Lung Image Classifier: Classification of Lung X-Rays as COVID-19 or Pneumonia via Random Forest, KNN and CNN Algorithms

CMPT 340 Biomedical Computing - Ghassan Hamarneh

Sabrina Dalen (200065923), Jin Lu (301256252), Hui Wu (301329435),
Caijie Zhao (301354256), and Huiyi Zou (301355563)

Department of Computer Science
Simon Fraser University, Burnaby BC V5A 1S6, Canada

Abstract. As X-Ray classification plays an important role in medical diagnosis, our project aimed to classify lung X-ray images as healthy or diseased with COVID-19 or viral pneumonia by training networks using convolutional neural network (CNN), random forest (RF), and k-nearest neighbors (KNN) algorithms. Image resolution size and the number of images used in training greatly affected the accuracy of both classifiers. The RF and KNN classifier required a balanced dataset and therefore used a small subset of the images, which further reduced its accuracy compared to the CNN classifier, which was able to use all the training data. Sensitivity of the CNN classifier was found by calculating the true positive ratio (TPR) to classify images as COVID-19, viral pneumonia, or healthy at 100%, 94%, and 83% respectively. The true negative rate (TNR) for classifying healthy lung images was 100%. Due to the inadequate number of images, the RF and KNN algorithms misclassified some COVID-19 images as healthy, but performed reasonably well on viral pneumonia and healthy lung classification. The TPR for COVID-19 was 29% for RF and 17% for KNN. The TPR for viral pneumonia was 78% for RF and 86% for KNN. Healthy lungs were classified perfectly by RF and had a TPR of 90% by KNN. This report also discusses gray-level co-occurrence matrix (GLCM) texture features extracted from the images and how to combine the above algorithms with the extracted features to group the images based on their classification. Overall, CNN performed better on classification, as RF and KNN were limited by requiring a balanced dataset.

Keywords: COVID-19, X-Rays, Random Forest, KNN, CNN, Pneumonia, Classifier, Machine Learning, Deep Learning

1 Introduction

As COVID-19 spreads worldwide and is declared a pandemic, virus identification from X-rays may assist medical research and diagnosis. Therefore, in this project, we implement CNN, RF, and KNN algorithms to train network models to classify lung X-ray images as COVID-19, viral pneumonia, or healthy. This report first describes the datasets used in the Material section, then provides a detailed description of the algorithms and formulas under Methods. Results of the image classification from all the algorithms and their comparison is displayed via tables and figures in the Results section. A summary of the development process, including obstacles faced are presented under Accomplishments. Contributions by each team member are then outlined. Finally, the Conclusions and Discussion section summarizes the results along with an analysis of the project.

2 Material

Datasets are X-ray images of lungs acquired from GitHub and Kaggle, providing training data of 107 images from COVID-19 patients, 1,407 images from viral pneumonia patients, and 1,400 images from healthy people. All training images are resized to 227 x 227 pixels.

In the deep learning process, the CNN classifier makes use of all training data. However, as explained in the Discussion, imbalanced datasets affect the accuracy of the RF and KNN algorithms. Therefore, the RF and KNN classifier uses only a subset of 107 images each from the COVID-19, viral pneumonia, and healthy classes.

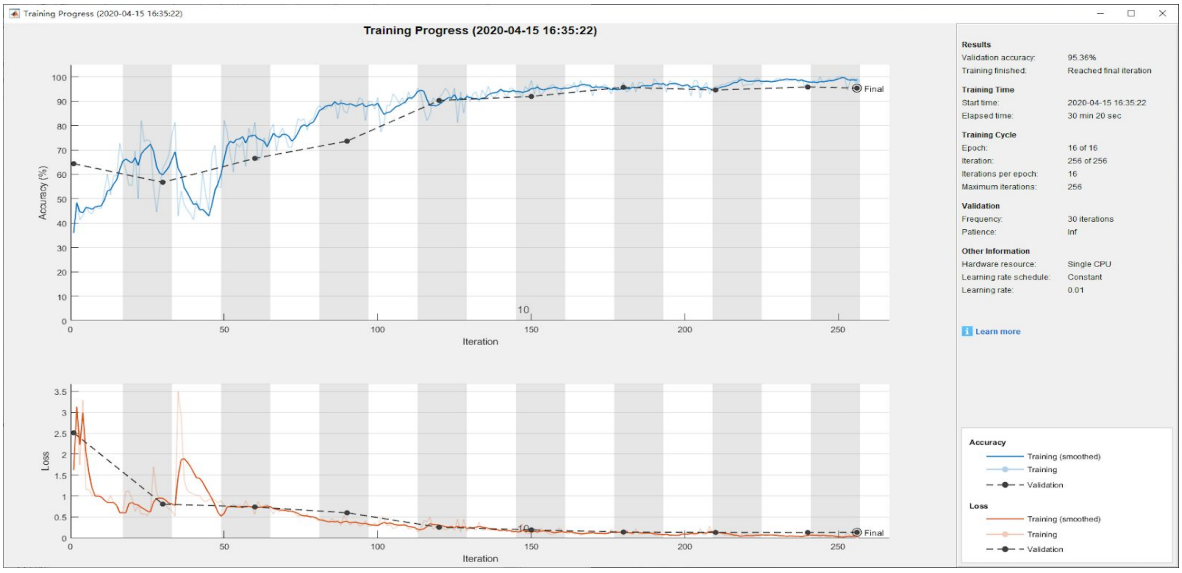
The COVID-19 images are RGB; the others are grayscale. Bone absorbs the most radiation, so for X-ray images of healthy people, the rib cage appears white and the lung can be viewed clearly. However, ground glass opacity (GGO) obscures the lungs in diseased cases.

3 Methods

3.1 Deep Learning Classification by CNN

For the training process, sets of images labeled by their sub-folder filename will be passed into the network. After the training procedure, the network can be loaded and used to classify X-Ray images.

Convolutional Neural Network Construction. We first chose AlexNet, but the validation accuracy is lower than 50%. To get a more suitable network, we tuned some variables related to network property (such as stride, padding and filter size of network layers) and training process(such as learning rate and number of epochs) and conducted multiple experiments. Finally, we found the AlexNet with the filter size of first convolutional layer changed from 11*11 to 3*3 has a higher accuracy up to 95%. The learning rate is set to 0.01 and the number of epochs is 16. The image below shows the learning progress:



Classification. An image classified by a convolutional neural network goes through all its layers using trained weights. Confidence levels for the image are then generated, showing how closely the image is expected to match each class. Finally, the class with the highest confidence level is chosen as the predicted class for the image.

3.2 Machine Learning Classification by RF and KNN

For machine learning methods, image features are required to be extracted from three categories of X-ray images before applying classification algorithms. After building up the models, KNN and RF are implemented to do the final classification.

Image Features Extraction. The X-ray images are grayscale. Their image texture gives us their intensities and features. The gray-level co-occurrence matrix (GLCM) is applied to extract the texture features from images and it is defined from the distribution of co-occurrence pixels in a given offset, the distance between the calculated pixel and

its neighbors. Assume an image with p different pixel values and the $p \times p$ co-occurrence matrix C is defined over an $n \times m$ image I and an offset $(\Delta x, \Delta y)$ in Eq. (1) as follows:

$$C_{\Delta x,\Delta y}(i,j) = \sum_{x=1}^{n} \sum_{y=1}^{m} \{1, if\ I(x,y) = i\ and\ I(x+\Delta x, y+\Delta y) = j\ ,\ and\ 0\ otherwise \quad (1)$$

where i and j are the pixel values.

Here, we extract six features: Contrast, Correlation, Energy, Homogeneity, Entropy, and Inverse Differential Moment (IDE) from the gray-level co-occurrence matrix calculated with four angles: 0, 45, 90, and 135 degrees. Therefore, we have got a feature vector with 24 elements for every image.

These features of an image derived from a gray-level co-occurrence matrix are:

$$Constrast = \sum_{i,j=0}^{N-1} P_{ij}(i-j)^2 \quad (2)$$

$$Correlation = \sum_{i,j=0}^{N-1} P_{ij} \frac{(i-\mu)(j-\mu)}{\sigma^2} \quad (3)$$

$$Energy = \sum_{i,j=0}^{N-1} {P_{ij}}^2 \quad (4)$$

$$Homogeneity = \sum_{i,j=0}^{N-1} \frac{P_{ij}}{1+|i-j|} \quad (5)$$

$$Entropy = \sum_{i,j=0}^{N-1} -\ln ln\left(P_{ij}\right) P_{ij} \quad (6)$$

$$IDE = \sum_{i,j=0}^{N-1} \frac{P_{ij}}{1+(i-j)^2} \quad (7)$$

Where P_{ij} is the element i,j of the gray-level co-occurrence matrix, N is the number of gray levels, and the mean μ and the variance of the intensities are calculated as:

$$\mu = \sum_{i,j=0}^{N-1} iP_{ij} \quad (8)$$

$$\sigma^2 = \sum_{i,j=0}^{N-1} P_{ij}(i-\mu)^2 \quad (9)$$

Classification. Random Forest (RF) and K-Nearest Neighbors (KNN) are non-parametric classification methods with high accuracy in machine learning. In this project, we use both methods to classify images by extracted numerical texture features information.

Random Forest. The RF is an ensemble learning method training sample data via building a forest of decision trees for classification. It features bagging to randomly select features in many B trees; at each node of a decision tree, the RF will randomly select one optimal attribute in a subset of all features as a decision of partition.

K-Nearest Neighbors. The KNN is a supervised classifier that needs to extract features from images stored as feature vectors and class labels for each sample. In the classification, an unlabeled vector is gained from X-ray image and an assigned label decided from its K neighbors in a defined distance, voting for the greatest number of labels among these neighbors. As K value selection is important to determine the result of classification. Generally, larger K value reduces noise on classification, but also makes the boundaries of class less distinct. In this project, we optimize

hyperparameters, the value of K and distance type, automatically using fitcknn built-in function in MATLAB. One of the most important advantages of implementing KNN is that the training phrase can be done pretty fast compared to other classification algorithms because KNN keeps all the data points from the training samples instead of doing any generalization.

4 Results

4.1 Texture Features Analysis

Image texture provides information in spatial distribution showing intensity levels. The contrast feature represents the amount of local gray level variation that when the variance of the pixel distribution is high, the texture is deep and images have clear views. The correlation feature is affected by the difference between the values in GLCM and its neighbors, the smaller differences cause the higher correlation. Similarities in GLCM raise the values of energy. The entropy describes the complexity of an image and it will be higher with more complex images. The homogeneity reflects the quality of being uniform throughout in image composition and inverse differential moment (IDE) shows the regularity level of an image.

Fig. 1 shows the result of comparison texture features among COVID-19, viral and normal X-ray images. As shown, normal people's X-ray images have clear view and more complex pixel information since generally they have higher value in contrast and entropy compared to patients' X-ray images. The pixel distribution of the normal is various, while other types have more consistent pixel values and COVID-19 images have the most similar values in GLCM, which increase values of energy, homogeneity, correlation and IDE. Although X-ray images of COVID-19 and viral pneumonia have similar texture features, low variance value in GLCM, COVID-19 images have more consistent values than viral pneumonia images.

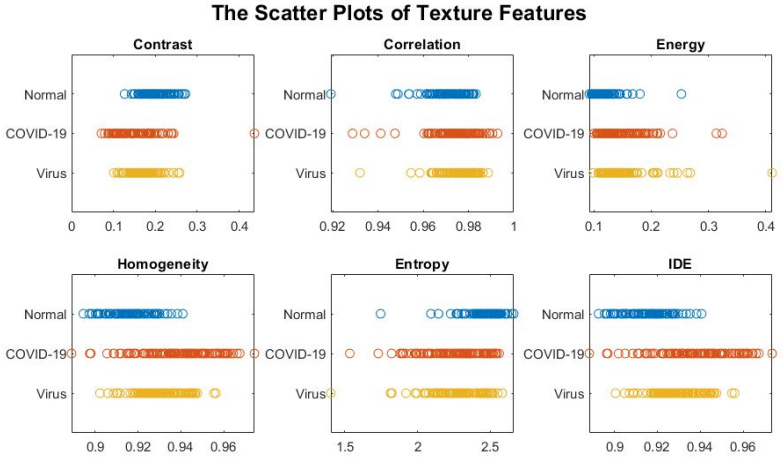


Fig. 1. A comparison of X-ray images texture feature from GLCM at 0-degree direction based on COVID-19, virus pneumonia patients and normal people

4.2 Algorithms Analysis

Fig.2, 3 and 4 show the classification results based on different algorithms. As shown, generally CNN has higher accuracy when classifying COVID-19, viral and normal lungs compared to the machine learning methods,KNN and RF, with 100% TNR, 83.3% and 94.23% TPR for COVID-19, normal and viral pneumonia respectively. However, for KNN and RF, they only have good performance in finding the normal and virus images but the COVID-19 TPR is lower than 30% since there are many images of patients who are infected but are classified as normal lungs.

The classification of normal lungs images always has similar and higher accuracy with more than 90% TNR for all classifiers but the variance of COVID-19 TPR is large from more than 80% (CNN) to less than 30% (KNN, RF).

Labels	ClassifiedNormal	ClassifiedCovid	ClassifiedVIRUS_PNEUMONIA	ActualTotal
NORMAL	47	0	3.0000	50
COVID-19	0	5.0000	0	5
VIRUS_PNEUMONIA	0	1.0000	49.0000	50
SUM	47	6.0000	52.0000	105
TrueRate	1	0.8333	0.9423	NaN

Fig. 2. The confusion matrix of CNN classification

	(RF)ClassifiedNormal	ClassifiedCovid	ClassifiedVirus	ActualTotal
NORMAL	37	2.0000	11.0000	50
COVID-19	0	5.0000	0	5
VIRUS	0	10.0000	40.0000	50
SUM	37	17.0000	51.0000	105
TrueRate	1	0.2941	0.7843	NaN

Fig. 3. The confusion matrix of RF classification

	(KNN)ClassifiedNormal	ClassifiedCovid	ClassifiedVirus	ActualTotal
NORMAL	46.0000	1.0000	3.0000	50
COVID-19	0	3.0000	2.0000	5
VIRUS	5.0000	13.0000	32.0000	50
SUM	51.0000	17.0000	37.0000	105
TrueRate	0.9020	0.1765	0.8649	NaN

Fig. 4. The confusion matrix of KNN classification

5 Accomplishments

In this project, we learned how to extract features from images, apply appropriate algorithms to build desired training models, and test the trained networks against a sample dataset. The importance of standardizing image input size was realized, as well as the need for a balanced dataset for the RF and KNN algorithms. Both of these requirements affected the accuracy of image classification. An obstacle faced was the trade-off between supplying more training data to build a more reliable model, versus the additional time required to read images. To overcome this, image sizes were scaled down. Accuracy for both the CNN and RF and KNN classifiers was shown by creating confusion matrices, along with a scatter plot of selected texture features to compare the classes.

6 Contributions

Our team collaborated throughout the project, from research and software development to testing and documentation. Carina first proposed using the CNN algorithm to classify X-ray images. Jin advised on how to use image processing to prepare the images for classification, which led the team to discover different machine learning approaches. Jin pre-processed the images and built the CNN classifier. Natalie and Carina built the RF and KNN classifiers. Sabrina refactored the code for consistency between files, replaced a repeated code block with a function, and added comments. The team added their results and analysis to the report, along with relative formulas and charts. Kate completed writing the major areas of the report, and Sabrina proofread and edited the report. Carina and Kate produced the demo video.

7 Conclusions and Discussion

7.1 Discussion

Accuracy of both trained networks was strongly influenced by the size of the dataset, in both the resolution size and the total number of images in each class. Training the networks with a smaller dataset provided faster, yet more inaccurate results. The CNN classifier was able to accept an imbalanced dataset without affecting its accuracy, so its model was trained using all image files from the training data. However, the RF and KNN algorithms were found to have accuracy highly dependent on a balanced dataset, so their model used a subset of the training data. The RF and KNN classifier accuracy was therefore limited by the smaller number of images used for training.

Image resolution sizes needed to be standardized prior to processing by the algorithms. The original dataset had unequally sized images with varying pixel counts, which would have caused a lack of consistency in feature extraction. After loading the dataset, the next step was pre-processing the resolution sizes of all training and test images, by either scaling-up or scaling-down, to a consistent 227 x 227 pixel size. Although the CNN classifier could accept an imbalanced dataset with a larger number of images, the training time was significantly affected by resolution size. Our team found the CNN classifier required 10 to 20 minutes to train with less than 3000 images. Doubling the resolution from 227 x 227 to 512 x 512 pixels resulted in a processing time of more than 30 minutes for the first epoch only, so the idea to increase the resolution size was discarded.

Both classifiers could readily scale to accept more training images, with less time constraints than increasing resolution size, if only more training data could be found. Therefore, several original datasets were combined to construct a larger training dataset consisting of 107 COVID-19, 1,407 viral pneumonia, and 1,400 healthy images of lung X-rays. Although the CNN classifier was able to use all training data, the RF and KNN classifier required a balanced dataset. As random forest uses bagging to randomly select subsets of features from each image, an imbalanced dataset would cause features in a class with more training images to have a higher probability of being chosen, reducing the accuracy of classification. For KNN, an imbalanced dataset increases the probability of K nearest neighbors at any random point queried from a class with more samples. Therefore, a subset of 107 images from each class were chosen to achieve a balanced dataset of maximum possible size. However, as the number of training images was therefore reduced, the accuracy of the RF and KNN classifier was lower than that of the CNN classifier, which was able to use all the training data.

For the RF and KNN classifier, we attempted to extract two global texture features, the Gabor wavelet transform and the gray level co-occurrence matrix (GLCM). After calculating the mean and variance from the Gabor wavelet transform based on three orientations and three scales, nine elements for the mean and nine elements for the variance were extracted, for a total of 18 features. We tried to build the RF and KNN classifier using 42 features, 24 from GLCM and 18 from the Gabor wavelet transform. However, the accuracy of the classification did not significantly improve to make the increased processing time of over 10 minutes worthwhile. Therefore, only the 24 features from GLCM were applied.

7.2 Conclusion

In this project, CNN, RF, and KNN algorithms were applied to lung X-rays, creating trained network models to classify images as COVID-19, viral pneumonia, or normal (healthy). Results of the image classifications showed CNN provided higher accuracy than RF and KNN. The accuracy of the CNN classifier improved with more data, so its model was trained with all available training data. However, the RF and KNN classifier required a balanced dataset, so its accuracy was limited by an inadequate supply of training data (107 images in each class).

8 References

1. ieee8023, "ieee8023/covid-chestxray-dataset," GitHub, 30-Mar-2020. [Online]. Available: https://github.com/ieee8023/covid-chestxray-dataset/tree/master/images. [Accessed: 02-Apr-2020].
2. P. Mooney, "Chest X-Ray Images (Pneumonia)," Kaggle, 24-Mar-2018. [Online]. Available: https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia. [Accessed: 02-Apr-2020].
3. E. Allibhai, "Building a Convolutional Neural Network (CNN) in Keras," Medium, 15-Nov-2018. [Online]. Available: https://towardsdatascience.com/building-a-convolutional-neural-network-cnn-in-keras-329fbbadc5f5. [Accessed: 02-Apr-2020].
4. W. Koehrsen, "An Implementation and Explanation of the Random Forest in Python," Medium, 31-Aug-2018. [Online]. Available: https://towardsdatascience.com/an-implementation-and-explanation-of-the-random-forest-in-python-77bf308a9b76. [Accessed: 02-Apr-2020].
5. O. Harrison, "Machine Learning Basics with the K-Nearest Neighbors Algorithm," Medium, 14-Jul-2019. [Online]. Available: https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761. [Accessed: 02-Apr-2020].
6. Bachir, "COVID-19 chest xray," *Kaggle*, 17-Apr-2020. [Online]. Available: https://www.kaggle.com/bachrr/covid-chest-xray. [Accessed: 21-Apr-2020].
7. N. Sajid, "COVID-19 Patients Lungs X Ray Images 10000," *Kaggle*, 23-Mar-2020. [Online]. Available: https://www.kaggle.com/nabeelsajid917/covid-19-x-ray-10000-images. [Accessed: 21-Apr-2020].
8. Kermany, Daniel; Zhang, Kang; Goldbaum, Michael (2018), "Labeled Optical Coherence Tomography(OCT) and Chest X-Ray Images for Classification", Mendeley Data, v2. Available: https://data.mendeley.com/datasets/compare/rscbjbr9sj/1/2. [Accessed: 21-Apr-2020].
9. W. Cao, J. Shan, N. Czarnek and L. Li, "Microaneurysm detection in fundus images using small image patches and machine learning methods," *2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, Kansas City, MO, 2017, pp. 325-331 [Accessed: 02-Apr-2020].
10. Nedjar, Imane & EL HABIB DAHO, Mostafa & Settouti, Nesma & Saïd, Mahmoudi & Chikh, Mohammed. (2015). Random forest based classification of medical x-ray images using a genetic algorithm for feature selection. Journal of Mechanics in Medicine and Biology. 15. 1540025. 10.1142/S0219519415400254 [Accessed: 02-Apr-2020].
11. O. Harrison, "Machine Learning Basics with the K-Nearest Neighbors Algorithm," *Medium*, 14-Jul-2019. [Online]. Available: https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761. [Accessed: 21-Apr-2020].
12. "K-nearest neighbors algorithm," *Wikipedia*, 14-Apr-2020. [Online]. Available: https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm. [Accessed: 21-Apr-2020].
13. "Co-occurrence matrix," *Wikipedia*, 16-Apr-2020. [Online]. Available: https://en.wikipedia.org/wiki/Co-occurrence_matrix. [Accessed: 21-Apr-2020].
14. *GLCM Texture Feature*. [Online]. Available: https://support.echoview.com/WebHelp/Windows_and_Dialog_Boxes/Dialog_Boxes/Variable_properties_dialog_box/Operator_pages/GLCM_Texture_Features.htm. [Accessed: 21-Apr-2020].

9 Acknowledgements

We would like to thank our TA, Mengliu Zhao, for suggesting the idea of image classification and posting several projects to use as a reference. We realized that image classification is a hot topic in computer vision and pattern recognition, laying the foundation of our project. Mengliu also provided helpful advice on CNN and recommended RF, KNN, and SVM. We also thank Joseph Paul Cohen, Paul Morrison, and Lan Dao, for their collection of COVID-19 data on Github, which served as the main dataset for training the CNN, RF, and KNN classifiers.

Similarly, we thank Kermany Daniel, Zhang Kang, Goldbaum Michael for their dataset of normal and viral pneumonia X-ray images. Finally, we appreciate the article from Imane Nedijar, Nesma Settouti, Mahmoudi Said and Mostafa El Habib Daho, which inspired our choice of which features to extract from the X-ray images.