Chest X-Ray Analysis of Tuberculosis by Convolutional Neural Networks with Affine Transforms

Tawansongsang Karnkawinpong
Department of Computer Engineering
Chulalongkorn University
Bangkok 10330, Thailand
Tawansongsang.K@student.chula.ac.th

Yachai Limpiyakorn
Department of Computer Engineering
Chulalongkorn University
Bangkok 10330, Thailand
Yachai.L@chula.ac.th

ABSTRACT
Applying deep learning techniques for classification of medical images has seen considerable growth in recent years. Among several, Convolutional Neural Net-works (CNNs) are a class of powerful models well known for image classification and segmentation. This research introduces the concept of computer-aided diagnosis that could help in early diagnosis of Tuberculosis infection. The three CNN architectures: AlexNet, VGG-16 and CapsNet, were customized to classify tuberculosis lesions in CXR images. Data augmentation with rotating was used to mimic the real world as CXR images may not be precisely vertical. The performance of the three classifiers was evaluated with the measures: accuracy, sensitivity and specificity. The result showed that CapsNet outperformed the other models when predicting affined images.

CCS Concepts
• Computing methodologies → Neural networks

Keywords
Computer-Aided Diagnosis; Convolutional Neural Network; Affine Transforms; Tuberculosis.

1. INTRODUCTION
The Tuberculosis (TB) is a contagious disease and endemic in under-resourced regions. In 2017, the Ministry of Public Health, Thailand, reported that a population of more than 40,000 infected with TB disease and 20 million infected without the disease. TB is mostly found in the lung region that is termed as Pulmonary Tuberculosis (PTB). PTB commonly manifests upper lung and it is hard to confirm its presence as it ap-pears different pathological patterns depending on various factors. PTB is a curable disease. However, due to the massive workload of physicians, it inevitably results in long and deleterious time-to-treatment periods for patients. A computer-aided triage system would mitigate these issues by providing the initial CXR interpretation to out-put a classification of normal/ infected per image.

The concept of computer-aided diagnosis (CAD) for chest x-rays has been around for the past fifty years. Traditionally, CAD systems use machine learning technique to perform tasks by analyzing relationships of existing data. In this study, we use CNNs, a type of deep learning that employs multiple hidden layers and has been remarkably successful for image classification, to detect whether having a TB lesion or not on the CXR images. The performance of classifying affined images is also investigated. The proposed approach is promising as one of the advantages of deep learning is its ability to excel with high-dimensional datasets, such as images. Convolutional Neural Networks are a class of powerful generative models with a variety of architectures. CNNs are regarded the best of image classification at present. They overcome old method performance of image classification without segmentation and manual feature selection. Features are automatically extracted via their own archi-tectures. In literature, researchers have proposed various CNN classifiers such as AlexNet in 2012 [1], VGG in 2014 [2], Inception V3 in 2015 [3], ResNet in 2015 [4], and Xception in 2016 [5]. To overcome the performance of image classification, the latest CNN architecture, capsules or capsules network (CapsNet), was proposed by Sabour et al. [6] in 2017. The classifier achieved state of the art performance on MNIST and it is claimed as the best-performing classifier for predicting data with affine transforms that are excluded from training datasets.

CSAI '18, December 8–10, 2018, Shenzhen, China

ACM ISBN 978-1-4503-6606-9/18/12…$15.00
DOI: https://doi.org/10.1145/3297156.3297251

2. RELATED WORK
In the world of CXR classification of TB, the first CNN model for TB detection is AlexNet proposed by Hwang et al. in 2016 [7]. In the model, pre-trained was used as the upper layer except in some lower layers where fine tuning was performed. Montgomery and Shenzhen datasets were used as testing data, whereas the model was trained and validated on customized dataset. The AUC performances on test datasets were 0.93 and 0.88 respectively. Cao et al. [8] applied GoogLeNet [9] model for tuberculosis diagnostics on mobile. They used pre-trained model and achieved the accuracy of 89.6%. Hooda et al. [10] used customized CNNs based on LeNet and AlexNet architectures for the detection of TB. They used CXR images from Mont-gomery and Shenzhen datasets. The customized CNNs had 19 layers and consisted of 7 Conv layers, 7 ReLu-layers and 3 Fully-connected layers. Dropout layers were used for preventing the overfitting layer. They compared performance of three different optimizers and found that Adam optimizer overcome the others achieving the accuracy of 94.73% and validation accuracy of 82.09%. Liu et al. [11] proposed TB detection model based on AlexNet and GoogLeNet architectures with different model parameters. The model was trained with a very unbalanced dataset. Another study by Rajaraman et al. [12] on comparison of deep learning models with CXRs aimed to compare the performance of their customized CNN with five pre-trained CNNs (AlexNet, VGG-16, VGG-19, Xception and ResNet) using Montgomery, Shenzhen, Kenya and India datasets, maintained by National Library of Medicine (NLM) and National Institutes of Health (NIH). The customized CNN model for the binary classifying as normal or TB, consisted of 3 Conv layers, 3 max pooling layers, 1 dropout layer and 2 Fully-

connected layers. They used softmax classifier with SGD momentum and L2-regularization optimizer. Each Conv layer was followed by batch normalization [13] and ReLu layer. The performance of customized model achieved accuracy of 0.824 and AUC of 0.900. The comparison results reported that without optimal features, AlexNet achieved the highest accuracy and AUC on all the four datasets. The datasets were randomly split into 80% for training and 20% for testing. However, with optimal features, VGG-16 outperformed the other models. Stirenko et al. [14] proposed data augmentation to increase dataset and prevent overfitting of deep CNN models. The study focused on comparisons of image rotation by 90n degrees, where n = 1, 2, 3 and rotation by 5 degrees. Rahul and Ajay [15] used data augmentation techniques to enhance the number of training images by six different techniques. Three augmented images were created by rotating the images by 90, 180 and 270 degrees. Two augmented images were created with horizontal and vertical flips. The last augmented image was obtained by performing histogram equalization. The training dataset was thus enlarged 7 times, resulting in improved performance.

3. METHOD

3.1 Dataset and Pre-processing

The dataset contains only chest radiographic images of lungs as normal or having TB lesion acquired from National Library of Medicine (NLM) and the Ministry of Public Health, Thailand. All the CXR images contained in a private Thai dataset are deidentified and each of them is confirmed by a radiologist to ensure all images are labelled accurately. The dataset contains totally 3310 images: normal lung 1803 images (example as shown in Figure 1) and having TB lesion 1507 images (example as shown in Figure 2). The class label of images is assigned to value "1" representing TB positive, or "0" denoting TB negative or normal. Table 1 describes the characteristics of each dataset. Observing that the image resolution of each dataset differs. All images were, therefore, down-sampled size to 32x32 pixel resolution to suit the input requirements for CNN model.

Figure 1. CXR images of Normal.

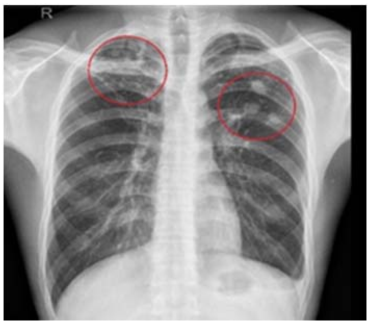

Figure 2. CXR images of Tuberculosis.

3.2 Data Augmentation

The initial dataset of 3310 images was divided into training-validation set of 2648 images (80%), and test set of 662 images (20%). The adaptation of data augmentation techniques [16] is applied to enhance the number of training images. Each of 2648 images was augmented by rotating between -10 to 10 degrees randomly. The size of training dataset was thus boosted from 2648 to 5296 images (2648 affined + 2648 original). Two test datasets were created. The first set contained the initial tests of 662 images plus 662 images with -10 to 10 degrees rotating. The second set contained the initial tests of 662 images plus 662 images with -30 to 30 degrees rotating. Examples of affined images with rotating are shown in Figure 3. Once having augment-ed data with rotating, the number of training, validation, and test is 4634, 662, and 1324, respectively.

Figure 3. Affine transformation of CXR images.

Table 1. Dataset description

Database	# of TB	# of Normal	Resolution
Shenzhen	336	326	948-3001x1130-3007
Montgomery	58	80	4020-4892x4020-4892
Thai	1113	1397	200-3072x343-3072

3.3 CNN Models

In this work, we use a Windows system with Intel(R) Core (TM) i7-6700k CPU @ 4.00GHz, 8 GB RAM, a Nvidia GeForce GTX 1080 Ti6 GB graphical processing unit (GPU), Keras with Tensorflow backend, and CUDA 8.0 for GPU acceleration. The ratio of the size of the training-validation dataset and the test dataset is 80-20%.

In [12], the study focused on comparison of the existing CNN architectures which have been popularly used in current image classification problems. It found that AlexNet outperformed the other CNNs with higher accuracy and AUC. While VGG-16 with optimal features techniques outperformed the other CNNs with higher accuracy. AlexNet and VGG-16 are thus selected for study in this work. The models are constructed using the learning rate=0.1, the value of learning rate decay=0.0001, and SGD is used as the optimizer algorithm.

The CapsNet architecture presented in [6] is applied in this work. The structure is shallow with 2 convolution layers (traditional convolution layer and primary capsules layer) and 1 fully-connected layer (DigitCaps layer) as illustrated in Figure 4. Conv1 and Conv2 are the layer used for extracting local features that are then used as inputs to the primary capsules (PrimaryCaps). The second layer or PrimaryCaps is a convolution capsule layer with 4 channels. Each primary capsule consists of 8 convolution unit (8D capsules) with a 3x3 kernel and a stride of 2.

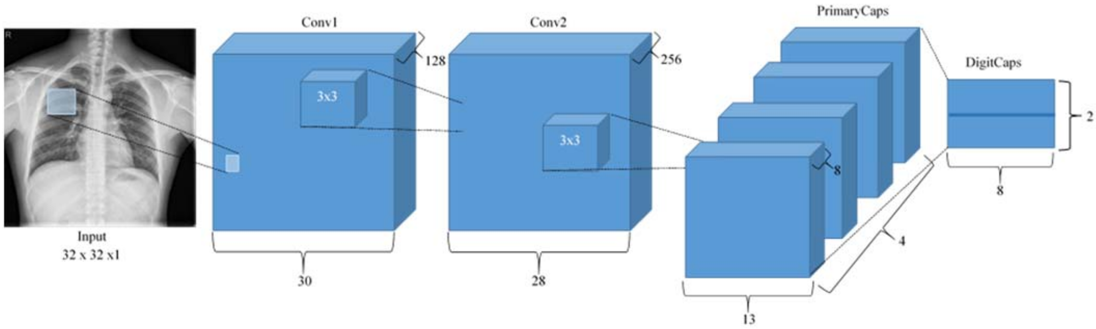

Figure 4. Customized architecture of capsule model.

4. EVALUATION

4.1 Metrics

The performance of the classifiers is evaluated by three measures: 1) accuracy, 2) sensitivity (also known as recall or the true positives rate) and 3) specificity (also known as false positive rate). Accuracy, sensitivity and specificity are defined as in (1), (2) and (3), respectively:

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+TN+FN} \tag{1}$$

$$\text{Sensitivity} = \frac{TP}{TP+FN} \tag{2}$$

$$\text{Specificity} = \frac{TN}{TN+FP} \tag{3}$$

where TP denotes the number of true positives,
FP denotes the number of false positives,
TN denotes the number of true negatives, and
FN denotes the number of false negatives.

4.2 Experimental Results

For AlexNet and VGG-16, the training images were divided into 128 batches and the model was trained for 200 epochs. The weights in each layer were randomly initialized. Biases were also random. After each epoch, weights were updated by SGD optimizer. The learning rate is 0.1 and learning rate decay is 0.0001. The number of training epochs of AlexNet and VGG-16 gauged by the validation set are illustrated in Figure 5 and Figure 6, respectively.

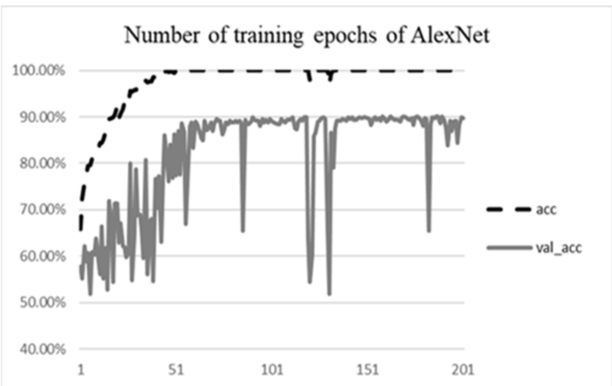

Figure 5. Visualization of training accuracy of AlexNet gauged with validation dataset.

Figure 6. Visualization of training accuracy of VGG-16 gauged with validation dataset.

For CapsNet, the training images were divided into 64 batches and the model was trained for 52 epochs. The weights in each layer were randomly initialized. Biases were also random. After each epoch, weights were updated by margin loss function [6] and Adam optimizer [16]. The learning rate is 0.001. The number of training epochs of CapsNet gauged by the validation set is illustrated in Figure 7.

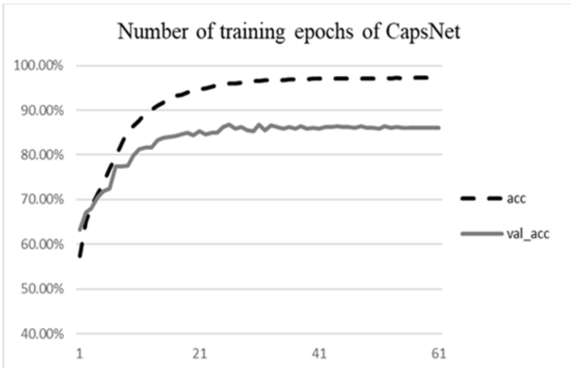

Figure 7. Visualization of training accuracy of CapsNet gauged with validation dataset

Summary of all the measures of the three classifiers is detailed in Table 2. Observing that the performance of AlexNet and VGG-16 is not quite different when classifying with test set containing -10 to 10 rotating instances. While CapsNet is defeated in case of testing with the first test set, it outperforms the others when classifying with the second test set containing -30 to 30 rotating images. The performance of AlexNet and VGG-16 significantly decrease compared to their performance when testing with -10 to 10 degrees rotation. This would result from the fact that most original CNNs use pooling layers to reduce parameter computing. The pooling layers, in turn, destroy the information of location and feature relationships in the images. On the contrary, CapsNets is proposed as a means that keeps both information of location and feature relationships. It is observed that the number of training epochs of CapsNet is much fewer than that of AlexNet and VGG-16. However, the duration of each epoch consumes much more time due to CapsNet needs to keep the location and details of each feature to create relationships with other features. Therefore, compared to AlexNet and VGG-16, CapsNet requires fewer training data but the model can better predict unseen instances. AlexNet and VGG-16 are considered high complex models, and thus, they require large training dataset. As seen in Figure 5 and Figure 6, the charts reflect the very fluctuation on validation accuracy, contrary to that shown in Figure 7.

Table 2. Summary of performance of the three customized models

	Affine test with -10 to 10 rotation			**Affine test with -30 to 30 rotation**		
	AlexNet	**VGG-16**	**CapsNet**	**AlexNet**	**VGG-16**	**CapsNet**
Accuracy	90.63%	90.79%	86.86%	70.09%	74.17%	80.06%
Sensitivity	89.07%	88.74%	85.43%	68.21%	65.56%	92.72%
Specificity	91.94%	92.50%	88.06%	71.67%	81.39%	69.44%

5. CONCLUSION

In the real world, CXR images are not precisely vertical. It may have various angle distorted from vertical images due to technical or human errors. This paper has studied the performance of the three CNNs in case the datasets contain affined images. The results show that CapsNet outperforms AlexNet and VGG-16 when classifying variant instances unseen in the training dataset. The model is more robust to affine transformations compared to those original CNNs that use pooling layers to reduce parameter computing. In order to provide valuable information to significantly decrease time-to-diagnosis, further improvement on achieving extremely high sensitivity (confidently classify as having TB lesion) is desired as well as further investigation of the proper model best-suited the problem domain should be carried out.

6. REFERENCES

[1] Krizhevsky, A., Sutskever, I., and Hinton, G.E., 2012. ImageNet classification with deep convolutional neural networks. In *Proceedings of the Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1* (Lake Tahoe, Nevada2012), Curran Associates Inc., 2999257, 1097-1105.

[2] Simonyan, K. and Zisserman, A., 2014. *Very Deep Convolutional Networks for Large-Scale Image Recognition*.

[3] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z.B., 2016. *Rethinking the Inception Architecture for Computer Vision*.

[4] He, K., Zhang, X., Ren, S., and Sun, J., 2016. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770-778. DOI= http://dx.doi.org/10.1109/CVPR.2016.90.

[5] Chollet, F., 2017. *Xception: Deep Learning with Depthwise Separable Convolutions*.

[6] Sabour, S., Frosst, N., and E Hinton, G., 2017. *Dynamic Routing Between Capsules*.

[7] Hwang, S., Kim, H.-E., Jeong, J., and Kim, H.-J., 2016. A novel approach for tuberculosis screening based on deep convolutional neural networks. In *SPIE Medical Imaging* SPIE, 8.

[8] Cao, Y., Liu, C., Liu, B., Brunette, M.J., Zhang, N., Sun, T., Zhang, P., Peinado, J., Garavito, E.S., Garcia, L.L., and Curioso, W.H., 2016. Improving Tuberculosis Diagnostics Using Deep Learning and Mobile Health Technologies among Resource-Poor and Marginalized Communities. In *2016 IEEE First International Conference on Connected Health: Applications, Systems and Engineering Technologies (CHASE)*, 274-281. DOI= http://dx.doi.org/10.1109/CHASE.2016.18.

[9] Szegedy, C., Wei, L., Yangqing, J., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A., 2015. Going deeper with convolutions. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1-9. DOI= http://dx.doi.org/10.1109/CVPR.2015.7298594.

[10] Hooda, R., Sofat, S., Kaur, S., Mittal, A., and Meriaudeau, F., 2017. Deep-learning: A potential method for tuberculosis detection using chest radiography. In *2017 IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, 497-502. DOI= http://dx.doi.org/10.1109/ICSIPA.2017.8120663.

[11] Liu, C., Cao, Y., Alcantara, M., Liu, B., Brunette, M., Peinado, J., and Curioso, W., 2017. TX-CNN: Detecting tuberculosis in chest X-ray images using convolutional neural network. In *2017 IEEE International Conference on Image Processing (ICIP)*, 2314-2318. DOI= http://dx.doi.org/10.1109/ICIP.2017.8296695.

[12] Rajaraman, S., Antani, S., Candemir, S., Xue, Z., Kohli, M., and Thoma, G., 2018. *Comparing deep learning models for population screening using chest radiography*.

[13] Ioffe, S. and Szegedy, C., 2015. *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift*.

[14] Stirenko, S., Kochura, Y., Alienin, O., Rokovyi, O., Gang, P., Zeng, W., and Gordienko, Y., 2018. *Chest X-Ray Analysis of Tuberculosis by Deep Learning with Segmentation and Augmentation*.

[15] Rahul, H. and Ajay, M., 2018. Automated Tuberculosis Classification of Chest Radiographs by Using Convolutional Neural Networks. *International Journal of Engineering Technology Science and Research 5*, 3, 310-317.

[16] Kingma, D. and Ba, J., 2014. *Adam: A Method for Stochastic Optimization*.