DO WE REALLY NEED MORE TRAINING DATA FOR OBJECT LOCALIZATION

Hongyang Li[1]*, Yu Liu*[1]*, Xin Zhang*[2*]*, Zhecheng An*[2]*, Jingjing Wang*[2]*, Yibo Chen*[1] *and Jihong Tong*[3]

[1] The Chinese University of Hong Kong [2] Tsinghua University [3] Eastern Liaoning University

http://www.ee.cuhk.edu.hk/~yangli/project/eic.html

ABSTRACT

The key factor for training a good neural network lies in both model capacity and large-scale training data. As more datasets are available nowadays, one may wonder whether the success of deep learning descends from data augmentation only. In this paper, we propose a new dataset, namely, *Extended ImageNet Classification* (EIC) dataset based on the original ILSVRC CLS 2012 set to investigate if more training data is a crucial step. We address the problem of object localization where given an image, some boxes (also called anchors) are generated to localize multiple instances. Different from previous work to place all anchors at the last layer, we split boxes of different sizes at various resolutions in the network, since small anchors are more prone to be identified at larger spatial location in the shallow layers. Inspired by the hourglass work, we apply a conv-deconv network architecture to generate object proposals. The motivation is to fully leverage high-level summarized semantics and to utilize their up-sampling version to help guide local details in the low-level maps. Experimental results demonstrate the effectiveness of such a design. Based on the newly proposed dataset, we find more data could enhance the average recall, but a more balanced data distribution among categories could obtain better results at the cost of fewer training samples.

Index Terms— Deep learning, computer vision, object localization, image recognition.

1. INTRODUCTION

In recent years, the emergence of deep learning [1] has greatly boosted the performance of many computer vision tasks, for example, object detection [2, 3, 4], classification [5, 6, 7], object tracking [8], etc. The essence behind the success of deep learning resides in the better expressive power of high-dimensional representation in the feature space [9]. To achieve a good neural network, one must resort to datasets of large scale [10, 11]. The recent proposed ImageNet dataset [10] contains 1000 classes of objects in the real-world and is supposed to have large data variance in scale, appearance and context. As deep learning approaches achieve better and better results, one may wonder if the gain descends from augmenting data only, or *do we really need more training data?*

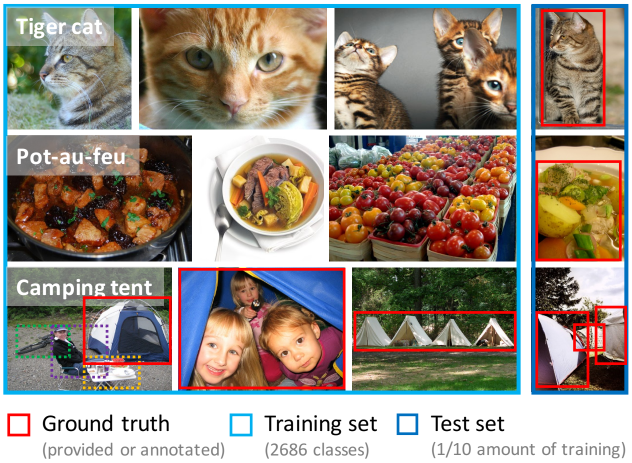

Fig. 1. Overview of the Extended ImageNet Classification dataset. It has branched out sub-classes to a great extent compared with the original ImageNet dataset. The last row and column show ground truth annotations of the training and test set, respectively.

We propose an extended version of the ImageNet classification set [10], namely, Extended ImageNet Classification (EIC) set, where it contains more than 2600 classes and has more 'difficult' images. Figure 1 describes an overview of the proposed EIC dataset. It branches out the original 1000 classes to many detailed sub-classes (*e.g.*, from cat to tiger cat, Persian cat, etc.) and there might be more than one instance in the image. The key problem to address now is that whether more training data is beneficial to obtain better results evaluated on the original smaller scale dataset.

Another concern we focus on is a better way of utilizing the feature maps in the network to obtain better representation (thus better classifier) of data. Previous works [3, 2, 12, 13, 4] resort to the classification or regression using high-level, summarized semantics *only*. While such a design is efficient in practice, it may lose important low-level details due to smaller spatial size after pooling. One may use multiple layers heuristically and concatenate them all to get satisfying results [14], we argue that a proper feature map flow from higher layers

[*] Corresponding author: zhangxin15@mails.tsinghua.edu.cn

978-1-5090-2175-8/17/$31.00 ©2017 IEEE

to lower ones is crucial to help leverage both high-level summarized semantics and low-level local details. Inspired by hourglass structure [15], we place different-sized boxes (also called anchors) at different depth in the network to fully make use of the feature maps (see Figure 2(a)).

In this paper, we address the problem of object localization[1] based on the extended ImageNet classification dataset. The main contributions are: (a) a new dataset is proposed and annotated for object localization, extending the original 1000 classes to 2700 categories; (b) we investigate whether a larger dataset is necessary to train a deep learning model for robust and representative features; (c) for localizing the bounding box, we embed the region proposal network framework in a multi-depth, hourglass style to fully leverage the information of feature maps on different resolutions. The new dataset will be available upon acceptance.

1.1. Related work

Recent years have witnessed a blossom of object proposal methodologies from fully evaluation of proposal protocol [16] to novel structure embedding using CRF [17], for example. DeepBox [18] uses CNN to rerank proposals from a bottom-up method in a data-driven, semantic manner. In [19], they train an ensemble of figure-ground segmentation models for highly accurate bottom-up object segmentation. DeepPropopsal [20] builds on activations of different convolutional layers of a pretrained CNN, combining the localization accuracy in early layers with the high semantics in later ones. Li *et al.* [4] propose a zoom-out-and-in model for identifying object boxes of different sizes from various resolutions in the network and the high-level semantics in deeper layers could better leverage the local details in shallow layers to detect small objects, by way of a decision scheme of map flow.

2. THE EXTENDED IMAGENET DATASET

The Extended ImageNet Classification dataset has 2686 categories with an overlap of 871 classes with the original classification set. The training set includes 2456727 images and the validation set holds around one tenth (273140) the number of images in training. As previously stated in Section 1, some training or test images have more than one instance and thus the actual number of ground truth boxes may exceed the number of images. The extended categories are chosen by the WordNet [21] tree that ImageNet originally descends from. Generally they are subclasses of the preceding (father) node. Most of the source images are directly from the ImageNet-at-large database[2] and the rest descend from Internet in order to make statistics of the proposed EIC dataset balanced. Figure 2(b) illustrates an example of the WordNet hierarchy.

Table 1. Statistics comparison between Extended ImageNet Classification (EIC) dataset and ILSVRC CLS 2012 counterpart. Abbreviations are: im=image, avg=average, cls=class, anno=annotation, obj=object. For the detailed definitions, refer to Section 2.

Dataset split	Extended ImageNet		ILSVRC CLS 2012	
	Train	Val.	Train	Val.
# of images	2,456,727	273,140	1,281,167	50,000
Avg im # per cls	251-1300	34-50	732-1300	50
Avg anno per im	1.53	1.17	1.41	1.02
Avg obj scale	25.37 %	25.50%	25.39%	25.61 %
Small obj %	4.81	4.27	2.35	2.47
Twisted obj %	42.77	44.55	40.72	42.94
Inner-cls distance	0.434	0.396	0.462	0.411
Inter-cls distance	1.12	1.46	1.52	1.55

Table 1 reports some important properties between the proposed dataset and the original counterpart. *Small* objects are the ones whose area is smaller than 32^2; *twisted* objects are the ones whose aspect ratio (w/h) is larger than 4 (wider, `balance beam`, for example) or smaller than 0.25 (taller, `tie`, for instance). The feature distance is defined as $\mathcal{D}(x_1, x_2) = 1 - \cos(x_1, x_2)$, where x_1, x_2 are the features of any two sample images. It holds that $D \in [0, 2]$. The average class distance d_k^1 is the mean of feature distance across all samples within class k; the *inner-class* distance is the mean of average class distance d_k. The representative class feature x_k^2 is the mean feature of all samples within class k; the *inter-class* distance is the mean of the distance $\mathcal{D}(x_i^2, x_j^2)$ across categories i, j in one dataset. We use layer fc6 in the VGG-16 model [22] to represent the feature of an image. From the table we can conclude that EIC contains more images, more harder samples (small or twisted); the inner distance within one category is smaller while the distance across classes are larger due to a more detailed classification among objects.

3. ALGORITHM

3.1. Network architecture

Figure 2(a) describes the network structure we employ for object localization. Such a conv-deconv spirit is inspired by [15]. Higher layers are typically of smaller spatial size due to pooling and contain summarized, high-level semantics while lower layers have more details on the object. These two type of layers can complement each other by upsampling high-level feature maps to match the size of its lower counterpart. Specifically, we adopt the basic building blocks in the Inception-batch-normalization [23] model. An input image is first fed into three convolutional layers, after which the feature maps are downsampled by a rate of 8. There are nine inception modules afterwards, denoted as `ICP3a-3c`, `ICP4a-4e`

[1] The problem of object localization assumes that each image has one class only and at least one instance of the class, whereas in object detection there could be multiple classes and instances in one image.

[2] http://image-net.org/

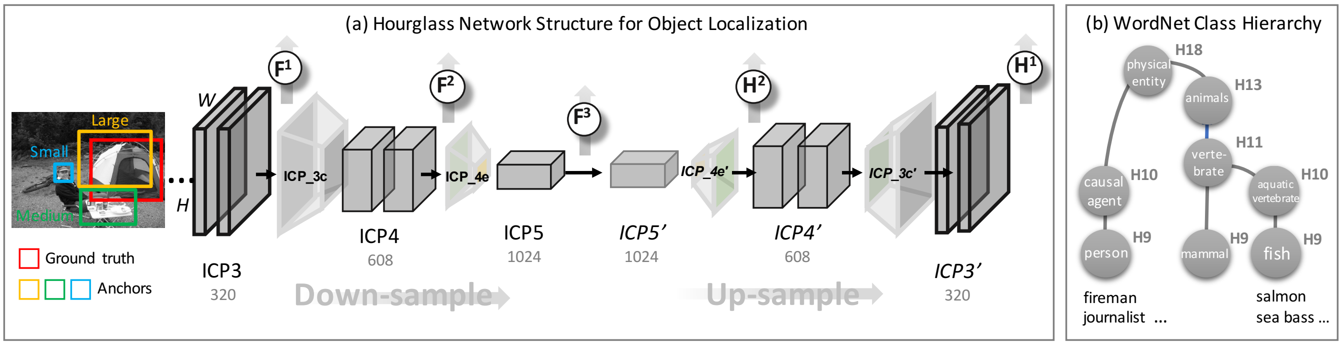

Fig. 2. (a) Hourglass network architecture for object localization. Different-sized anchors are placed at different resolutions of the network, fully leveraging the information of feature maps especially for small objects. Number under each module name (say ICP5) indicates the channel number. (b) An example of the hierarchy in WordNet categories. 'H18' means the height from a leaf node in the tree. Orange circles are the original 1000 classes in ImageNet classification set, where we further divide into many sub-classes and propose the extended dataset.

and `ICP5a-5b`. Max-pooling is placed after `ICP3c` and `ICP4e`. The up-sample architecture is exactly the mirrored version of the down-sample part with max-pooling being replaced by deconvolution. We denote `ICPx'` as the mirrored version of its counterpart `ICPx`.

The localization task is formulated in the region proposal network framework [2] where a set of pre-defined anchors (boxes) are placed at the feature map. In our design, three set of anchors, namely, large, medium and small are placed at the higher, middle and lower layers in the network respectively, corresponding to different resolutions of the feature maps. In this way, small anchors could still catch important details in the lower layers, whereas in previous work all anchors reside in the same layer and thus small anchors could be missed. In this paper, we split the set of anchor candidates $\mathcal{A}$ into three clusters: $\mathcal{A} = \{\mathcal{A}^m\}, m = 1, \cdots, 3$. The scales of anchors in each level are $\{16, 32\}$, $\{64, 128\}$, $\{256, 512\}$, respectively. As is illustrated at the top of Figure 2, we denote the feature maps from `ICP3b`, `ICP4d` and `ICP5b`, as $\mathbf{F}^1$, $\mathbf{F}^2$ and $\mathbf{F}^3$; their mirrored counterparts from `ICP3b'` and `ICP4d'` as $\mathbf{H}^1$ and $\mathbf{H}^2$. The combination of feature maps $\mathbf{F}^m$ and $\mathbf{H}^m$ on level m are implemented via a convolution operation:

$$\mathbf{G}^m = \sigma(\mathbf{w}_F^m \otimes \mathbf{F}^m + \mathbf{w}_H^m \otimes \mathbf{H}^m + \mathbf{b}^m), \tag{1}$$

where $\mathbf{G}^m$ are the merged feature maps for loss input.

3.2. Training loss and inference

Let $L^m(p_i, t_i, k_i^*, t_i^*)$ be the loss for sample i on resolution level m, where $p_i = \{p_{i,k} | k = 0, \dots K\}$ is the estimated probability, t_i indicates the estimated regression offset, K is the total number of classes[3], k_i^* denotes the ground-truth class label, and t_i^* represents the ground-truth regression offset. The network is trained using the following loss function:

$$\begin{aligned} & L^m(p_i, t_i, k_i^*, t_i^*) \\ & = -\frac{1}{N_1^m} \sum_i \log p_{i,k_i^*} + \frac{1}{N_2^m} \sum_i [k_i^* = 1] \mathcal{S}(t_i^*, t_i), \end{aligned} \tag{2}$$

where $\mathcal{S}$ is the smoothed $\mathcal{L}_1$ loss between ground truth target t_i^* and predicted target t_i, which is defined in [3]. N_1^m and N_2^m are the batch size of classification and regression on level m, respectively. Therefore, the total loss is defined as the summed loss across all levels: $L = \sum_{m=1}^{M} L^m(p_i, t_i, k_i^*, t_i^*)$, where M is the number of resolution levels. There are several remarks regarding the training in practice: (a) *Adjust image scale during training*. Each image is resized to the extent where at least one of the ground truth boxes is covered by anchors from $\mathcal{A}^m$. (b) *Control the number of negative samples in a batch*. We strict the number of negative samples to be twice the number of positive ones. (c) *Additional gray category*. We find adding an additional gray label (thus K=2) will better separate the positive from the negative. The number of gray samples is set to be half of the total number of positive and negative ones.

During inference, we take an inner-level and inter-scale NMS [24] scheme. Since the scale varies dynamically during training, we also forward the network in several scales, ranging from 1400 to 200 with an interval of 200. For a certain scale, we concatenate output boxes from all the levels and conduct an inner-level NMS with a threshold of 0.7; then we merge results from all scales and perform an inter-scale NMS with a threshold of 0.5.

4. EXPERIMENT

4.1. Setup and evaluation metric

We pretrain an Inception-BN [23] on the EIC dataset, which could achieve around 79% top-5 accuracy. Inception-BN is

[3] Since there is only one class in each image, $K = 1$.

Table 2. Component analysis on the hourglass structure. We use 30 anchors and treat training as a two-class problem.

Structure	Rec@0.5
Down-sample alone	89.25
Down-sample + splitAnc	87.94
Deeper down-sample + splitAnc	92.33
Deeper hourglass	94.51

Table 3. Component analysis on the anchor design and sampling scheme. We use the hourglass network in all settings. '×× ↑' denotes absolute increase of recall by each individual strategy (see Section 3.2 for details) based on and compared to the '30 ac. + dyTrainScale' setting. 'all' means adopting all strategies together.

Scheme	Rec@0.5	AR@300
9 anchors (short for ac.)	87.33	-
30 ac.	94.51	-
30 ac. + dyTrainScale	95.33	59.34
+ ctrlNegRatio	↑ 1.78	-
+ grayCls	↑ 1.13	-
30 ac. + all	97.81	68.45

used as the first down-sampling part of our network and the mirrored up-sampling part is also initiated from the pretrained model. The base learning rate is set to 0.0001 with a 50% drop every 7,000 iterations. The momentum and weight decay is set to be 0.9 and 0.0005, respectively. The maximum training iteration is 200,000 (roughly 8 epochs). We use a batch size of 300 with each class having at most 100 samples. Thirty anchors are used with scale increase from 16 to 512 exponentially and aspect ratio being [0.15, 0.5, 1, 2, 6.7] on our proposed dataset. We use recall under different IoU thresholds and number of proposals as the main metric. The mean value of recall from IoU 0.5 to 0.95 is known as *average recall* (AR). AR summarizes the general proposal performance and is shown to correlate with the average precision (AP) performance of a detector better than other metrics [25].

4.2. Component analysis

Table 2 depicts the analysis on network design. Rec@0.5 is the recall at IoU threshold 0.5 using top 300 proposals, evaluated on EIC validation set. Using the down-sample part alone, or splitting anchors (splitAnc) into different resolutions in the network, we have lower recall. If we increase the depth to the same as final hourglass structure (third case), the performance goes better; however, by carefully merging feature maps of different resolutions, we have the highest recall of 94.51, which proves the effectiveness of such a structure. Table 3 reports the performance gain where several strategies during training are adopted.

4.3. Investigation on training data

Finally, we investigate the necessity of adding more data for training a deep learning model. Table 4 describes the results under different training data schemes. The balanced case uses around two thirds of the original data on both datasets, respectively, as the number of images in most categories is 1300. For some classes that do not have enough images to reach 700, we manually augment the samples by translating, jittering the color channels, rotation, etc. Note that all settings use the same learning policy and stopping criteria.

Table 4. Investigation on different training data strategies. All results are evaluated on ILSVRC CLS 2012 validation set. base means during each iteration, the model is trained with samples in the order of object class; random indicates the average result from five randomly shuffled image lists, where class order among iterations is irrelevant; balanced denotes that based on the random sampling, for each epoch, the total amount of samples in each class is exactly the same 700.

Training data strategy	AR@10	AR@100	AR@500
ILSVRC_1k, base	38.45	50.02	54.72
ILSVRC_1k, random	53.76	65.21	76.67
ILSVRC_1k, balanced	52.17	66.58	75.32
EIC, base	42.19	46.73	49.01
EIC, random	**59.31**	71.82	78.56
EIC, balanced	58.72	**72.39**	**81.27**
Selective search [26]	45.82	57.63	69.45
GOP [27]	52.66	63.21	74.93

Several remarks can be drawn from the table. First, a larger dataset (EIC vs ILSVRC_1k) is beneficial to gain better results as more simples will ease overfitting if the model capacity is large. Second, the base ordering is inferior for training the neural network as the model will severely bias towards direction in the feature space due to continuous samples of one class. Third, a random sampling scheme ensures the classifier can witness various samples and the weights are quickly learned separately for each class, making the model robust and easy to converge. At last, we find the amount of training data is not the most crucial point for obtaining a better model, but rather a good balance of the distribution among training samples weigh more.

5. CONCLUSION

In this paper, we propose the Extended ImageNet Classification dataset to investigate whether more training data is beneficial towards a good deep learning model. We address the object localization problem by applying a conv-deconv structure in the region proposal framework, allowing different sizes of anchors placed at various depth in the network. Such a design could best leverage feature information, of which are summarized semantics at higher layers and preserved details at lower ones. Experiment results show that more training data is good, and yet a balanced data distribution could achieve better results at the cost of less data. In the future, we will investigate the data concern on more challenging datasets.

6. REFERENCES

[1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NIPS*, 2012, pp. 1106–1114.

[2] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in *NIPS*, 2015.

[3] Ross Girshick, "Fast R-CNN," in *ICCV*, 2015.

[4] Hongyang Li, Yu Liu, Wanli Ouyang, and Xiaogang Wang, "Zoom out-and-in network with recursive training for object proposal," *arXiv preprint:1702.05711*, 2017.

[5] Yu Liu, Hongyang Li, and Xiaogang Wang, "Learning deep features via congenerous cosine loss for person recognition," *arXiv preprint:1702.06890*, 2017.

[6] Hongyang Li, Jiang Chen, Huchuan Lu, and Zhizhen Chi, "CNN for saliency detection with low-level feature integration," *Neurocomputing*, vol. 226, pp. 212–220, 2017.

[7] Hongyang Li, Wanli Ouyang, and Xiaogang Wang, "Multi-bias non-linear activation in deep neural networks," in *ICML*, 2016.

[8] Zhizhen Chi, Hongyang Li, Huchuan, and Ming-Hsuan Yang, "Dual deep network for visual tracking," *IEEE Trans. on Image Processing*, vol. 26, pp. 2005–2015, 2017.

[9] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, 2006.

[10] 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.

[11] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollar, "Microsoft COCO: Common Objects in Context," *arXiv preprint:1405.0312*, 2014.

[12] Hongyang Li, Peng Su, Zhizhen Chi, and Jingjing Wang, "Image retrieval and classification on deep convolutional sparknet," in *IEEE International Conference on Signal Processing, Communications and Computing*, 2016.

[13] Zhizhen Chi, Hongyang Li, Jingjing Wang, and Huchuan Lu, "On the importance of network architecture in training very deep neural networks," in *IEEE International Conference on Signal Processing, Communications and Computing*, 2016.

[14] Bharath Hariharan, Pablo Arbelez, Ross Girshick, and Jitendra Malik, "Hypercolumns for object segmentation and fine-grained localization," in *CVPR*, 2014.

[15] Alejandro Newell, Kaiyu Yang, and Jia Deng, "Stacked hourglass networks for human pose estimation," in *ECCV*, 2016.

[16] Neelima Chavali, Harsh Agrawal, Aroma Mahendru, and Dhruv Batra, "Object-proposal evaluation protocol is 'gameable'," in *CVPR*, 2016.

[17] Zeeshan Hayder, Xuming He, and Mathieu Salzmann, "Learning to co-generate object proposals with a deep structured network," in *CVPR*, 2016.

[18] Weicheng Kuo, Bharath Hariharan, and Jitendra Malik, "DeepBox: Learning objectness with convolutional networks," in *ICCV*, 2015.

[19] Philipp Krahenbuhl and Vladlen Koltun, "Learning to propose objects," in *CVPR*, 2015.

[20] Amir Ghodrati, Ali Diba, Marco Pedersoli, Tinne Tuytelaars, and Luc Van Gool, "DeepProposals: Hunting objects and actions by cascading deep convolutional layers," *arXiv preprint: 1606.04702*, 2016.

[21] George A. Miller, "Wordnet: A lexical database for english," *Commun. ACM*, vol. 38, no. 11, pp. 39–41, Nov. 1995.

[22] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations*, 2015.

[23] Sergey Ioffe and Christian Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *ICML*. 2015.

[24] Bogdan Alexe, Thomas Deselaers, and Vittorio Ferrari, "Measuring the objectness of image windows," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2189–2202, Nov. 2012.

[25] J. Hosang, R. Benenson, P. Dollár, and B. Schiele, "What makes for effective detection proposals?," *IEEE Trans. on PAMI*, 2015.

[26] J.R.R. Uijlings, K.E.A. van de Sande, T. Gevers, and A.W.M. Smeulders, "Selective search for object recognition," *International Journal of Computer Vision*, 2013.

[27] Philipp Krahenbuhl and Vladlen Koltun, "Geodesic object proposals," in *ECCV*, 2014.