, such a network on the full SUN dataset with all query images converged in 2 to 5 hours.

The ratio of testing data accuracy to queried data accuracy is an indicator of queried dataset accuracy or similarity to the testing dataset. Assuming no constructive errors, such as those CNNs have been demonstrated to fall to when synthesizing examples [21], the number of correctly classified images is a lower bound on how many really belong into the category. A large difference between this number and the actual number (B), directly indicates how much further benefit the new data can have for training.

4. Results

We performed experiments in two setups: the 6 most numerous SUN 397 classes, artificially divided into "labelled" and "query" subsets, and the full SUN 397 dataset with images queried from Google's Image Search[2]. For each set of train, test, and pseudolabel accuracies in figures 4 and 5, the network was trained independently.

In order to compare our approach to the original pseudolabel method, all query labels would have to be ignored. Therefore, tests comparing the two approaches on any dataset would be biased, and aren't included here.

4.1. *Artificial Dataset*

By varying the percentage of correct images in the "query" subset, it was possible to analyze the tolerances of the algorithm. The 6 classes with most images in the SUN 397 dataset contain between 1126 and 2439 images, and these were divided

[2] this data cannot be made public due to licensing issues

into training, testing, and query subsets. The query subset was then diluted with images from all other SUN 397 classes to varying degrees. Experimental results are shown in Figure 4.

Training accuracy, which increases beyond testing accuracy when overtraining, goes down with more training images, as well as with a higher proportion of correct query images. This demonstrates that by applying our method, overtraining is being mitigated. Test accuracy benefits most from pseudolabels with 60 to 160 training images per class, and only when there are at least 20% correct images in the query dataset.

Interestingly, with only 10 training images per class and highly accurate query data, classifier accuracy fluctuates, and sometimes reaches better results than by using the same amount of correct images by training without pseudolabels. This may be because the classifier is able to ignore outliers among training images, which correspond to unhelpful examples.

4.2. Full SUN 397 dataset

The SUN 397 dataset is randomly divided into a train set and test set, by using n images for training, and the rest for testing. We performed experiments with $n = [5, 20, 50]$

The query set was retrieved from Google's image search separately for each category, by searching the full name of the SUN 397 category (ex.: "swimming pool indoor"), and retrieving all full scale original images. Only images which produced an erroneous http query were ignored, and the number of images found was between 230 and 1 359, with mean 796 for a total of 316 024 images. See Figure 6 for the distribution of counts. An automated image similarity search was applied to remove duplicate images, in order to avoid overcounting problems. Other online image search tools could have been used, but Google was selected because it returns hundreds of images, and its accuracy has been analyzed in previous work [9].

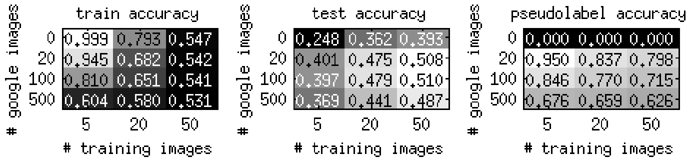


Figure 5: train, test, and pseudolabel accuracy sets with SUN 397 supplemented by online query images, for various numbers of training images and top google image search queries. Top rows are without pseudolabels.

Figure 5 shows the accuracy distribution across classes with and without pseudolabels. Pseudolabel accuracy is the ratio of images whose query label agrees with the predicted label. Note that the quality of queried images decreases with additional images, offsetting the benefit from pseudolabels on larger queries. We can see that the pseudolabel approach reaches higher accuracy than classifiers trained without it, but that too many additional Google images are detrimental. This may be because they vastly outnumber training images, and training labels are drowned out by the noise.

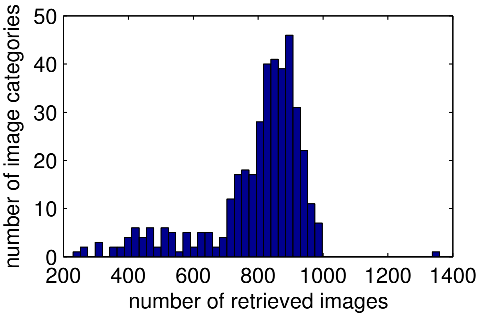


Figure 6: Image counts for categories queried through Google

5. Conclusion and Future Work

The goal of this paper was to demonstrate that CNN training in the semi-supervised setting can be beneficial with small datasets supplemented by images queried from online image search. Experimental results demonstrate that this method is of significant benefit especially if the number of training samples is small (60 - 160), or the images in the training sample are not as representative as the query data.

By adapting pseudolabels to real-world datasets, novel results have been accomplished, facilitating progress in deep learning where image data is sparse. The method was justified, experimentally analysed, and validated. Finally, it was shown that by using our pseudolabel approach, accuracy on the SUN 397 dataset was substantially improved from 39% to 51% with 50 training images in each category.

Future work includes searching for a way to work with online image search data only, without the need for a labelled dataset, or with a labelled dataset of a few images only. The existing method does not work with useful data from other categories, and may benefit from a model of interclass relationships. In addition, querying images from a given category is a non-trivial problem, and the method may be expanded to handle further information on where query images came from, and map intra-class disparities. It will also be beneficial to experiment on classifiers created with subsets of the ImageNet dataset, so that accuracy can be compared for various class sizes even for hierarchical classes of varying complexity.

Acknowledgements

This research is supported by the ARTEMIS joint undertaking, under grant agreement no. 621439 (ALMARVI) and IT4Innovations centre of excellence, and NPU II grant of MSMT CR no. LQ1602.

6. References

[1] O. Chapelle and A. Zien. Semi-supervised classification by low density separation. *AI STATS 2005*, 2004.

[2] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. *arXiv preprint arXiv:1405.3531*, 2014.

[3] R. G. Cinbis, J. Verbeek, and C. Schmid. Multi-fold mil training for weakly supervised object localization. In *CVPR*, pages 2409–2416. IEEE, 2014.

[4] J. J. DiCarlo, N. Pinto, and D. D. Cox. Why is real-world visual object recognition hard? 2008.

[5] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. *arXiv preprint arXiv:1310.1531*, 2013.

[6] M. Everingham, S. A. Eslami, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 111(1):98–136, 2014.

[7] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

[8] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Computer Vision and Image Understanding*, 106(1):59–70, 2007.

[9] A. P. P. Griffin, G. Holub. The caltech 256. 2007.

[10] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *arXiv preprint arXiv:1406.4729*, 2014.

[11] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.

[12] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical report, Technical Report 07-49, University of Massachusetts, Amherst, 2007.

[13] I. Kokkinos. Surpassing humans in boundary detection using deep learning. *CoRR*, abs/1511.07386, 2015.

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[15] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989.

[16] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[17] D. Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. *Workshop on Challenges in Representation Learning, ...*, 2013.

[18] W. Li, L. Duan, I. W. Tsang, and D. Xu. Batch mode adaptive multiple instance learning for computer vision tasks. In *CVPR*, pages 2368–2375. IEEE, 2012.

[19] W. Li, L. Duan, D. Xu, and I. W. Tsang. Text-based image retrieval using progressive multi-instance learning. In *ICCV*, pages 2049–2055. IEEE, 2011.

[20] 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 *Computer Vision–ECCV 2014*, pages 740–755. Springer, 2014.

[21] A. Nguyen, J. Yosinski, and J. Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. *CVPR*, 2015.

[22] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Weakly supervised object recognition with convolutional neural networks. Technical Report HAL-01015140, INRIA, 2014.

[23] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson. Cnn features off-the-shelf: an astounding baseline for recognition. In *CVPR*, pages 512–519. IEEE, 2014.

[24] S. Reed, H. Lee, D. Anguelov, C. Szegedy, D. Erhan, and A. Rabinovich. Training deep neural networks on noisy labels with bootstrapping. *arXiv preprint arXiv:1412.6596*, 2014.

[25] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 2015.

[26] F. Schroff, A. Criminisi, and A. Zisserman. Harvesting image databases from the web. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(4):754–766, 2011.

[27] F. Schroff, A. Zisserman, and A. Criminisi. *Semantic image segmentation and web-supervised visual learning*. PhD thesis, University of Oxford, 2009.

[28] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks. Dec. 2013.

[29] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[30] S. Sukhbaatar, J. Bruna, M. Paluri, L. Bourdev, and R. Fergus. Training convolutional networks with noisy labels. *arXiv preprint arXiv:1406.2080*, 2014.

[31] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going Deeper with Convolutions. *CoRR*, abs/1409.4, 2014.

[32] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. J. Goodfellow, and R. Fergus. Intriguing properties of neural networks. *CoRR*, abs/1312.6, 2013.

[33] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *CVPR*, pages 1701–1708. IEEE, 2014.

[34] A. Torralba and A. A. Efros. Unbiased look at dataset bias. In *CVPR*, pages 1521–1528. IEEE, 2011.

[35] K. E. Van De Sande, T. Gevers, and C. G. Snoek. Evaluating color descriptors for object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1582–1596, 2010.

[36] S. Vijayanarasimhan and K. Grauman. Keywords to visual categories: Multiple-instance learning forweakly supervised object categorization. In *CVPR*, pages 1–8. IEEE, 2008.

[37] L. Wan, M. Zeiler, S. Zhang, Y. L. Cun, and R. Fergus. Regularization of neural networks using dropconnect. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pages 1058–1066, 2013.

[38] Y. Wei, W. Xia, J. Huang, B. Ni, J. Dong, Y. Zhao, and S. Yan. Cnn: Single-label to multi-label. *arXiv preprint arXiv:1406.5726*, 2014.

[39] R. Wu, S. Yan, Y. Shan, Q. Dang, and G. Sun. Deep Image: Scaling up Image Recognition. *arXiv preprint arXiv:1501.02876*, 2015.

[40] Z. Wu, Y. Zhang, F. Yu, and J. Xiao. A gpu implementation of googlenet.

[41] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *CVPR*, pages 3485–3492. IEEE, 2010.

[42] S. Yan, J. Dong, Q. Chen, Z. Song, Y. Pan, W. Xia, Z. Huang, Y. Hua, and S. Shen. Generalized hierarchical matching for sub-category aware object classification. In *Visual Recognition Challange workshop, ECCV*, volume 5, 2012.

[43] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson. How transferable are features in deep neural networks? In *Advances in Neural Information Processing Systems*, pages 3320–3328, 2014.

[44] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In *Advances in neural information processing systems*, pages 487–495, 2014.