

ViSER: Visual Self-Regularization

Hamid Izadinia*
 University of Washington

Pierre Garrigues*
 Twitter

Abstract

We propose using large set of unlabeled images as a source of regularization data for learning robust representation. Given a visual model trained in a supervised fashion, we augment our training samples by incorporating large number of unlabeled data and train a semi-supervised model. We demonstrate that our proposed learning approach leverages an abundance of unlabeled images and boosts the visual recognition performance which alleviates the need to rely on large labeled datasets for learning robust representation. In our approach, each labeled image propagates its label to its nearest unlabeled image instances. These retrieved unlabeled images serve as local perturbations of each labeled image to perform Visual Self-Regularization (ViSER). Using the labeled instances and our regularizers we show that we significantly improve object categorization and localization on the MS COCO and Visual Genome datasets.

1. Introduction

Image recognition has rapidly progressed and extremely effective performance is observed using the ImageNet dataset [22, 7, 35, 17]. Despite this progress, ImageNet is biased towards single object images which is in contrast with photos taken by people typically capturing a range of objects in various context. Also, object categories in ImageNet is a subset of the lexical database WordNet [29] which makes it biased to certain categories and does not match the scope of more general image recognition tasks such as object detection and localization in context. MS COCO [25] and Visual Genome [21] provide realistic benchmark for image recognition systems. MS COCO contains $\sim 300K$ images and ~ 80 object categories, whereas Visual Genome contains 100K images and thousands of categories. CNNs are also showing the best performance on these datasets [34, 21]. MS COCO and Visual Genome are annotated via crowdsourcing platforms such as Amazon Mechanical Turk and hence is time-consuming and expensive to obtain additional labels. However, we have access to huge quantities of unlabeled or *weakly* labeled images. For example, the Yahoo Flickr Creative Commons 100M dataset (YFCC) [40] is comprised of a hundred million Flickr photos with user-provided annotations such as photo tags, titles, or descriptions.

*Work was done at Flickr, Yahoo Research.



Figure 1. The t-SNE [27] map of the whole set of images (including MS COCO and YFCC images) labeled as ‘Bus’ category after applying our proposed ViSER approach. Can you guess whether green or blue background correspond to the human annotated images of MS COCO dataset?

Answer key: blue: MS COCO, green: YFCC

We present a simple yet effective semi-supervised learning algorithm that is able to leverage labeled and unlabeled data to improve classification accuracy on the MS COCO and Visual Genome datasets. We first train a fully convolutional network using the multi-labeled data (e.g. MS COCO or Visual Genome). Then, for each training sample we retrieve the nearest samples in YFCC using the cosine similarity in a semantic space of the penultimate layer in the trained fully convolutional network. We call these *Regularizer* samples which can be considered as real perturbed samples compared to the Gaussian noise perturbation considered in virtual adversarial training [31]. To be practical at scale, we propose an approximate distributed algorithm to find the images with semantically similar attention activation. Our experimental results show that our method is applicable in object-in-context retrieval as well as image recognition and significantly improves performance over previous methods when trained using only the labeled data.

2. Related work

The recognition and detection of objects that appear “in context” is an active area of research. The most common benchmarks for this task are the PASCAL VOC [10] and MS COCO [25] datasets. It has been shown in [33, 38, 9, 3, 42] that it is possible to classify and localize objects using training data without object bounding box information. We refer to this training data as *weakly-supervised*. The size of labeled

“objects in context” datasets is typically small. However, we have access to large amounts of unlabeled web images. The YFCC 100M dataset has one hundred million images that have user annotations such as tags, titles, and description. There has been recent efforts to leverage user annotation to build object classifiers. For instance, [18] proposes a noise model that is able to better capture the uncertainty in the user annotations and improve the classification performance. Annotations are used in [19] and [12] as target labels to learn image features and neural networks from scratch. However, classifier performance is lower when training on noisy data. Contrary to these approaches, we propose a form of curriculum learning [4] where we first train a model on a small set of clean data, and then augment the training set by mining instances from a large set of *unlabeled* images.

While it is shown that by making small perturbations to the input it is possible to make adversarial examples which can fool machine learning models [39, 23, 24], adversarial examples can be used as a means for data augmentation to improve the regularization capability of the deep models. Our method is related to adversarial training techniques [13, 31, 30] in the sense that additional training instances with small perturbations are created and added to the training data. In contrast, we retrieve *real* adversarial examples from a large set of unlabeled images. Such instances usually correspond to large perturbations in the input space but follow the natural distribution of the data which is analogous to the adversarial perturbations. We call our retrieved image instances as *Regularizer* and use them to re-train the model and further improve performance.

Using semi-supervised learning, classifiers can be trained via labeled and unlabeled data such as in Naive Bayes, EM algorithm [32], ensemble methods [5] and propagating labels based on similarity as in [43]. In our case, the size of the unlabeled set is three orders of magnitude larger than the labeled set. The size of unlabeled set is critical in order for the label propagation to work effectively, and we propose approximations using MapReduce to make the search practical [6]. Large-scale nearest neighbor search is used for a variety of tasks such as scene completion [16], image editing with the PatchMatch algorithm [2], image annotation with the TagProp algorithm [15], and image captioning [8]. Similarly labels can be propagated using semantic segmentation [14]. This method is applied on ImageNet which has a bias towards a single object appearing in the center of the image. We focus on images where objects appear in context.

3. Proposed Method

Multiple instance learning for multilabel classification:

We adopt a fully convolutional neural network architecture [26] which produces the output of a $H \times W \times N$ tensor for N object classes. For each object class the corresponding heatmap provides information about the object location as illustrated in Figure 2. We use VGG16 [36] as base architecture and implement our algorithm via TensorFlow [1].

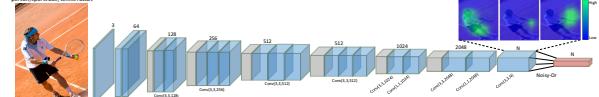


Figure 2. Our Fully Convolutional Network for object localization.

We are given a set of annotated images $\mathcal{A} = \{(x_i, y_i)\}_{i=1\dots n}$, where x_i is an image and $y_i = (y_i^1, \dots, y_i^N) \in \{0, 1\}^N$ is a binary vector determining which object category labels are present in x_i . Let f^l be the object heatmap for the l th label in the final layer of the network. The probability at location j is given by applying a sigmoid unit to the logits f^l , e.g. $p_j^l = \sigma(f_j^l)$. We have a weakly labeled setting and do not have access to object locations. We incorporate a multiple instance learning approach with Noisy-OR operation [28, 41, 11]. The probability for label l is given by Equation 1. For learning the parameters of the FCN, we use stochastic gradient descent to minimize the cross-entropy loss \mathcal{L} formalized in Equation 2.

$$p^l = 1 - \prod_j (1 - p_j^l). \quad (1)$$

$$\mathcal{L} = \sum_{l=1}^N -y^l \log p^l - (1 - y^l) \log (1 - p^l) \quad (2)$$

Visual Self-Regularization: Deep neural networks are vulnerable to adversarial examples [39]. Let x be an image and η a small perturbation such that $\|\eta\|_\infty \leq \epsilon$. If the perturbation is aligned with the gradient of the loss function $\eta = \epsilon \text{sign}(\nabla_x \mathcal{L})$ which is the most discriminative direction in the image space, then the output of the network may change dramatically, even though the perturbed image $\tilde{x} = x + \eta$ is virtually indistinguishable from the original. [13] suggests that this is due to the linear nature of deep neural networks and show that augmenting the training set with adversarial examples results in regularization similar to dropout. In Virtual Adversarial Training [31] the perturbation is produced by maximizing the smoothness of the local model distribution around each data point. The virtual adversarial example is the point in an ϵ ball around the datapoint that maximally perturbs the label distribution around that point as measured by the Kullback-Leibler divergence:

$$\eta = \arg \min_{r: \|r\|_2 \leq \epsilon} KL[p(y|x, \theta) || p(y|x + r, \theta)]. \quad (3)$$

We propose to draw perturbations from a large set of *unlabeled* images \mathcal{U} whose cardinality is much higher than \mathcal{A} . For each example x , we use the example \tilde{x} that is nearby in the space defined by the penultimate layer in our fully convolutional network. This layer contains spatial and semantic information about the objects present in the image, and therefore x and \tilde{x} have similar semantics and composition while they may be far away in pixel space. We consider the cosine similarity metric to find samples which are close to each other in the feature space and for efficiency we compute the dot product of the L2 normalized feature vectors. Let θ denote the optimal parameters found after minimizing the cross-entropy loss using the training data in \mathcal{A} , and $f_\theta(x)$ be the L2 normalized feature vector obtained from

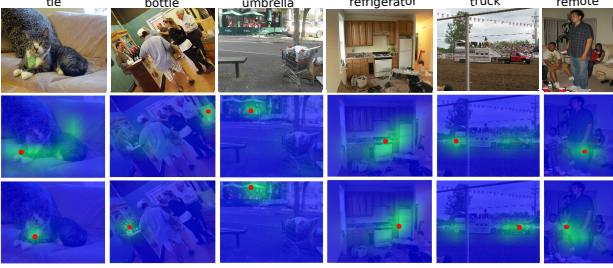


Figure 3. Object localization comparison between “FCN,N-OR”(mid row) and “FCN,N-OR,VISER”(last row).

the penultimate layer of our network($\text{Conv}(1,1,2048)$). The similarity between two images x and x' is computed by their dot product $\mathcal{S}(x, x') = f_\theta(x)^T f_\theta(x')$. For each training sample (x_i, y_i) in \mathcal{A} , we find the most similar item in \mathcal{U}

$$\tilde{x}_i = \arg \max_{x \in \mathcal{U}} \mathcal{S}(x_i, x), \quad (4)$$

and transfer the labels from x_i , to generate a new, *Real Adversarial (Regularizer)*, training sample (\tilde{x}_i, y_i) . Similar to adversarial and virtual adversarial training, our method improves the classification performance. We interpret our sample perturbation as a form of adversarial training where additional examples are sampled from a similar semantic distribution as opposed to noise. We also used the ϵ perturbation of each labeled sample in the gradient direction (similar to adversarial training) to find the nearest neighbor in unlabeled set and observed similar performance.

Large scale approximate regularizer sample search: We use MapReduce [6] to find approximate nearest neighbors in a distributed fashion since we use the YFCC dataset with 100 million images for our unlabeled set. We first pre-compute the feature representations $f_\theta(x_i)$ for $x_i \in \mathcal{A}$. The size of \mathcal{A} for datasets such as MS COCO or Visual Genome is small enough that it is possible for each mapper to load a copy into memory. A mapper then iterates over samples x in \mathcal{U} , computes the feature representation $f_\theta(x)$ and its inner product with the pre-computed features in \mathcal{A} . It emits tuples for the top k_m matches that are keyed by the index in \mathcal{A} , and also contain the index in \mathcal{U} and similarity score. After the shuffling phase, the reducers can select for each sample in \mathcal{A} the k_r closest samples in \mathcal{U} . We use $k_m = 1000$ and $k_r = 10$. We are able to run the search in a few hours, with the majority of the time being in the mapper phase where we compute the image feature representation. Note that our method does not guarantee that we can retrieve the nearest neighbor for each sample in \mathcal{A} . Indeed, if for a sample x_i there exists k_m samples x_j such that $f_\theta(x_j)^T f_\theta(\tilde{x}_i) \geq f_\theta(x_i)^T f_\theta(\tilde{x}_i)$, then the algorithm will output either no nearest neighbor or another sample in \mathcal{U} . However we found our approximate method to work well in practice.

4. Experiments

Semi-Supervised Multilabel Object Categorization and Localization:

We use MS COCO [25] and Visual Genome [21] as our source of clean training data and YFCC [40] for the source of unlabeled images with the



Figure 4. Top regularizer examples from unlabeled YFCC data (row 2-6) retrieved for MS COCO multi-label image query (row 1).

images presented in Visual Genome or MS COCO discarded from YFCC. The data is 14TB and is stored in Hadoop Distributed File System. We conduct distributed nearest-neighbor search via a CPU cluster. For training and evaluation, we use the standard split used in [25] for MS COCO and only use object category annotations in the Visual Genome.

We evaluate via the object classification and point-based object localization task introduced in [33] and use the mean Average Precision (AP) metric. Tables 1 and 2 summarize classification and localization results on the MS COCO and Visual Genome datasets. We compare our performance with [33, 38, 3]. To handle the uncertainty in object localization, [33] considers the last fully connected layers of the network as convolution layers and a max-pooling layer is used to hypothesize the possible location of the object in the images. In contrast, we use Noisy-OR as our pooling layer. In [38], a multi-scale FCN network called ProNet is proposed that aims to zoom into promising object-specific boxes for localization and classification. Our method uses a single fully convolutional network, is simpler and has a lighter architecture as compared to ProNet. In all tables ‘FullyConn’ refers to the standard VGG16 architecture while ‘FullyConv’ refers to the fully convolutional version of our network (see Figure 2). The Noisy-OR loss is abbreviated as ‘N-OR’, and we denote our algorithm with VISER.

As shown in Table 1, our proposed algorithm reaches 50.64% accuracy in the object localization on the MS COCO dataset which is more than a 4% boost over [38] and a 9.5% boost over [33]. Also, without doing any regularization and by only using Noisy OR (N-OR) paired with a fully convolutional network, we obtain higher localization accuracy than Oquab et al. [33] and different variants of ProNet [38]. In the object classification task, our VISER outperforms baselines of [38, 33] by a margin of more than 4.5% and gains an accuracy of 75.48% for the MS COCO dataset. Other variants of [38] are less accurate than our fully convolutional network architecture with Noisy-OR pooling (‘FullyConv, N-OR’). In Table 1 and 2, we compare three forms of regularization: adversarial training (‘AT’) [13], virtual adversarial training

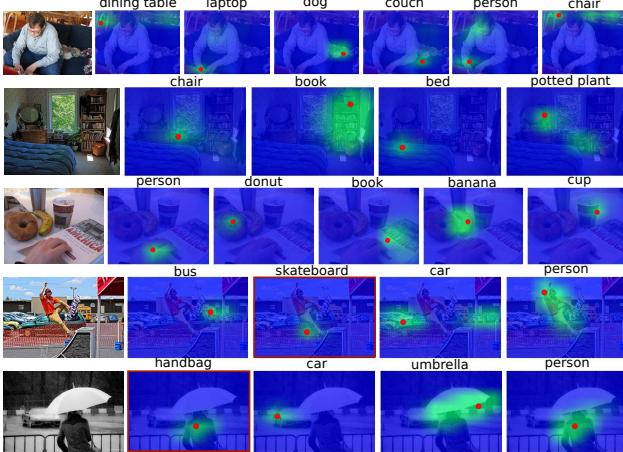


Figure 5. Localization result of ViSER on MS COCO validation.

Table 1. Classification and localization mean AP on MS COCO.

Method	Classification	Localization
Oquab et al. [33]	62.8	41.2
ProNet (proposal) [38]	67.8	43.5
ProNet (chain cascade) [38]	69.2	45.4
ProNet (tree cascade) [38]	70.9	46.4
Bency et al. [3]	54.1	49.2
FullyConn	66.68	–
FullyConv,N-OR	72.52	47.47
FullyConv,N-OR,AT [13]	74.38	49.75
FullyConv,N-OR,VAT [31]	74.30	49.42
FullyConv,N-OR,ViSER	75.48	50.64

(‘VAT’) [31], and our *Visual Self-Regularization* (ViSER) using the YFCC dataset as source of unlabeled images.

Object-in-Context Retrieval: To qualitatively evaluate ViSER, we show several examples of the *Regularizer* retrieved instances in Figure 4. The unlabeled images retrieved by our approach have high similarity with the queried labeled image. Furthermore, most of the objects in the labeled images also appear in the retrieved images which demonstrates the effectiveness of our label propagation approach.

Figure 1 shows the results of our ViSER approach on the ‘Bus’ category. We visualize the t-SNE [27] grid map [20] of the whole set of images labeled as ‘Bus’ which includes instances from both the labeled images in the MS COCO and unlabeled instances from the YFCC dataset. To produce the t-SNE visualization we take the output of the penultimate layer of our network as explained in Section 3. A different background color (blue vs. green) is assigned to images depending on whether they are from the labeled or unlabeled set. Notice that it is challenging to determine the color corresponding to each dataset as photos are from a similar domain. It suggests that there are many images in the large unlabeled web resources that can potentially be used to populate the fully annotated datasets with more examples.

Figure 5 shows qualitative performance of “FCN,N-OR,ViSER” for multi-label localization. We visualize the

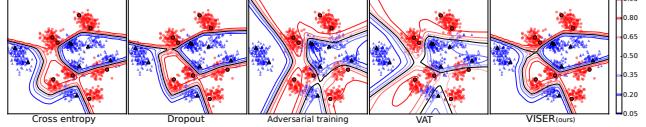


Figure 6. Generalization comparison on a synthetic data. Black bordered samples are training and rest of instances are test set. The contour shows the $p(y = 1|x, \theta)$, from $p = 0$ (blue) to $p = 1$ (red).

Table 2. Classification and localization mean AP on Visual Genome.

Method	Classification	Localization
FullyConn	9.94	–
FullyConv,N-OR	12.35	7.55
FullyConv,N-OR,AT [13]	13.96	9.05
FullyConv,N-OR,VAT [31]	13.95	9.06
FullyConv,N-OR,ViSER	14.82	9.74

Table 3. Classification error on test synthetic dataset (lower is better).

	cross entropy	dropout [37]	AT [13]	VAT [31]	ViSER
Error(%)	9.24±0.65	9.26±0.71	8.96±0.85	8.94±0.39	8.51±0.49

object localization score maps with high probability localization regions shown in green and localized objects displayed by red dots. Several examples of the localization score maps produced by “FCN,N-OR” and “FCN,N-OR,ViSER” are shown in Figure 3. We see that “FCN,N-OR,ViSER” can locate both small and big objects more accurately.

Classification on Synthetic Data: To evaluate the ability of our algorithm to leverage unlabeled data to regularize learning, we generate a synthetic two-class dataset with a multimodal distribution as shown via contour visualization of the estimated model distribution in Figure 6. The dataset contains 16 training instances (each class has 8 modes with random mean and covariance for each mode and 1 random sample per mode is selected), 1000 unlabeled and 1000 test samples. We linearly embed the data in 100 dimensions. We use a neural network with two fully connected layers of size 100, each followed by a ReLU activation and optimized via the cross-entropy loss. We compare the generalization behavior of ViSER with the following regularization methods: dropout [37], adversarial training [13], and virtual adversarial training (VAT) [31]. As Table 3 summarizes misclassification test error over 50 runs, our proposed ViSER learns a better local class distribution as adversarial samples follow the true distribution of the data and are less biased to the training instances, compared to the baselines.

5. Conclusion and Future Work

We presented a simple yet effective method to leverage a large unlabeled set accompanied with a small labeled set to train more generalizable representations. Our ViSER approach is able to find *Regularizer* examples from a large unlabeled set and can achieve significant improvement for visual classification and object localization. In future work, our performance can be further improved by incorporating user provided data such as ‘tags’. Also, our method can be applied for domains beyond visual recognition.

References

- [1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous systems. 2015.
- [2] C. Barnes, E. Shechtman, A. Finkelstein, and D. Goldman. Patchmatch: A randomized correspondence algorithm for structural image editing. *ACM TOG*, 2009.
- [3] A. J. Bency, H. Kwon, H. Lee, S. Karthikeyan, and B. Manjunath. Weakly supervised localization using deep feature maps. In *ECCV*, 2016.
- [4] Y. Bengio, J. Louradour, R. Collobert, and J. Weston. Curriculum learning. In *ICML*, 2009.
- [5] K. P. Bennett, A. Demiriz, and R. Maclin. Exploiting unlabeled data in ensemble methods. In *ACM SIGKDD*, 2002.
- [6] J. Dean and S. Ghemawat. Mapreduce: simplified data processing on large clusters. *Communications of the ACM*, 2008.
- [7] 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.
- [8] J. Devlin, S. Gupta, R. Girshick, M. Mitchell, and C. L. Zitnick. Exploring nearest neighbor approaches for image captioning. *arXiv preprint arXiv:1505.04467*, 2015.
- [9] T. Durand, T. Mordan, N. Thome, and M. Cord. Wildcat: Weakly supervised learning of deep convnets for image classification, pointwise localization and segmentation. In *CVPR*, 2017.
- [10] M. Everingham, A. Zisserman, C. K. Williams, L. Van Gool, M. Allan, C. M. Bishop, O. Chapelle, N. Dalal, T. Deselaers, G. Dorkó, et al. The pascal visual object classes challenge 2007 (voc2007) results. 2007.
- [11] H. Fang, S. Gupta, F. Iandola, R. K. Srivastava, L. Deng, P. Dollár, J. Gao, X. He, M. Mitchell, J. C. Platt, C. L. Zitnick, and G. Zweig. From captions to visual concepts and back. In *CVPR*, 2015.
- [12] P. Garrigues, S. Farfade, H. Izadinia, K. Boakye, and Y. Kalantidis. Tag prediction at flickr: a view from the darkroom. *arXiv preprint arXiv:1612.01922*, 2016.
- [13] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.
- [14] M. Guillaumin, D. Küttel, and V. Ferrari. Imagenet auto-annotation with segmentation propagation. *IJCV*, 2014.
- [15] M. Guillaumin, T. Mensink, J. Verbeek, and C. Schmid. Tagprop: Discriminative metric learning in nearest neighbor models for image auto-annotation. In *ICCV*, 2009.
- [16] J. Hays and A. A. Efros. Scene completion using millions of photographs. In *ACM TOG*, 2007.
- [17] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.
- [18] H. Izadinia, B. C. Russell, A. Farhadi, M. D. Hoffman, and A. Hertzmann. Deep classifiers from image tags in the wild. In *Multimedia COMMONS*, 2015.
- [19] A. Joulin, L. van der Maaten, A. Jabri, and N. Vasilache. Learning visual features from large weakly supervised data. In *ECCV*, 2016.
- [20] A. Karpathy. t-SNE visualization of CNN codes. <http://cs.stanford.edu/people/karpathy/cnnembed/>.
- [21] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, M. Bernstein, and L. Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV*, 2017.
- [22] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.
- [23] A. Kurakin, I. Goodfellow, and S. Bengio. Adversarial examples in the physical world. *ICLR Workshop*, 2017.
- [24] A. Kurakin, I. J. Goodfellow, and S. Bengio. Adversarial machine learning at scale. In *ICLR*, 2017.
- [25] 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*, 2014.
- [26] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015.
- [27] L. v. d. Maaten and G. Hinton. Visualizing data using t-sne. *JMLR*, 2008.
- [28] O. Maron and T. Lozano-Pérez. A framework for multiple-instance learning. *NIPS*, 1998.
- [29] G. A. Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 1995.
- [30] T. Miyato, A. M. Dai, and I. Goodfellow. Adversarial training methods for semi-supervised text classification. In *ICLR*, 2017.
- [31] T. Miyato, S.-i. Maeda, M. Koyama, K. Nakae, and S. Ishii. Distributional smoothing with virtual adversarial training. In *ICLR*, 2016.
- [32] K. Nigam, A. K. McCallum, S. Thrun, and T. Mitchell. Text classification from labeled and unlabeled documents using em. *Machine learning*, 2000.
- [33] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Is object localization for free?–weakly-supervised learning with convolutional neural networks. In *CVPR*, 2015.
- [34] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 2015.
- [35] 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. *IJCV*, 2015.
- [36] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- [37] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *JMLR*, 2014.
- [38] C. Sun, M. Paluri, R. Collobert, R. Nevatia, and L. Bourdev. Pronet: Learning to propose object-specific boxes for cascaded neural networks. In *CVPR*, 2016.
- [39] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.
- [40] B. Thomee, D. A. Shamma, G. Friedland, B. Elizalde, K. Ni, D. Poland, D. Borth, and L.-J. Li. YFCC100M: The new data in multimedia research. *Communications of the ACM*, 2016.
- [41] C. Zhang, J. C. Platt, and P. A. Viola. Multiple instance boosting for object detection. In *NIPS*, 2005.

- [42] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning deep features for discriminative localization. In *CVPR*, 2016.
- [43] X. Zhu and Z. Ghahramani. Learning from labeled and unlabeled data with label propagation. 2002.