A bone fracture detection system based on Convolutional Neural Networks using X-ray images

Minhajul Islam
dept. of CSE
Ahsanullah University Of Science and Technology
170104001@aust.edu

Neloy Barman
dept. of CSE
Ahsanullah University Of Science and Technology
180204053@aust.edu

Sanjay Kumar Mandal
dept. of CSE
Ahsanullah University Of Science and Technology
180204039@aust.edu

Ashfiqun Mustari
dept. of CSE
Ahsanullah University Of Science and Technology
180204067@aust.edu

I. Introduction

Nowadays, Bone fractures are one of the most common injuries in the world. An incredible number of people suffer from this disorder and the implications of an untreated fracture may lead to permanent damage or even death.

X-rays, magnetic resonance imaging (MRI), and computed tomography (CT) help to identify fractures. The quality of the clinical image captured from the image capture tool like X-ray, MRI, and CT scans is of poor quality. So it becomes hard for the medical practitioner to detect the abnormality. Because the clinical images to report totally lies on the doctors' experience. Sometimes the physician gives reports of clinical images which is alarming for patients because of their lack of experience.

To overcome the problems, An efficient automatic computer-aided detection system(CAD) needs to be designed which helps the doctor and the radiologist to diagnose several clinical issues.

In this paper, we are using some deep learning models(InceptionV3, Resnet50, Densenet, Vgg-19) to detect fractured bones in the arm using X-rays, using the MURA dataset. The MURA dataset consists of images from musculoskeletal X-Ray of bones. There are 40,561 multi-view radiographic images in the MURA dataset, which are collected from 14,863 studies performed on 12,173 patients. The careful collection of the musculoskeletal radiographic images, the well-defined labels, and a large number of samples make this dataset ideal for Bone Fracture detection with Convolutional Neural Networks.

II. Related Works

The paper [1] focuses on developing a model that classifies the x-ray images in fracture bone and healthy bones. The dataset consists of 100 images of different human bones. The deep neural network gets over fitted due to a small data set. Therefore, an image augmentation technique was has applied on the data set. After augmentation the data set size is 4000, out of which 2000 are of the fractured bone and 2000 are of the healthy bone. The pooling layer and convolution layer (CL) extract features from the images. The accuracy of the proposed model is 92.44% for healthy and fractured bones using 5 fold cross validation. On the other hand, the accuracy on 10% and 20% of the test data is 95% and 93% respectively.

In this paper [2] a deep learning model is used to detect fractured bones in arm using X-rays. The dataset consists of 4000 pictures of arm fracture X-ray radiographs which is collected from MURA dataset. The proposed method is an improved two-stage R-CNN method. Inspired by feature pyramid architecture, a novel backbone network is designed to extract features from preprocessed images. The original images were manually annotated by radiologists. Then, the proposed network was trained by the images with annotated ground-truth bounding-boxes. The accuracy of the model is 62.04%.

The study form [3] represents a binary classification task on the determination of bone fracture & detection of the fractured region using X-ray images. The dataset contains total 3053 images collected from websites and hospital files. 1800 for training and 201 images for testing purpose were used. The remaining were used for the comparison. The fine-tuned ResNet50 with Faster R-CNN worked as the feature extractor. The experimentation result leads to the bone region. Four kinds of texture filtering better identified fractures. The proposed network, CrackNet achieved 88.39% accuracy, 87.5% recall and 89.09% precision.

1931 CT images were collected from a hospital for study [4]. It contained 255 fractured and 732 non-fractured images. VGG-16 and ResNet-50 both showed 98% accuracy on classifying fracture.

III. Dataset

Dataset plays a very crucial part in training deep learning models. It is even more crucial when it comes to training deep learning models to perform medical image classification as a wrong data can lead to wrong diagnosis of disease. Therefore, we have chosen a dataset named MURA, which has been assembled by board-certified radiologists from Stanford

Hospital [5] and is now the largest dataset available on radiographic images.

The MURA dataset consists of images from musculoskeletal X-Ray of bones. The radiography images are of the elbow, forearm, finger, humerus, hand, wrist, and shoulder, which are the seven basic upper extremity radiography studies. Between 2001 and 2012, board-certified radiologists from Stanford Hospital manually classified each image as normal or abnormal at the time of clinical radiography interpretation in a diagnostic radiology setting.

Some of the images from the MURA dataset are shown below-

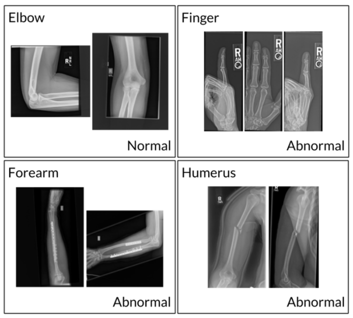

Fig. 1. Radiography Images from MURA Dataset

There are 40,561 multi-view radiographic images in the MURA dataset, which are collected from 14,863 studies performed on 12,173 patients.

The distribution of 14,863 radiographic studies is given below-

TABLE I
DATASET

Study	Number of Studies		
	Normal	Abnormal	Total
Elbow	1186	726	1912
Finger	1320	738	2110
Hand	1598	587	2185
Humerus	592	135	727
Forearm	659	351	1010
Shoulder	1463	1552	3015
Wrist	2274	1423	1912

There is a total of 9045 normal and 5818 abnormal musculoskeletal radiographic studies of the upper extremity including the shoulder, humerus, elbow, forearm, wrist, hand, and finger in the MURA dataset. The 14,863 studies are consisted of a total of 40,561 multi-view radiographic images. The careful collection of the musculoskeletal radiographic images, the well-defined labels, and a large number of samples make this dataset ideal for Bone Fracture detection with Convolutional Neural Networks.

IV. Background Study

In our study, four deep learning algorithms have been implemented to detect bone fracture , which are ResNet50, InceptionV3, VGG19, and DenseNet121.

A. DenseNet121

DenseNet is a modern architecture of neural networks which is used for visual object recognition. It is similar to ResNet. The main building block is DenseBlocks. It is much simpler and more efficient because it is connected directly with every other layer. If there are 'L' layers then there are L(L+1)/2 direct connections. The number 121 in DenseNet121 refers to the number of layers with trainable weights.

In detail, the architecture consists of 5 convolution and pooling layers, 3 transition layers (6,12,24), 1 classification layer (16), and 2 dense blocks (1 * 1 and 3 * 3 Conv). Thus, 5+(6+12+24+16)×2)=121. That is where the name DenseNet121 comes from. All layers are densely connected with the help of Dense Blocks.

DenseNet allows us to reuse features that channel connections. It extends the previous layer's output layer with the next layer in order to maintain the feed-forward characteristics of the system. It also makes communication between layers easy. Here, the l-th layer receives features from all the previous layers as input. The equation stand as Xl = Hl[(X0, X1, X2, . . . , Xl1)]. Here, [(X0, X1, X2 ,. . . , Xl1)] is the combination of output maps of previous layers. Batch Normalization (BN), activation (ReLU), and pooling and convolution(CONV) are major operations of this non-linear transformation function, Hl.

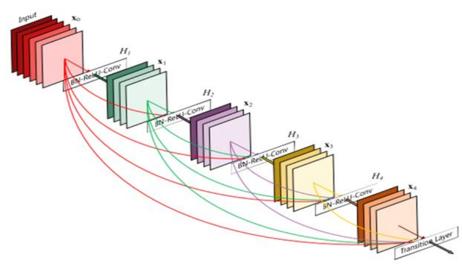

Fig. 2. Block Diagram of DenseNet121

DensNet121 has some great advantages also. It eliminates the vanishing gradient problem, strengthens feature propagation, features reusable, and also reduces the complexity by reducing the number of parameters.

B. VGG19

VGG-19 is a convolutional neural network that is 19 layers deep. We can load a pre-trained version of the network trained on more than a million images from the ImageNet database. The pretrained network can classify images into 1000 object categories, such as keyboard, mouse, pencil, and many animals.

As a result, the network has learned rich feature representations for a wide range of images. So in simple language, VGG is a deep CNN used to classify images. The layers in the VGG19 model are as follows-

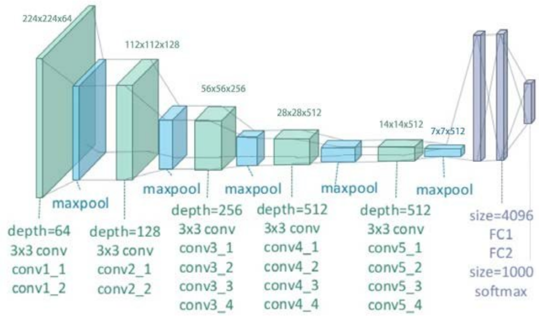

Fig. 3. Block Diagram of VGG19

A fixed size of (224 * 224) RGB image was given as input to this network which means that the matrix was of shape (224,224,3). The only pre-processing that was done is that they subtracted the mean RGB value from each pixel, computed over the whole training set. Used kernels of (3 * 3) size with a stride size of 1 pixel, which enabled them to cover the whole notion of the image. Spatial padding was used to preserve the spatial resolution of the image. Max pooling was performed over 2 * 2-pixel windows with stride 2. This was followed by a Rectified linear unit(ReLu) to introduce non-linearity to make the model classify better and to improve computational time as the previous models used tanh or sigmoid functions which proved much better than those. There are three fully connected layers from which the first two were of size 4096 and after that, a layer with 1000 channels for 1000-way ILSVRC classification, and the final layer is a softmax function.

C. InceptionV3

Inception v3 is an image recognition model that has been shown to attain greater than 78.1% accuracy on the ImageNet dataset. The model is the culmination of many ideas developed by multiple researchers over the years. It is based on the original paper: "Rethinking the Inception Architecture for Computer Vision" by Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna.

The model itself is made up of symmetric and asymmetric building blocks, including convolutions, average pooling, max pooling, concatenations, dropouts, and fully connected layers. Batch normalization is used extensively throughout the model and applied to activation inputs. Loss is computed using Softmax.

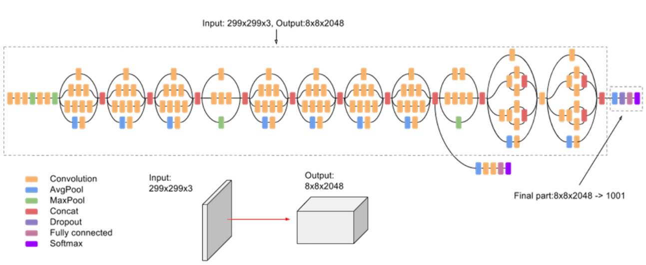

Fig. 4. Diagram of InceptionV3

D. ResNet50

ResNet50 is a convolutional neural network (CNN) that is 50 layers deep which is also one of the most popular Deep residual networks. It was proposed at the ILSVRC 2015.

The ResNet50 architecture consists of following elements: A 7x7 kernel convolution with 64 other kernels and a 2 sized stride. Furthermore, a max pooling layer with a 2 sized stride. Then we have 3 layers consisting of 1x1, 64 kernels, nd 2 more layers with 3x3, 64 kernels, and lastly 1x1, 256 kernels. So, these layers are multiplied by 3 we got 9 layers. In addition, we get 12 more layers which consists of 1×1128 kernels, 3×3128 kernels, and 1×1512 kernels, which is repeated for 4 times. Next 18 more layers consists of 1×1256 cores, 2 cores 3×3256 and 1×1,1024 repeated for 6 times. Following 9 more layers consists of 1×1512 cores, 3×3512 cores, and 1×12048 cores repeated for 3 times. Thus we get total 50 layers to this network. Lastly, Average pooling is done by a fully connected layer with 1000 nodes, using the softmax activation function.

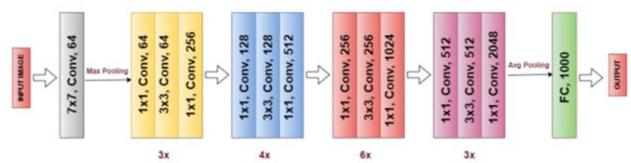

Fig. 5. Block Diagram of ResNet50

V. Evalution Metrics

To evaluate model's performance we have calculated accuaracy, precision, recall & F1-Score for each of the model.

The accuracy of a prediction model is an evaluation metric that measures the accuracy of predictions made by a classification model compared to the actual targets.

$$A_{CC} = \frac{\text{The number of correctly classified targets}}{\text{Total number of targets}} \tag{1}$$

Precision is the fraction of correctly predicted positive values to the total predicted positive observations.

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

The recall is the fraction of actual positives that are identified correctly. It's a measure of how well the classifier determines the positive examples, and it's also a measure of how well the classifier identifies the correct class.

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

The F1-score is the weighted average of Precision and Recall.

$$F1 - Score = 2x\frac{PrecisionxRecall}{Precision + Recall} \tag{4}$$

VI. Experimental results

In our first experiment, we added extra layers at the top of each pre-trained model to conclude our mutlti-class classification. Then we trained those layers with the X-ray image dataset keeping the base architecture of those models constant. The models show poor performance. Then, we unfreeze some layers from the base architecture of each model. We added predefined weights to handle class imbalance. Then we trained some layers from the pre-trained models along with our added extra layers. The performance improved a bit from the previous experiments. The following tables show the performance metrics for each of the models.

TABLE II
RESULTS OF DENSENET-121

Class	DenseNet-121			DenseNet-121(Adding Weight)		
	Precision	Recall	F1-score	Precision	Recall	F1-score
Fractured Elbow	80	58	67	84	60	70
Fractured Finger	75	46	57	75	54	63
Fractured Forearm	84	40	54	85	44	58
Fractured Hand	84	17	28	85	23	36
Fractured Humerus	77	61	68	77	59	67
Fractured Shoulder	75	63	68	72	65	68
Fractured Wrist	80	58	68	79	58	67
Non-Fractured Elbow	60	95	74	62	95	75
Non-Fractured Finger	54	76	63	57	70	63
Non-Fractured Forearm	75	78	77	75	81	78
Non-Fractured Hand	61	94	74	62	94	75
Non-Fractured Humerus	62	74	67	60	69	64
Non-Fractured Shoulder	68	84	75	68	80	74
Non-Fractured Wrist	69	89	78	68	90	77
Accuracy	69.81%			71.05%		

TABLE III
RESULTS OF VGG-19

Class	VGG-19			VGG-19(Adding Weight)		
	Precision	Recall	F1-score	Precision	Recall	F1-score
Fractured Elbow	57	38	46	56	52	54
Fractured Finger	71	62	67	68	81	74
Fractured Forearm	78	9	17	36	35	35
Fractured Hand	40	4	7	45	52	49
Fractured Humerus	62	52	57	50	68	58
Fractured Shoulder	62	64	63	67	67	67
Fractured Wrist	65	53	59	67	58	62
Non-Fractured Elbow	46	74	57	54	61	58
Non-Fractured Finger	60	61	61	74	56	64
Non-Fractured Forearm	45	53	48	40	47	43
Non-Fractured Hand	50	85	63	61	59	60
Non-Fractured Humerus	59	48	53	52	53	53
Non-Fractured Shoulder	59	71	65	66	66	66
Non-Fractured Wrist	59	71	64	69	59	64
Accuracy	55.90%			58.02%		

TABLE IV
RESULTS OF INCEPTIONV3

Class	InceptionV3			InceptionV3(Adding Weight)		
	Precision	Recall	F1-score	Precision	Recall	F1-score
Fractured Elbow	74	69	71	74	69	71
Fractured Finger	85	71	77	85	71	77
Fractured Forearm	76	57	65	76	57	65
Fractured Hand	55	67	61	55	67	61
Fractured Humerus	76	77	76	76	71	73
Fractured Shoulder	76	71	73	75	67	71
Fractured Wrist	75	67	71	70	86	77
Non-Fractured Elbow	70	86	77	72	82	77
Non-Fractured Finger	72	82	77	72	82	77
Non-Fractured Forearm	70	77	74	70	77	74
Non-Fractured Hand	74	65	70	74	65	70
Non-Fractured Humerus	81	82	82	81	82	82
Non-Fractured Shoulder	76	81	78	76	81	78
Non-Fractured Wrist	79	83	81	79	83	81
Accuracy	69.81%			74.12%		

TABLE V
RESULTS OF RESNET-50

Class	ResNet-50			ResNet-50(Adding Weight)		
	Precision	Recall	F1-score	Precision	Recall	F1-score
Fractured Elbow	57	38	46	56	52	54
Fractured Finger	71	62	67	68	81	74
Fractured Forearm	78	9	17	36	35	35
Fractured Hand	40	4	7	45	52	49
Fractured Humerus	62	52	57	50	68	58
Fractured Shoulder	62	64	63	67	67	67
Fractured Wrist	65	53	59	67	58	62
Non-Fractured Elbow	46	74	57	64	61	58
Non-Fractured Finger	60	61	61	74	56	64
Non-Fractured Forearm	45	53	48	40	47	43
Non-Fractured Hand	50	85	63	61	59	60
Non-Fractured Humerus	59	48	53	52	53	53
Non-Fractured Shoulder	59	71	65	66	66	66
Non-Fractured Wrist	59	71	64	69	59	64
Accuracy	45.34%			49.29%		

References

[1] D. Yadav and S. Rathor, "Bone fracture detection and classification using deep learning approach," in *2020 International Conference on Power Electronics & IoT Applications in Renewable Energy and its Control (PARC)*. IEEE, 2020, pp. 282–285.

[2] B. Guan, G. Zhang, J. Yao, X. Wang, and M. Wang, "Arm fracture detection in x-rays based on improved deep convolutional neural network," *Computers & Electrical Engineering*, vol. 81, p. 106530, 2020.

[3] Y. Ma and Y. Luo, "Bone fracture detection through the two-stage system of crack-sensitive convolutional neural network," *Informatics in Medicine Unlocked*, vol. 22, p. 100452, 2021.

[4] Y. D. Pranata, K.-C. Wang, J.-C. Wang, I. Idram, J.-Y. Lai, J.-W. Liu, and I.-H. Hsieh, "Deep learning and surf for automated classification and detection of calcaneus fractures in ct images," *Computer methods and programs in biomedicine*, vol. 171, pp. 27–37, 2019.

[5] P. Rajpurkar, J. Irvin, A. Bagul, D. Ding, T. Duan, H. Mehta, B. Yang, K. Zhu, D. Laird, R. L. Ball *et al.*, "Mura: Large dataset for abnormality detection in musculoskeletal radiographs," *arXiv preprint arXiv:1712.06957*, 2017.