Applying Hierarchical Mixture of Experts to Multi-Class Image Classification

Frank Yu
Department of Mechanical and Mechatronics Engineering
University of Waterloo
Toronto, Canada
q.frank.yu@gmail.com

***Abstract*—This paper proposes applying hierarchical mixture of experts model to image classification. This ensemble method can be utilized to combine various expert image classifiers to perform classification on a wider number of image classes. The main benefits of this approach is faster training and better interpretability in comparison to deep learning models and other ensemble techniques. In addition, minor test accuracy improvements were observed during experimentation on multiple datasets.**

***Index Terms*—Artificial neural network, ensemble methods, image classification, hierarchical mixture of experts**

I. Introduction

Multi-class image classification is a type of visual object recognition problem that is widely popular in machine learning. Such problems are often posed in the form, ”attribute a previously unseen image with one of several object labels”. Currently, the domain of image classification is dominated by Deep Learning (DL) approaches, often in the form of Convolutional Neural Networks (CNN). DL models are appealing due their ability to identify intricate structures from raw high-dimensional datasets such as images [3]. First proposed by LeCun et al. in 1998 [2], CNNs utilize multiple convolution layers to extract higher level features from images. Deep CNNs such as AlexNet [4] and VGG-19 [5] have proven success on ImageNet, a large multi-class image dataset.

Since DL models are end-to-end solutions, they operate as black boxes which has drawbacks. The training process for deep CNNs is data and time extensive [1]. Even with developments in GPU technology, training deep CNN models can take hours or even days [4]. As well, it is difficult to optimize components of an end-to-end solution in isolation. Thus, the iteration time for making changes and testing DL model is long. Furthermore, their black-box nature makes interpreting results difficult, limiting their usability in fields such as medicine [6]. As well, poor interpretability makes CNN optimization difficult as wrong classifications are difficult to diagnose. To overcome these shortcomings, I propose a method of incorporating principles from Hierarchical Mixture of Experts (HME) to create an ensemble HME CNN image classifier.

HME is an ensemble method that organizes classifiers into into a soft decision tree structure using conditional probability [7]. HME ensembles are comprised of expert classifiers which are trained on a subset of all available training data making them optimized for specific data instances. The proposed method for applying HME to multi-class image classification has three stages: 1) I propose a method of identifying groups of image classes, or super classes, that exhibit high interclass separation from other such super classes. These super classes form the basis of segmenting classification functionality between multiple classifiers. This process could be performed recursively on super classes to identify child super classes. This leads to a super class tree. 2) Expert CNN classifiers are assigned super classes and are trained to classify images within their super class. They could classify them either into individual classes or other super classes. They are trained only using the subset of training images from their super class. 3) Expert CNN classifiers are integrated into an HME ensemble. Each individual CNN prediction on unseen data is normalized into probability values using softmax [9]. Conditional probability is used to consolidate prediction probabilities between the different experts into a single output class.

The motivation for utilizing HME principles is to address the poor interpretability and long training time required for deep CNN classifiers. The use of expert classifiers enables decoupling of classification functionality; large classification problems can be decomposed into smaller classification problems that can be tackled in parallel. Furthermore, it enables different models, training, data augmentation, or optimization techniques to be applied to different subsets of image classes. HME also enforces a explicit soft decision tree structure which is highly interpretable, providing greater visibility over classification results. While each individual CNN still operates as a black box, tracing through the soft decision tree enables identification of bottlenecks in classification performance. Note that improvements in classification accuracy is not a direct goal of HME CNN. Rather, it is hoped that better, more powerful classifiers can be developed by improving interpretability and reducing training time.

Experimental results indicate minor test accuracy improvements across two multi-class image datasets. They also indicate that expert CNNs converge to high validation accuracies faster than single CNN classifiers. As well, performance bottlenecks can be identified amongst the expert CNN classifiers used in the experiment which displays the desired interpretability benefits.

II. Related Works

Hierarchical Mixture of Experts was proposed in 1994 [7] and uses conditional probability to combine expert classifiers using gating classifiers. This pattern could be used recursively such that expert classifiers could also function as gating networks to additional expert classifiers, as shown in Figure 1.

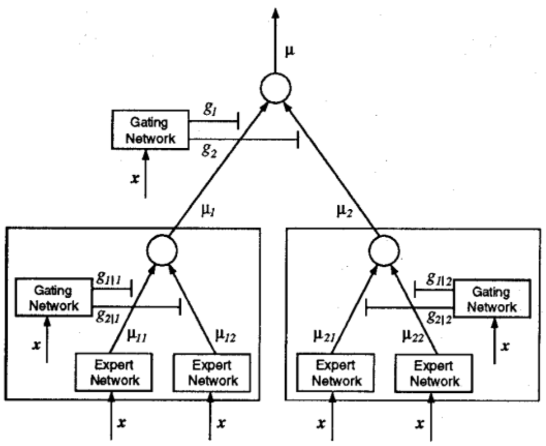


Fig. 1. Hierarchical Mixture of Experts Model [7]

This ensemble scheme has been applied to NLP [16] and signal classification [15]. While the HME model is meant to train as an ensemble, the proposed HME CNN trains each individual expert CNN classifier in isolation.

Other ensemble techniques also exist to improve classification performance. One such technique is bagging [17] which is a voting based approach for aggregating classifications amongst multiple models. This approach has proven success in computer vision tasks [20] including hand pose estimation [19]. Another ensemble technique is boosting which aims to create a series of classifiers; in each iteration, misclassified examples from the previous classifiers are sampled with greater frequency during training [21]. Again, this approach also has proven success in computer vision tasks [20]. Both techniques depend on the variance in training data between different classifiers. As well, both produce multiple classifiers which are able to perform the entire multi-class classification task individually in addition to operating as an ensemble.

The proposed HME CNN is different in that each expert classifiers within the hierarchy cannot perform the entire multi-class classification task in isolation. However, the benefit is that each expect only requires a subset of the data for training. By creating multiple performant classifiers, both bagging and boosting approaches exacerbate problems suchs long training times and poor interpretability. Where the goal of these approaches is to improve test accuracy directly, the path for HME CNN improving accuracy is more indirect.

III. HME CNN Architecture

The HME CNN is comprised of several CNN classifiers organized into a tree like structure. The number of CNNs and the ensemble structure depends on the nature of the image dataset. Assume there is an image dataset which 8 classes of images with labels 0-7.

A. Identifying Image Super Classes

The first step is to identify groups of image classes that exhibit high interclass separation from other image classes. Let such image classes be known as image super classes. Several algorithms exist measure image similarity between different images. Point intensity approaches that compare images pixel by pixel often fail to match images after rotations or translations. Other algorithms extract features or descriptors from different images and measure similarity based on these descriptors. One such approach is measuring the similarity of different images based on their histograms. However, histograms fail to capture structural information created by edges within an image.

Perhaps other descriptors such as SIFT [11] or SURF [12] would be more successful for identifying similarity. However, empirically it was found that the simpler method of manually selected image classes that look similar formed a good bases for super classes. Hence, rather than pursuing an algorithmic approach, the proposed method for identifying image super classes utilizes human intuition; group similar classes by manually profiling datasets. Other useful techniques include the fact that often related items such as different types of birds within an animal image dataset would exhibit similar features and textures. As well, tools such as confusion matrices can be used with weak classifiers to identify trends in misclassifications. Consistent misclassification between two images classes would suggest poor interclass separation. Metrics such as Top-K classifications can also be used.

To validate the selected super classes are do in fact exhibit large interclass seperation, a typical CNN image classifier will be used. While CNN model classify image well, their poor interpretability makes them unsuitable for identifying super classes. However, once super classes are chosen, they are a useful tool for validation. First, train and test a CNNs to classify between all classes. Next, remap the dataset so that all classes within the same super class have the same label. Using this new dataset, train and test the same CNN model to classify images into their super classes. If interclass separation is high, the accuracy of the super class classification should be much higher than the single class classification.

This procedure can be used recursively to further decompose classes within each super class. Some super classes will only contain one element if the image class is highly distinct from all others. It is not required to continue performing this process until all super classes only contain a single element. Performing this process should result in a tree consisting image classes and their parent super classes. Let this be known as the super class tree. Figure 2 shows one potential super class tree for the example dataset with 8 image classes.

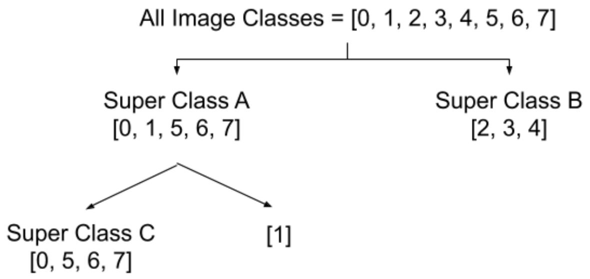


Fig. 2. Potential Super Class Tree with 8 Image Classes

B. Creating Expert Classifiers

Expert classifiers will be required to classify images into super classes. Furthermore, expert classifiers will also be required to classify images within each super class leaf node. Based on the super class tree shown in Figure 2, Figure 3 shows the required expert classifiers along with their classification responsibility. In the example, all expert classifiers are shown as CNNs since they have proven success in image classification tasks, however, they can be replaced by other types of classifiers.

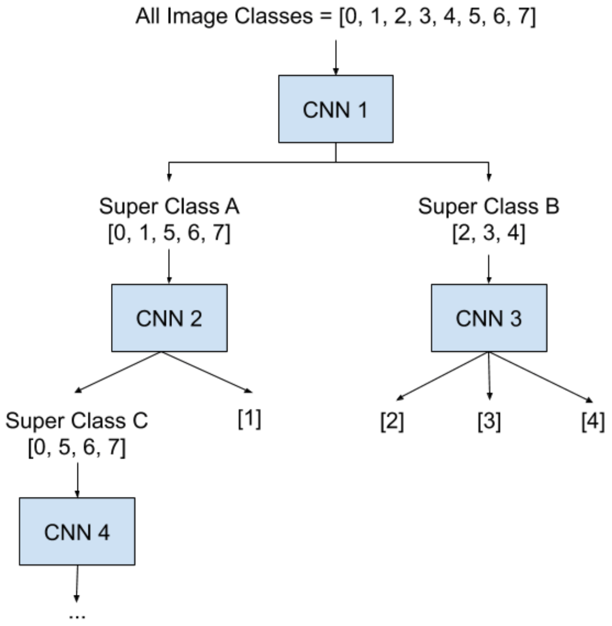


Fig. 3. Potential Ensemble Structure

The exact architecture and training process for each expert classifier would depend on the super class of images that it is required to classify. All expert classifiers are trained in isolation from each other. Expert classifier should only be trained on image classes relevant to them. Hence, the total image training set must be filtered and remapped for each expert classifier: all images of classes that are outside of the classifiers' expertise should be filtered out. Afterwards, labels for the remaining images should be remapped based on the expert classifier's output. Looking at Figure 3, CNN 2 would only be trained on images belonging to super class A and have 2 outputs; it classifies images as either belonging to super class C or class 1.

C. Building the Ensemble

After all expert classifiers are trained, their classifications are combined through conditional probability. Given an previously unseen image I, I image is fed forward through all the expert classifiers. The output of each expert classifier is then normalized into probability using the softmax function [9]. Then, the probability that I has class $C = c$ is calculated for all potential class values c using conditional probability. The probability that an image belongs to a specific class c is conditional on whether the image belongs to its parent super class:

$$P(C = c) = P(C = c|C \in S_1)P(C \in S_1) \qquad (1)$$

Where S_1 is a parent of class c. This conditioning occurs recursively for super class trees with more than 2 levels. For instance, in the example shown in Figure 3, the probability that an image is in class 7 would be:

$$P(C = 7) = \\ P(C = 7|C \in S_C)P(C \in S_C|P \in S_A)P(c \in S_A) \qquad (2)$$

Where S_i denotes super class i. $P(C = 7|C \in S_C)$, $P(C \in S_C|P \in S_A)$, and $P(c \in S_A)$ would be obtained from the softmax output of expert classifiers CNN4, CNN2, and CNN1 respectively based on the example in Figure 3. Since labels were remapped during training, the inverse mapping process should be performed to translate labels back to their original class definitions.

IV. Experiments

Two different experiments were conducted to evaluate HME CNN classifier ensembles against single CNN classifiers. Each experiment was performed using a different publicly available dataset. The goal of these experiments is two fold: 1) To show that multi-class image classification tasks can be successfully decomposed into separate independent sub classification tasks, and 2) that an ensemble of expert classifiers, trained to solve each sub classification task, can be used in conjunction to compete with single CNN classifiers.

A. Datasets

1) Fashion-MNIST Dataset: This dataset contains images of different articles of clothing including bags and shoes. It is designed to pose a more challenging classification task than MNIST [8] and it is widely popular benchmark for multi-class

image classification models. There are ten classes of clothing articles: T-shirt/top, Trouser, Pullover, Dress, Coat, Sandal, Shirt, Sneaker, Bag, Ankle boot. Each image is 28 x 28 pixels monochrome. In total, there are 60,000 images for training and 10,000 test images. There is equal distribution of all images classes within the training set with 6,000 images each. The testing set is standardized, and Top-1 testing accuracy is the widely accepted performance metric [8]. Hence, the number of correctly classified images C is compared against the test set for total accuracy as follows:

$$Acc_f = \frac{|C|}{10000} \tag{3}$$

2) KIMIA Path960 Dataset: This dataset contains histopathology images obtained from whole slide images of various tissue [10]. There are 20 classes which correspond to visually selected tissue texture/patterns [10]. There are 960 images in total, 48 images of each tissue class. Each image is 308 x 168 pixels with RGB color channels. This dataset was selected due to its constrasting characteristics to the Fashion-MNIST dataset: Path960 images are coloured and larger. Furthermore, the dataset is smaller yet contains more classes than Fashion-MNIST. The average Top-1 test accuracy from K-fold validation was selected as the performance metric for this dataset with $K = 5$. To ensure image class balance, stratified k-fold was employed. Hence, the number of correctly classified images C compared against the test set T is averaged over 5 tests:

$$Acc_h = \frac{1}{5}\sum_{i=1}^{5}\frac{|C|}{|T|} \tag{4}$$

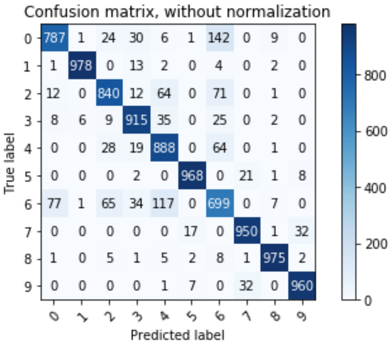
Where T varied between either 180 or 200 due to stratified k-fold.

B. Data Profiling and Super Class Identification

To identify super classes within Fashion-MNIST, I built a shallow CNN classifier which produced the confusion matrix shown in Figure 4. The confusion matrix shows frequent misclassifications between classes 0-4 and 6. Intuitively, this makes sense as these classes were all cloth based items such as shirts, pullovers, dresses etc. Furthermore, there were miss classifications between classes 5, 7, and 9 which are Sandal, Sneaker, and Ankel Boot respectively. The remaining class 8, which is Bags, did not seem to belong to either of these groups.

From these results, 3 image super classes were hypothesized which were Cloth with classes [0-4, 6], Shoes with classes [5, 7, 9] and Bag with class 8. Experimentally, the expert classifier for classifying between these three super classes was able to achieve $> 99\%$ test accuracy which is much higher than the public scores shown on the Fashion-MNIST github page[1].

Since the KIMIA Path960 images were visually selected, identifying super classes within these images was much easier. Moreover, classes with similar colors also exhibited similar structural features. Four super classes were identified: White [0, 4, 7], Blue [1, 15], Pink [2, 3, 5, 6, 8, 9, 10-14, 17-19] and Brown [16]. Experimentally, the expert classifier for classifying between these super classes was able to achieve 100% test accuracy.



	0	1	2	3	4	5	6	7	8	9
0	787	1	24	30	6	1	142	0	9	0
1	1	978	0	13	2	0	4	0	2	0
2	12	0	840	12	64	0	71	0	1	0
3	8	6	9	915	35	0	25	0	2	0
4	0	0	28	19	888	0	64	0	1	0
5	0	0	0	2	0	968	0	21	1	8
6	77	1	65	34	117	0	699	0	7	0
7	0	0	0	0	0	17	0	950	1	32
8	1	0	5	1	5	2	8	1	975	2
9	0	0	0	0	1	7	0	32	0	960

Fig. 4. Fashion-MNIST Confusion Matrix

C. Experiment Parameters

For each dataset, a single CNN classifier was compared to an HME CNN ensemble. In each experiment, the same CNN architecture was used for the single CNN classifier and each expert classifier in the HME CNN ensemble. The only exception was the output shape since expert classifiers had less outputs. Training parameters, such as number of epochs, image augmentations, etc were also standardized within each experiment. For training, 25% of the training data was siphoned off for validation. Batch normalization was used for inital convolution layers to prevent overfitting and improve learning speeds [13]. All classifiers were trained for 25 epochs. To prevent overfitting, the model with the highest validation accuracy was saved and loaded as the final trained model.

For Fashion-MNIST, the CNN model that was used had three convolution layers and two dense layers. Random horizontal flipping was used to augment training images as many successful CNN classifiers on the Fashion-MNIST github page employed it during training[2]. Based on the super classes found in the previous section, the HME CNN ensemble for Fashion-MNIST is shown in Figure 5.

For KIMIA Path960, the CNN model that was used had three convolution layers and one dense layer. Several image augmentations with proven success on medical images [14] were employed during training including rotations, scaling, translations, flips. Empirically, it was also found that filling empty portions of the image, created by augmentations, with a mirror reflection improved also accuracy. Based on super classes found previously, the HME CNN ensemble for KIMIA Path960 is shown in Figure 6.

D. Results

Tables I and II summarize experiment results for each dataset. The accuracy improvement for the Fashion MNIST is

[1] https://github.com/zalandoresearch/fashion-mnist

[2] https://github.com/zalandoresearch/fashion-mnist

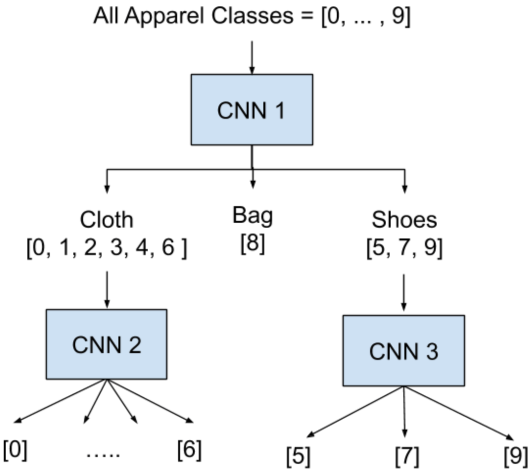


Fig. 5. Fashion-MNIST HME CNN

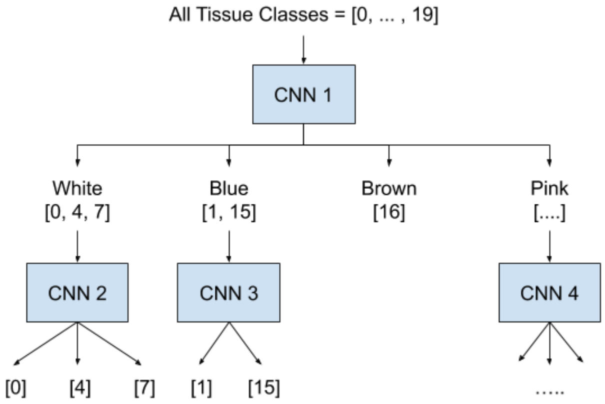


Fig. 6. KIMIA Path960 HME CNN

minor, while there was a 4.6% test accuracy improvment from utilizing a single CNN classifier to a HME CNN ensemble. Moreover, the HME CNN maintained this higher test accuracy with greater consistency as its test accuracy also had a lower standard deviation than a single CNN.

TABLE I
FASHION MNIST RESULT SUMMARY

Classifier	Test Accuracy
Single CNN	90.5%
HME CNN	91.1%

In addition to overall test accuracy, the test accuracy of each individual expert classifier was also collected with respect to the subset of test data belonging to their super class. Table III shows the test accuracy of each expert for Fashion-MNIST. Note, the names of the experts correspond to names shown in Figure 5. From this data, one can easily identify that the bottleneck in classification performance within this ensemble is CNN 2 which had a 10% lower test accuracy than the next lowest value. Overall, training for the CNN ensemble took longer due to overlaps in images between parent and child super class. For instance, CNN 1 and CNN 2 both trained on cloth image data. However, by identifying CNN 2 as the performance bottleneck, further model improvements and training would only need to be performed on CNN 2 which trains on a subset of image classes.

TABLE II
KIMIA PATH960 RESULT SUMMARY

K-Fold Iteration	Single CNN Test Acc	HME CNN Test Acc
1	89.5%	97.5%
2	92.0%	97.0%
3	93.5%	92.5%
4	82.2%	96.7%
5	98.3%	95.0%
Average	91.1%	95.7%
STD	0.05%	0.02%

TABLE III
FASHION MNIST EXPERT CLASSIFIER

Classifier	Test Accuracy
CNN 1	99.6%
CNN 2	87.3%
CNN 3	97.0%

During the experiments, it was also observed that expert classifiers trained much faster than the single CNN classifiers. Figure IV-D and IV-D show validation accuracy vs epoch for single CNN classifier and individual expert classifiers during training. Note that the expert classifiers in IV-D are named by their super class groups with root being the parent to all super classes. The data shows that most of the expert classifiers are able to converge to higher scores much earlier on than the single CNN classifier.

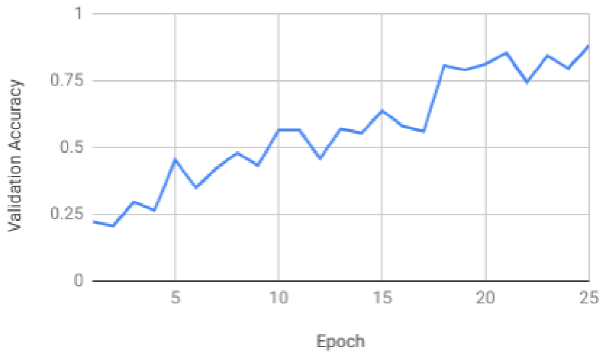


Fig. 7. KIMIA Path960 Single CNN Training

In general, HME CNN models achieved greater accuracy and potentially require less time to further their performance. While setting up the ensembles and training all the experts for the first time takes far longer than training a single CNN, this initial work is amortized as further model improvements would only involve a subset of experts.

V. DISCUSSION

The goal of project was to maintain the classification performance of deep CNN models while improving on their

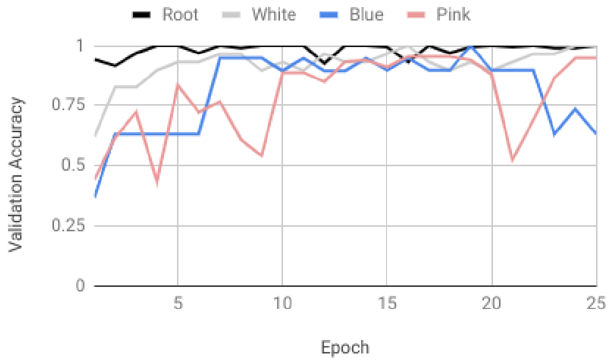


Fig. 8. KIMIA Path960 Ensemble Training

interpretability and reducing training time. While test accuracy improvements were observed in experiments, the HME CNN ensemble should not be able to capture more complex information than a single CNN; connecting the same CNN architecture into a two layer hierarchy is functionally the same as an additional dense layer before the output. Rather, the test accuracy improvement is more likely a testament to the decoupling of functionality. During the testing process, the best model based on validation accuracy is chosen as the final model. Since an ensemble contained multiple classifiers, the best version of each expert within the ensemble is chosen. In contrast, for a single CNN, the best version at classifying cloths vs shoes are coupled together, therefore the best average performer is chosen.

This could also explain why the single CNN classifies took longer to converge to validation accuracy than expert classifiers. Based on Figure IV-D, the accuracy of the single CNN could potentially improve with more training time. However, this additional training time is not free as it limits how many variations of models can be tested etc.

Moreover, to ensure a fair comparison, the experiments did not highlight potential benefits from modifying the training scheme between expert classifiers. For instance, some image classes are potentially invalid after certain image augmentations. Meanwhile, another class of images would greatly benefit from these augmentations to prevent overfitting. While this can potentially be resolved by augmenting different image classes separately, another benefit of decoupling would be using different models for different expert classifiers. For instance, there were large color differences between the super classes chosen for the KIMIA Path960 dataset. Hence, potentially a histogram based classifier could be used for simplicity over a CNN.

One downside to using the HME CNN ensemble is that predictions become more computationally expensive as images are fed forward through multiple CNNs. This could potentially be solved by implementing thresholds so that not all experts need to be consulted. This would reduce the search complexity within the tree of the ensemble.

The data shown in III illustrates the greater interpretability with respect classification performance. Also, tracing through the soft decision tree structure provides greater background information on classifications than single CNN classifiers.

REFERENCES

[1] N. Pinto, D.D. Cox, and J.J. DiCarlo. Why is real-world visual object recognition hard? PLoS computational biology, 4(1):e27, 2008.

[2] LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE 86(11) (1998) 22782324

[3] LeCun, Yann & Bengio, Y & Hinton, Geoffrey. (2015). Deep Learning. Nature. 521. 436-44. 10.1038/nature14539.

[4] Krizhevsky, Alex & Sutskever, Ilya & E. Hinton, Geoffrey. (2012). ImageNet Classification with Deep Convolutional Neural Networks. Neural Information Processing Systems. 25. 10.1145/3065386.

[5] Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)

[6] Miotto, R., Wang, F., Wang, S., Jiang, X., Dudley, J.T.: Deep learning for healthcare: review, opportunities and challenges. Briefings in bioinformatics (2017)

[7] Jordan, M.I.J., R.A., Hierarchical mixture of experts and the em algoritm. Neural computing, 1994. 6(181-214).

[8] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. arXiv preprint arXiv:1708.07747, 2017.

[9] Hinton, G.E., Vinyals, O., & Dean, J. (2015). Distilling the Knowledge in a Neural Network. CoRR, abs/1503.02531.

[10] M. D. Kumar, M. Babaie, S. Zhu, S. Kalra, H. R. Tizhoosh, ”A Comparative Study of CNN BoVW and LBP for Classification of Histopathological Images”, arXiv preprint arXiv:1710.01249, 2017.

[11] David GL, Distinctive Image Features from Scale-Invariant Keypoints, International Journal of Computer Vision, 2004

[12] HE Bay, ES Andreass, TU Tinne, LV Gool, ”Speeded-Up Robust Features (SURF)”, International Journal of Computer Vision and Image Understanding, vol. 110, no. 3, pp. 346-359, 2008.

[13] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In ICML, 2015.

[14] Hussain Z, Gimenez F, Yi D, Rubin D. Differential Data Augmentation Techniques for Medical Imaging Classification Tasks. AMIA Annu Symp Proc. 2018;2017:979984. Published 2018 Apr16.

[15] V. Ramamurti-J. Ghosh - 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings

[16] Andrew Estabrooks-Nathalie Japkowicz - Proceedings of the 2001 workshop on Computational Natural Language Learning - ConLL ’01 - 2001

[17] M. Korytkowski, L. Rutkowski, and R. Scherer, Fast image classification by boosting fuzzy classifiers, Information Sciences, vol. 327, pp. 175182, 2016.

[18] Breiman, L. (1996a). Bagging predictors. Machine Learning, 24(2), 123140

[19] Hengkai Guo-Guijin Wang-Xinghao Chen-Cairong Zhang-Fei Qiao-Huazhong Yang - 2017 IEEE International Conference on Image Processing (ICIP) - 2017

[20] D. Opitz and R. Maclin, Popular Ensemble Methods: An Empirical Study, Journal of Artificial Intelligence Research, vol. 11, pp. 169198, 1999.

[21] Freund, Y., & Schapire, R. (1996). Experiments with a new boosting algorithm. In Proceedings of the Thirteenth International Conference on Machine Learning, pp. 148156 Bari, Italy.