Patch-based Classification on Low Resolution Images

Wei Cheung [1] **Xiaojun Sun** [1] **Tian Du** [1]

Abstract

The state-of-art image classification methods can learn subtle details between visually similar classes, however low-resolution data source poses a significant challenge. Meanwhile, low-resolution images exist in various industry fields given the actual physical conditions as well as the limited information capturing devices. In addition, classifying low resolution images is critical for developing robust, resolution-agnostic classification systems. This project attempts to explore the BagNet architecture on low-resolution images, since it has been proven to be effective on ignoring the spatial ordering of the image features, which might help when classifying low-resolution imagery.

1. Introduction

Although there is impressive widespread improvements of image classification network architectures, the problem of classifying low resolution images remains highly challenging. While image classifiers performance on standard datasets such as CIFAR-10, MNIST or ImageNet are quite good, classification accuracy on more complex low resolution datasets is significantly worse. Are performance from high resolution image classifier benefits due simply to training set resolution? If the classification accuracy is aided by high image resolution, how can we use this knowledge to improve low resolution image classification performance? In this work, we explore these questions in detail and seek to improve low resolution image classification performance.

2. Related Work

Image processing by extracting features from local areas on images is widely researched field in computer vision. Traditional approaches include the Scale-Invariant Feature Transform (SIFT), which involves extracting from a set of reference images and storing them in a database, and then comparing new images to features in this database. It is widely applied in fields like object recognition, 3D modeling, robotic mapping and navigation and video tracking. Features from accelerated segment test (FAST) is another approach to extract local features that by corner detection which is commonly applied in video processing work because of its high-speed advantages. Unsupervised learning approaches on spatially distributed data has also been researched on, where representations of geospatial data is learned with unsupervised methods, to improve predictions in downstream tasks such as classification.

Once local features are extracted, methods to combine local features to make global predictions also have been researched. Linear models or averaging of local features is a natural option experimented in BagNet. Kernal method is another approach, and has shown that linear kernels, as well as other additive class of kernels can produce decent performance, while the former is much more efficient to train. A study on ultra high resolution Whole Slide Tissue Image explored by using multi-class logistic regression and SVM on patch features extracted from Convolutional Neural Networks, and showed that patch-based models can outperform traditional image-based models.

3. Dataset and Features

The Tiny ImageNet Dataset is created for Stanford CS231N. It is similar to the ImageNet data. It has 200 classes, and each class has 500 training samples, 50 validation samples and 50 test samples. All images are of size 64 x 64. The labels include both the class and bounding box for each image, but for the purpose of this project we are only aiming to predict the class of the images.

4. Methods

Our modeling approach consists of two steps: (1) first we use multiple stacked ResNet blocks to extract feature representation in the form of a 2048-dimension vector from each patch (of size p x p pixels) in a sample image, and then (2) we apply a linear classification model over these features to make a global prediction on the whole image.

4.1. Extract features from patches of an image

We use a Deep Neural Network architecture that largely resembles the ResNet-50 architecture and the major difference is that we used 1x1 convolutions instead of many 3x3 convolutions, therefore the dimension of the first convolutional

layer to p x p pixels.

4.2. Apply linear classifier on top of local feature representation

Here we used a combination of simple averaging and a linear classifier on top of the aggregated features. This linear structure allows us to identify which local image patches contribute most to the global image prediction, as shown in the heatmaps we produce in the following session.

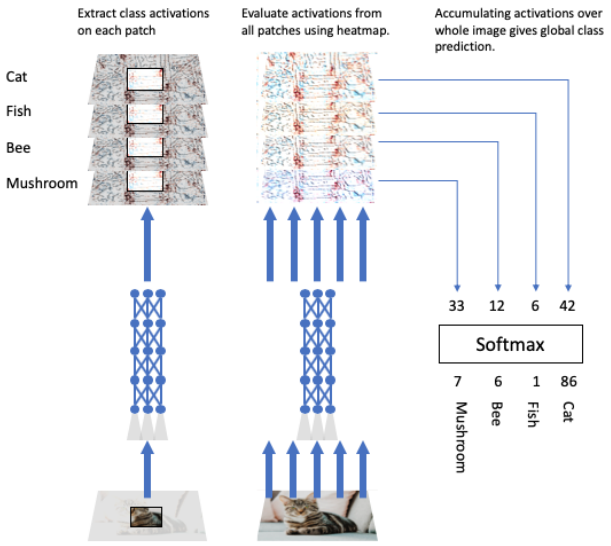

Figure 1. Architecture of the BagNet

5. Experiments/Results/Discussion

5.1. Experiments

We implemented the BagNets in PyTorch with 3 different variants, namely, BagNet-9, BagNet-17, and BagNet33. The numbers denote the receptive field size of the topmost convolutional layer as q by q pixels, which is also the patch size of the area to be integrated. Because Tiny ImageNet has a total of 200 classes so we modified the network by replacing the final fully connected layer so that it can output the results based on 200 classes. The models were then evaluated on multiple Google Cloud Platform (GCP) instances using NVIDIA Tesla K80 GPU. To better explore the utilization of BagNets on the low-resolution image dataset such as Tiny ImageNet, we conducted control experiments to evaluate the impact of patch size, optimizer, and inference mode on model performance as a guide for future implementation of our model on other applications. Specifically, we fixed the number of training epochs to either 5 or 15 due to limited computing resources. We also chose the mini-batch size of 100 images per batch to achieve higher training efficiency while avoiding the exceeding the GPU memory constraint. The cross-entropy loss was selected as the loss function for all the experiments because it often performs well on multi-class classification tasks and measures how well the softmax output is. We then fine-tune the learning model with patch size (9, 17, and 33), optimizers (Adadelta, Adam, and SDG), and inference mode (pre-trained parameters or trained from scratch). Additionally, we applied the ResNet18 as a baseline model to compare against the bagNet models. To start with, we trained ResNet18 using SDG with a learning rate of 0.001, which will decay by a factor of 0.1 every 7 epochs. The best validation accuracy for the model with and without pre-trained weights are 0.5714, and 0.2594, respectively. This result meets expectation as the ResNet18 was trained on ImageNet and Tiny ImageNet can be viewed as a downsampled version of original ImageNet dataset so the pre-trained weights help to boost the accuracy.

5.2. Role of patch size

We then compared the performance of the three variants of BagNet with patch sizes of 9, 17, and 33. Experiments 4, 6, 10 in table 1 summarized the experimental details where we used SDG as the optimizer, cross-entropy as lose function, 15 training epochs, and models were trained from scratch only on Tiny ImageNet. The best validation accuracy of the 3 models was 0.0052, 0.0258 and 0.0212, which are relatively low comparing to baseline models. We suspect that the BagNet overlooked spatial or global relationships while only focusing on local features. Also, the low-resolution images posed a difficulty for feature detection, it may also be possible that SDG doesn't work well with such tasks.

5.3. Role of optimizer

To validate our hypothesis that optimizer may impact the learning performance of the models, we conducted another set of experiments where we compared SDG, Adadelta, and Adam on BagNet-9, BagNet-17 and BagNet-33. This set of experiments (4, 5, 6, 8 10 12, 14 16, and 17, 14) fixed number of epochs for training, initial learning rate, and the decay learning rate. The models were all trained from scratch. We saw a consistent trend that both Adadelta and Adam outperforms SGD significantly in all three BagNet variants. The best validation accuracy trained using Adadelta is 0.317, 0.4065, and 0.4395 for BagNet-9, BagNet-17, and BagNet-33. Additionally, models trained by Adam scores slightly better than those from Adadelta, the best accuracy is 0.3162, 0.4234, and 0.4432, respectively. Considering that this learning model was only trained on Tiny ImageNet (limited number of training samples and low image resolution), this accuracy is relatively high. We hypothesized that Adadelta and Adam may converge faster than SDG for such tasks as both were designed specifically for training neural networks and were extensions of adaptive gradient algorithms.

Table 1. Comparison of Different Learners Models

MODEL	PATCH SIZE	OPTIM IZER	PRE TRAIN	EPOCH	BEST VAL ACCURACY
RN18	NA	SGD	N	15	0.2594
RN18	NA	SGD	Y	15	0.5714
BAGNET	9	SGD	N	5	0.0062
BAGNET	9	SGD	N	15	0.0052
BAGNET	9	ADAM	N	15	0.3162
BAGNET	9	ADADELTA	N	15	0.317
BAGNET	17	SGD	N	5	0.0169
BAGNET	17	SGD	N	15	0.0258
BAGNET	17	ADAM	N	5	0.2359
BAGNET	17	ADAM	N	15	0.4234
BAGNET	17	ADAM	Y	5	0.3522
BAGNET	17	ADADELTA	N	15	0.4065
BAGNET	17	ADADELTA	Y	5	0.4000
BAGNET	33	SGD	N	15	0.0212
BAGNET	33	ADAM	N	5	0.1653
BAGNET	33	ADAM	N	15	0.4432
BAGNET	33	ADADELTA	N	15	0.4395

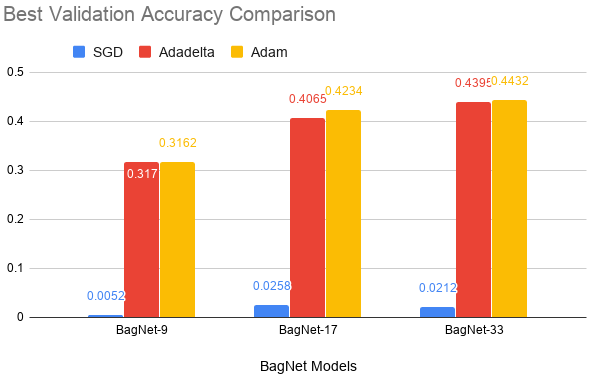

Figure 2. Best validation accuracy comparison between BagNet-9, BagNet-17, and BagNet-33 using SDG, Adadelta, and Adam optimizer. Models were all from scratch for 15 epochs. Initial and decayed learning rate was fixed for Adam and SGD

5.4. Discussion

As demonstrated by the baseline ResNet18 model, pre-trained ResNet outperforms the vanilla ResNet18 network, likely because the pre-trained weights were obtained from ImageNet dataset, which can help the recognition of Tiny ImageNet images. In this project, it's computationally expensive to train the BagNet on ImageNet from scratch so we only trained the model based on Tiny ImageNet training set and used data augmentation methods. Nevertheless, we found that BagNet outperforms the ResNet (both without pre-trained weights) with an accuracy of 0.4065 comparing to 0.2359, suggesting that BagNet works well on low-resolution datasets and the concept of feature extraction coupled with linear classifier can achieve comparable or even better results than ResNet18. We also demonstrated that the choice of optimizer and patch size both impact the learning process and eventually model performance. We summarize the key points as below:

- Adadelta and Adam both outperforms SGD significantly and yields best validation accuracy of about 43 percent
- Best validation accuracy increases with the increase of BagNet patch size
- Best validation accuracy can be further improved if BagNets are pre-trained with ImageNet dataset.

6. Future Work

The future work on this topic will be on fine-tuning the current architecture to perform classification on both low and high resolution images. Or in other words, a resolution-agnostic network architecture. The challenge of designing a multiple-resolution image classifier is to maximize the performance in both the target and source domains. There are several attempts we could try:

- use super-resolved / super-resolution image to enhance the image feature before feed the data into the network
- use adversarial training approach that has been proven to be resistant to small changes in the input image that are imperceptible to the human eye

7. Contributions

Wei Cheung: Conducted heatmap analyses, experiments on hyper-parameters to optimize model performance. Researched on datasets, model approaches.

Xiaojun Sun: Explored feature extraction methods SIFT, FAST, and ORB. Evaluated learning models with different patch sizes, optimizers, and inference mode.

Tian Du: Exploration and implementation of the BagNet architecture. Evaluated and compared different models configurations.

References

Brendel, W. and Bethge, M. Approximating cnns with bag-of-local-features models works surprisingly well on imagenet. *International Conference on Learning Representations*, 2019. URL https://openreview.net/pdf?id=SkfMWhAqYQ.

Hou, L., Samaras, D., Kurc, T. M., Gao, Y., Davis, J. E., and

Saltz, J. H. Patch-based convolutional neural network for whole slide tissue image classification. *Institute of Electrical and Electronics Engineers*, 2016.

Jean, Wang, Samar, Azzari, Lobell, and Ermon. Tile2vec: Unsupervised representation learning for spatially distributed data. *Association for the Advancement of Artificial Intelligence*, 2018.

Lowe, D. G. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60 (2), 1994.

Morrow and Shankar. Efficient additive kernels via explicit feature maps. *Institute of Electrical and Electronics Engineers*, 2009.