Neural Computing and Applications

Augmenting and Ensembling CNN Models to Increase the Accuracy of Alzheimer’s Disease MRI Classification

--Manuscript Draft--

Manuscript Number:	
Full Title:	Augmenting and Ensembling CNN Models to Increase the Accuracy of Alzheimer’s Disease MRI Classification
Article Type:	Original Article
Keywords:	CNN networks, Alzheimer's disease, augmentation, ensembling
Corresponding Author:	Kristin Levine, MS George Washington University Washington, DC UNITED STATES
Corresponding Author Secondary Information:	
Corresponding Author's Institution:	George Washington University
Corresponding Author's Secondary Institution:	
First Author:	Kristin Levine, MS
First Author Secondary Information:	
Order of Authors:	Kristin Levine, MS
Order of Authors Secondary Information:	
Funding Information:	
Abstract:	Lack of data is an on-going problem in Alzheimer’s disease (AD) research. AD can be hard to diagnose and even experienced neurologists struggle to read MRIs accurately. While many studies have used CNN models to classify MRI images, this study compares flipping and shifting images to determine which type of data augmentation is most helpful. This study also compares three different pre-trained CNN models (ResNet50, VGG16, and Xception) as well as an ensembled model. The models easily obtained excellent (99%) accuracy on the training/validation sets, even without augmentation, though test set accuracy lagged behind (65-73%). Shifting images improved test set accuracy by 2-5%, depending on the model. Ensembling the three models together improved test set accuracy by another 5-11%, depending on the model. Augmenting images to increase and balance datasets, as well as ensembling well-performing models, holds great promise in improving machine learning (ML) models to classify various stages of AD.

Augmenting and Ensembling CNN Models to Increase the Accuracy of Alzheimer's Disease MRI Classification

Kristin Levine

Advisor: Amir Jafari
The George Washington University, Washington DC
Data Science Program
ORCID #: 0000-0002-2099-6875

1 Abstract:

Lack of data is an on-going problem in Alzheimer's disease (AD) research. AD can be hard to diagnose and even experienced neurologists struggle to read MRIs accurately. While many studies have used CNN models to classify MRI images, this study compares flipping and shifting images to determine which type of data augmentation is most helpful. This study also compares three different pre-trained CNN models (ResNet50, VGG16, and Xception) as well as an ensembled model. The models easily obtained excellent (99%) accuracy on the training/ validation sets, even without augmentation, though test set accuracy lagged behind (65-73%). Shifting images improved test set accuracy by 2-5%, depending on the model. Ensembling the three models together improved test set accuracy by another 5-11%, depending on the model. Augmenting images to increase and balance datasets, as well as ensembling well-performing models, holds great promise in improving machine learning (ML) models to classify various stages of AD.

2 Introduction

Alzheimer's disease is the most common dementia, yet it remains difficult to diagnose. PET scans to measure amyloid protein build-up and cerebrospinal fluid (CSF) biomarkers can help to provide an accurate diagnosis, however they are expensive and invasive tests.[1] MRIs are much more accessible.

However, MRIs depend on an experienced neurologist to interpret them. A dependable and accurate ML model has the potential to help a general practitioner in a small town to read an MRI as accurately as a specialist at a large research hospital. Even with a trained specialist, one study found experienced radiologists only had an accuracy of 57.5 to 70% percent when reading an MRI to diagnose AD.[2] There has been progress in developing ML models for other conditions like breast cancer.[3]

In addition to diagnosing AD, being able to differentiate mild cognitive impairment from normal cognition is important for clinical trials, as potential treatments are more likely to be effective before the condition progresses to full-blown AD.

In the past, neurological ML models mainly used MRIs for feature extraction, i.e. to get a measurement for a part of the brain, such as the size of the hippocampus. While this has proved helpful, there is significant domain knowledge required in order to understand which parts of the brain to target. Extracting features is challenging; we also can't be sure that we are picking out the most helpful features.

CNNs offer a different approach of using computer vision to look at the image as a whole. There is less domain knowledge needed as the computer decides which are the relevant features. However, using a CNN network is not without its own challenges. CNN networks work best with lots of data; in dementia research there is a lack of labeled images. While some approaches have explored using unlabeled data to help train a network[4], the majority of researchers continue to use labeled datasets.

One way around a lack of data is data augmentation. In this paper, three different approaches are compared: using the original non-augmented dataset, using a flipped training set, and using a shifted training set.

Another method of dealing with a lack of data when creating a CNN model is to use transfer learning, i.e. using pretrained networks as a starting point for your model. There are plenty of other studies that have used these pretrained models to look at Alzheimer's imaging data.[5] [6] [7] In this study, three common models (ResNet50, VGG16, and Xception) will be used individually as well as combined, to try to quantify the gains that can be had by ensembing models.

3 Methods

A Kaggle dataset was used for this project: Alzheimer's Dataset (4 class of images). This dataset contains 6400 jpg images (slices) taken from MRI images of possible Alzheimer's patients, divided into 4 categories.

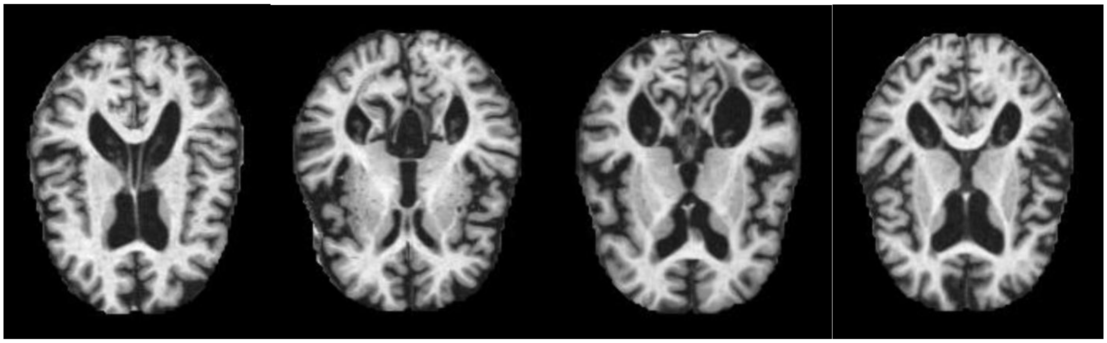


Figure 1: The four classes of images. From left to right: non-impaired, very mild impairment, mild impairment, moderate impairment

The original dataset was quite unbalanced, although this is typical for this type of research. As patients become more impaired, there is often less data available.

The train set was divided into train/validation sets, using a 70/30 split. Thirty percent of the training images were moved to a separate validation folder. The flipping and shifting transformations were only applied to the train set. The final non-augmented dataset consisted of:

Table 1: Number of images in non-augmented dataset

Train Set	**# Images**
Non-impaired (cognitively normal)	1792
Very mild impairment	1254
Mild impairment	502
Moderate impairment	36
Validation Set	
Non-impaired (cognitively normal)	768
Very mild impairment	538
Mild impairment	215
Moderate impairment	16
Test Set	
Non-impaired (cognitively normal)	640
Very mild impairment	448
Mild impairment	179
Moderate impairment	12

Flipping the Data

The original jpg images were flipped using cv2.flip() in three directions: horizontally (using flip code 1), vertically (using flip code 0) and around both axes (using flip code -1). The data was also balanced for all three classes, except for the smallest minority class, by selecting varying numbers of the original images to flip. The smallest class did not contain enough images for it to be balanced.

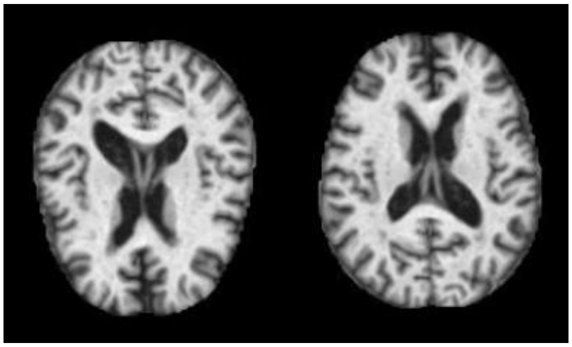


Figure 2: Original image on the left; flipped image on the right

Flipped Train Set	**# Images**
Non-impaired (cognitively normal)	2008
Very mild impairment	2008
Mild impairment	2008
Moderate impairment	144

The validation and test sets for the flipped data remained the same as for the non-augmented data.

Shifting the Data

The original jpg images were also shifted, using cv2.warpAffine(). By using more shifts on the smallest class, a balanced dataset was created. This seemed like it might be a good transformation to try because MRIs are all taken at slightly different angles, depending on the machine and the person.

Shifted Train Set	**# Images**
Non-impaired (cognitively normal)	2008
Very mild impairment	2008
Mild impairment	2008
Moderate impairment	2016

After augmenting the data, we had 3 different training sets: one with no augmentation, one with flipped images, and one with shifted images. The validation and test sets remained the same.

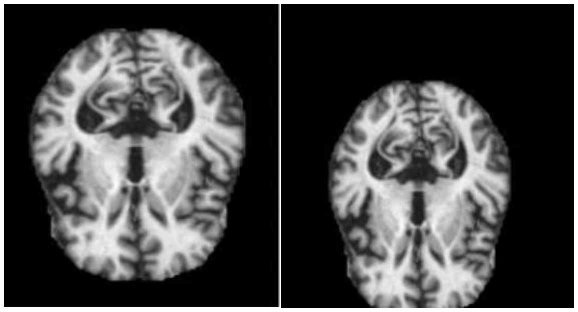


Figure 3: Original image on the left; shifted image on the right

Pre-trained CNN Networks

Three pre-trained CNN networks were selected for this project: ResNet50, VGG16, and Xception.

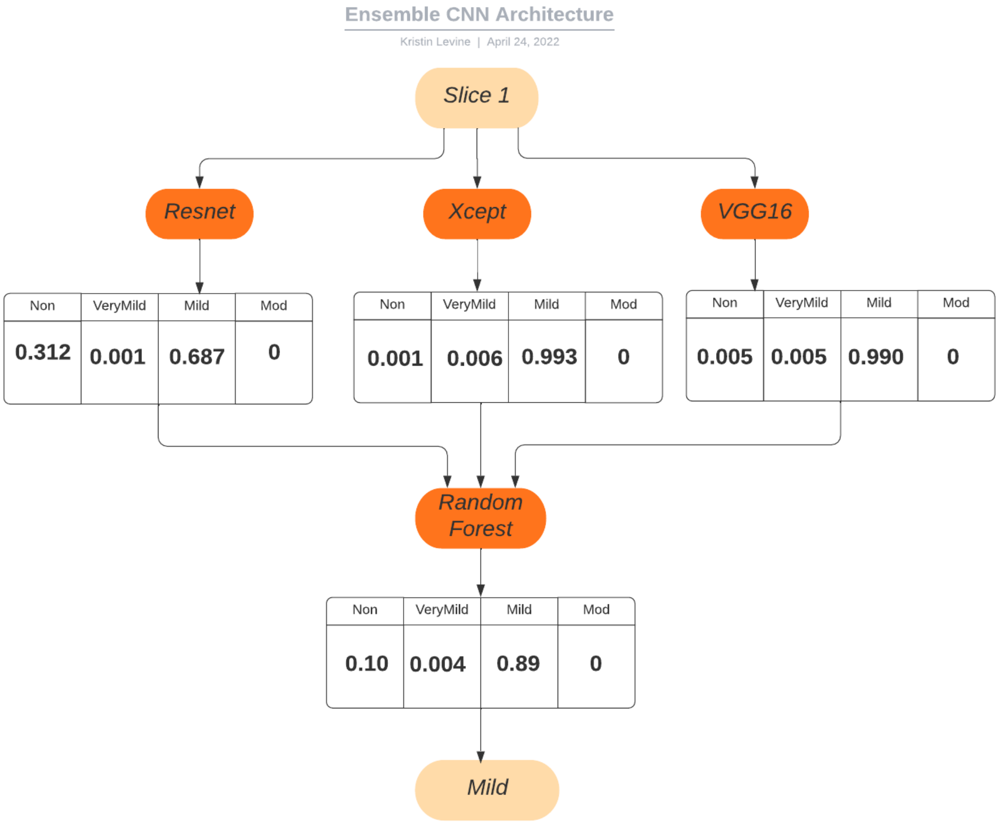
ResNet50 was the first CNN model to introduce the concept of a skip connection: a signal feeding into a layer is also added to the output of another layer that is located higher up in the stack. It won the ImageNet challenge in 2015. Input shape (224, 224, 3)

VGG16 won the ImageNet competition in 2014 and uses smaller filter, but a deeper network. It was named after the Visual Geometry Group from Oxford that developed it. Input shape (224, 224, 3)

Xception uses depthwise separable convolution and improves upon the Inception model. It dates from 2017 and is 71 layers deep. Input shape (299, 299, 3)

Creating the Models

Each pre-trained model was run on each dataset, for a total of nine models. Finally, the best three models were selected and ensembled together using a standard random forest algorithm.



Non	VeryMild	Mild	Mod
0.312	0.001	0.687	0

Non	VeryMild	Mild	Mod
0.001	0.006	0.993	0

Non	VeryMild	Mild	Mod
0.005	0.005	0.990	0

Non	VeryMild	Mild	Mod
0.10	0.004	0.89	0

Figure 4: For each slice, each model generates a probability for each class. These probabilities are combined in a data frame; a random forest is then run to come up with a final prediction.

4 Results

4.1 Individual Model Results

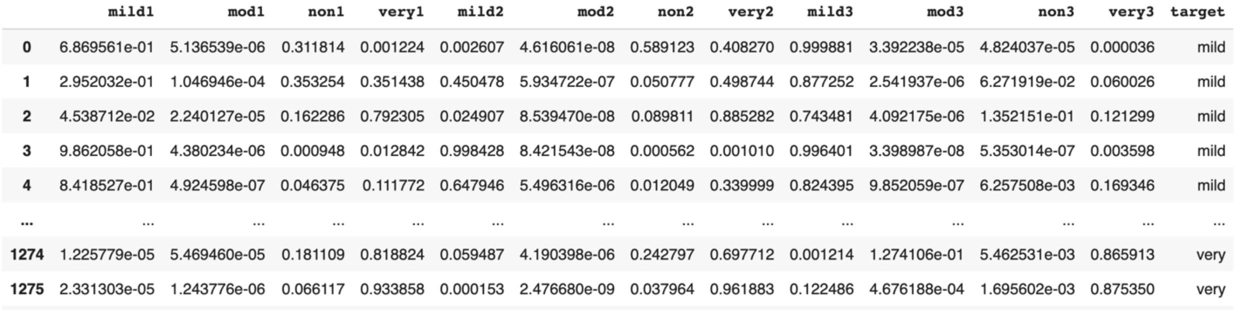
All the models were trained on Google Colab Pro, using a batch size of 32, with dropout of 0.1. The frozen layers were trained for 10 epochs with a learning rate of 0.001, using both early stopping and reduce LR on plateau callbacks. Each model was then unfrozen and trained it for an additional 10 epochs, starting with a learning rate of 0.0001.

Table 2: Results from nine individual models

Model	Augmentation	Train Acc	Val Acc	Test Acc
ResNet	None	1.00	0.9740	0.7303
VGG16	None	0.8597	0.8699	0.6529
Xception	None	0.9980	0.9453	0.6701
ResNet	Flip	0.9992	0.9818	0.7224
VGG16	Flip	0.9162	0.8530	0.6482
Xception	Flip	0.9990	0.9655	0.7177
ResNet	Shift	0.9981	0.9876	0.7482
VGG16	Shift	0.9778	0.8783	0.6904
Xception	Shift	0.9994	0.9681	0.7209

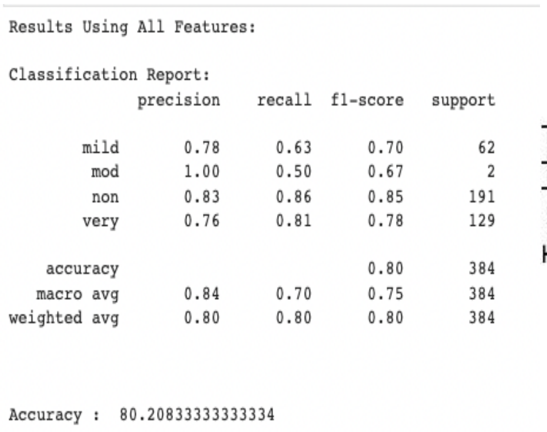
4.2 Ensembled Results

To ensemble the models, the probability estimates from each individual CNN model with the best results (the shifted models for all three of the pre-trained models) were combined in a data frame.



	mild1	mod1	non1	very1	mild2	mod2	non2	very2	mild3	mod3	non3	very3	target
0	6.869561e-01	5.136539e-06	0.311814	0.001224	0.002607	4.616061e-08	0.589123	0.408270	0.999881	3.392238e-05	4.824037e-05	0.000036	mild
1	2.952032e-01	1.046946e-04	0.353254	0.351438	0.450478	5.934722e-07	0.050777	0.498744	0.877252	2.541937e-06	6.271919e-02	0.060026	mild
2	4.538712e-02	2.240127e-05	0.162286	0.792305	0.024907	8.539470e-08	0.089811	0.885282	0.743481	4.092175e-06	1.352151e-01	0.121299	mild
3	9.862058e-01	4.380234e-06	0.000948	0.012842	0.998428	8.421543e-08	0.000562	0.001010	0.996401	3.398987e-08	5.353014e-07	0.003598	mild
4	8.418527e-01	4.924598e-07	0.046375	0.111772	0.647946	5.496316e-06	0.012049	0.339999	0.824395	9.852059e-07	6.257508e-03	0.169346	mild
...	...	...	...	...	...	...	...	...	...	...	...	...	...
1274	1.225779e-05	5.469460e-05	0.181109	0.818824	0.059487	4.190398e-06	0.242797	0.697712	0.001214	1.274106e-01	5.462531e-03	0.865913	very
1275	2.331303e-05	1.243776e-06	0.066117	0.933858	0.000153	2.476680e-09	0.037964	0.961883	0.122486	4.676188e-04	1.695602e-03	0.875350	very

This data was trained again using a random forest, with a 70/30 train test split, yielding a final accuracy of 80.21%.



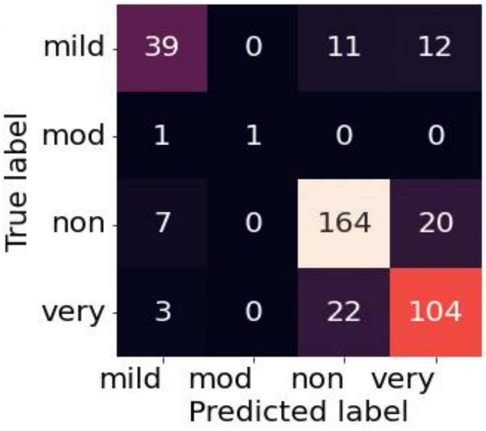
```
Results Using All Features:

Classification Report:
              precision    recall  f1-score   support

        mild       0.78      0.63      0.70        62
         mod       1.00      0.50      0.67         2
         non       0.83      0.86      0.85       191
        very       0.76      0.81      0.78       129

    accuracy                           0.80       384
   macro avg       0.84      0.70      0.75       384
weighted avg       0.80      0.80      0.80       384


Accuracy :  80.20833333333334
```



5 Discussion

Each of the pretrained models did an excellent job on the training and validations sets; all of them struggled to deal with the test set.

It is interesting to note that flipping the images did *not* help the models; in fact, for two of the models, ResNet and VGG16, it made them worse. This is probably due to the fact that all these images were already standardized to face in one direction. It is also unclear if you can flip images of the brain – different hemispheres play different roles. It is possible that flipping the images actually confuses the models, because it now sees changes in a different part of the brain.

Shifting the images, however, was helpful. It improved the test accuracy for all three of the models – improving the ResNet model by about 2%, the VGG16 model by about 4%, and the Xception model by 5%.

Table 3: Improvements through shifting images

Model	**Test Acc (No Aug)**	**Test Acc (Shifted)**	**Improvement**
ResNet	0.7303	0.7482	0.0179
VGG16	0.6529	0.6904	0.0375
Xception	0.6701	0.7209	0.0508

The models also varied in how well they classified each category in the test set.

Table 4: Precision, recall, and F-1 score for the test set

Class	**Precision**			**Recall**			**F1-score**		
	R	V	X	R	V	X	R	V	X
Mild	0.82	0.58	0.80	0.37	0.37	0.50	0.51	0.45	0.62
Mod	1.00	1.00	0.56	1.00	0.17	0.83	1.00	0.29	0.67
Non	0.77	0.80	0.84	0.86	0.78	0.69	0.81	0.79	0.76
Very	0.70	0.58	0.61	0.73	0.70	0.84	0.72	0.64	0.71

For the “mild” class:

- ResNet had the best precision (0.82)
- Xception had the best recall (0.50) and f1 score (0.62)

For the “mod” class:

- ResNet had perfect precision, recall, and f1-score for this class (note: the test set was very small – 12 images)

For the “non” class:

- Xception had the best precision (0.84)
- ResNet had the best recall (0.86) and f1 score (0.81)

For the “very” class:

- ResNet had the best precision (0.70) and f1 score (0.72)
- Xception had the best recall (0.84)

Ensembling the models results in a dramatic improvement in the accuracy of the test set data, ranging from 5-11%.

Table 5: Improvements from ensembling

Model	**Test Acc (Shifted)**	**Test Acc (Ensembled)**	**Improvement**
ResNet	0.7482	0.8021	0.0539
VGG16	0.6904	0.8021	0.1117
Xception	0.7209	0.8021	0.0812

After noticing that the VGG16 model did not perform the best in any of the categories, the ensembled model was rerun using only the two highest models: Shifted ResNet (0.7482) and Shifted Xception (0.7209). It is interesting to note that this two-model ensemble did not perform as well, achieving a test accuracy of only 0.7839. So while the VGG16 model may not be the best in any one category, adding it to the ensembled model improved the combined model accuracy by almost 2%.

6 Conclusion

When comparing non-augmented, flipped, and shifted datasets, shifting the images to create more data improved test set accuracy – even as it had no effect on the accuracy of train and validation sets. In this study, it improved test performance by 2-5%, depending on the model used.

In a field such as Alzheimer's research, shifting images to create more data is a way to improve test-set performance and perhaps increase the transferability of the model to other datasets.

Even when various models provide good results individually on train and validation sets, performance can be increased on the test set by ensembling them together to take advantage of their individual strengths and minimize their weaknesses. In this study, ensembling three models improved test performance by 5-11%. This benefit was seen even when all of the models had excellent accuracy on the training and validation sets; it even improved our final score when one of the models was a bit weaker than the other two.

Ensembling is another method that could be put to good use when dealing with small, limited datasets in neurodegenerative disease research.

Further research directions might include more detained investigations into the optional amount to shift images; exploring more pre-trained models; and/or investigating other image transformations such as brightness or normalization techniques.

Data and Code Availability

The datasets analyzed during the current study are available here. Code notebooks are available here.

Funding

No funding was received to assist with the preparation of this manuscript.

7 Bibliography

1. Sun, H., Wang, A., Wang, W. & Liu, C. An Improved Deep Residual Network Prediction Model for the Early Diagnosis of Alzheimer's Disease. *Sensors* vol. 21 4182 (2021).

2. Syaifullah, A. H. *et al.* Machine Learning for Diagnosis of AD and Prediction of MCI Progression From Brain MRI Using Brain Anatomical Analysis Using Diffeomorphic Deformation. *Front. Neurol.* **11**, 576029 (2020).

3. Yala, A. *et al.* Multi-Institutional Validation of a Mammography-Based Breast Cancer Risk

Model. *J. Clin. Orthod.* JCO.21.01337 (2021).

4. Alzubaidi, L. *et al.* Novel Transfer Learning Approach for Medical Imaging with Limited Labeled Data. *Cancers* **13**, (2021).

5. Helaly, H. A., Badawy, M. & Haikal, A. Y. Deep Learning Approach for Early Detection of Alzheimer's Disease. *Cognit. Comput.* 1–17 (2021).

6. Li, Y. *et al.* Transfer learning-trained convolutional neural networks identify novel MRI biomarkers of Alzheimer's disease progression. *Alzheimers. Dement.* **13**, e12140 (2021).

7. Tanveer, M. *et al.* Classification of Alzheimer's disease using ensemble of deep neural networks trained through transfer learning. *IEEE J Biomed Health Inform* **PP**, (2021).