International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 05 | May 2020 www.irjet.net p-ISSN: 2395-0072

Abnormality Classification and Detection in Musculoskeletal Radiographs

Rohit PrasannaKumar[1], Kiran Khatkale[2], Gauri Raghuwanshi[3], Shashank Pramod[4], Satishkumar Varma[5]

[1,2,3,4] BE Student, Dept. of Computer Engg. Pillai college of Engineering, New Panvel, Maharashtra, India
[5] Professor and Head, Dept. of Information Tech. Pillai college of Engineering, New Panvel, Maharashtra, India

---***---

Abstract - *Bone fracture is created by stresses higher than the bone can bear. It leads to bone discontinuity. Healing of fracture takes time. Timely diagnostic and effective detection of bone fracture is essential to minimize the complications. The conventional radiography is the most common means of bone fracture evaluation in clinical practice. In this paper image processing techniques are used to detect bone fracture. The MURA standard dataset is used as a ground truth in the experiment to train and detect abnormalities on elbow, forearm, hand, humerus, shoulder, finger and wrist studies. The detection of abnormalities on various bones of different body part's radiographs. The experimental results are compared radiologist's results available in the MURA standard dataset to analyze the performance.*

***Key Words*: Bone fracture, Abnormalities, Canny Edge Detection, Sobel Edge Detection, Prewitt Edge Detection, Convolutional Neural Network, ResNet, DenseNet, Image Augmentation and Flask.**

1. INTRODUCTION

Bones are the rigid organs in the human body which protects important organs such as heart, lungs, brain and any other internal organs. The human body has 206 bones. The largest bones are femur bones, and the smallest bones are auditory ossicles. Fracture of a bone is a common problem in human beings. It can occur due to accidents or any other case in which pressure is applied on the bones. There are different types of bone fracture that occur: oblique, compound, comminuted, spiral, greenstick and transverse.

2. LITERATURE SURVEY

The detection of bone fracture is done manually. Which means the doctor or the radiologist just looks at the x-ray image and prepares the results accordingly. Mahmoud Al-Ayyoub et al. [1] proposed the idea of an efficient system for quick as well as accurate diagnosis of hand bone fractures based on the data gained from the x- ray images. Input is taken which are many labeled x-rays images. These x- rays are of hand bones. The set contains both normal bones as well as fractured bone images. At first these images are enhanced using filtering algorithms so as to remove the noise from the image. Then these images are processed further for edge detection using some set of methods. Tools like Wavelet and Curvelet transforms are used for converting each image into a set of features. Based on the extracted images, classification algorithms are built. At the end the performance and accuracy of the images are evaluated. The results come out to be 91.8%. Pierre Sermanet et al. [2] proposed a way to show that training a convolutional network to classify, locate and detect objects in images, can boost the classification accuracy, detection and localization accuracy of all tasks. A single ConvNet is proposed as a new integrated approach for object detection, recognition and localization. Also, introduction of a novel method for localization and detection. Many localization predictions are combined so that detection can be performed without training on background samples. This method avoids more consumption of time and lets the network focus more on positive classes for higher accuracy S. Febrianto Kurniawan et al. [3] proposed a system that detects bone fracture using Canny Edge Detection Algorithm. This system uses the OpenCV library combined with the Canny Edge Detection algorithm to detect the bone fracture. This method is an optimal edge detection algorithm on determining the end of a line with a threshold that can be changed and less error rate. Nancy Johari et al.[4] discussed to find out the accuracy of a bone fracture detection using the Canny Edge Detection Algorithm. This framework will provide more accuracy with less effort and time. While using the Sobel operator the parameter sigma is kept at 4.75 which help to enhance the efficiency of the system. It also helps to diagnose the hairline fracture more effectively. Using this value, edges can be treated in such a way that all the distortions and joints are visible that increases the success rate of the system. Pranav Rajpurkar et al. [5] proposed a system which could be able to detect abnormalities in bones by analysing the x-ray images of bones. The system is able to highlight the area of fracture that occurred in the bone. The result comes out to be accurate and could provide more details on the fracture. In order to get the results, 169 layers Convolutional Neural Network model is used. Before training the model, image processing techniques are used. They are mainly the edge detection algorithms so as to detect the edge of the bones from the given radiographs. Mariam M Saii et al. [6] devised a novel method for fracture detection and classification. The basic stages include pre-processing of bone image and structured operations to obtain the ROI region which is manipulated by a post processing stage to remove non-fracture pixels. The approach extracts three features from bone image. Mainly transverse, cracks and divergence features in order to define the fracture type or integrity of bone image. 92% true detection rate was achieved for general bone fractures, 93.33% true detection rate for finger bone fractures and 93.33% true rejection rate. Wint Wah Myint et al. [7] proposed an idea to detect fracture or non-fracture and classify the type of fracture of the leg bone Tibia in x-ray image. The system is developed with three steps mainly Pre-processing, Feature Extraction and Classification. A sharpening technique is used in pre-processing known as Unsharp Masking (USM). This technique enhances the image and highlights the edges in the image. The sharpened image is processed further for feature extraction by using Harris Corner detection algorithm to extract corner feature points. Two approaches are used for detection and classification of fractures. Simple Decision Tree is used for detection of fracture and K-Nearest Neighbor (KNN) is used for classification of fracture types. Dennis Banga et al. [8] developed the ensemble200 model. The current model by Pranav et al.[5] which is 169 layer DenseNet on the abnormality detection task, lacks in performance. As the overall performance was lower than the results of the radiologists. The ensemble200 model scored 0.66 Cohen Kappa which is lower than the DenseNet model but the model performance with the F1 score is more as compared to the DenseNet model. The Cohen Kappa score variability with the different studies is lower. The best Cohen Kappa score on the upper extremity studies is 0.7408 for wrist and the lowest is 0.5844 for hand. Whereas the ensemble200 outperformed DenseNet model on the finger studies with a Cohen Kappa score of 0.653 showing reduced performance variability on the model performance.

Summary of Related Work

The overview of different works is given in Table 1.

Table 1. Overview of literature survey

Literature	Advantages	Disadvantages
Mahmoud Al Ayyoub et.al September 2013 [1]	Accuracy is 91.8%	Low quality images equals less accuracy
Pierre Sermanet et al. February 2014 [2]	No training on background samples and less time consumption	No use of back-propping throughout the whole network
S. Febrianto Kurniawan et al. June 2014 [3]	Tested with real data and implemented successfully	Needs to be improvised and reduce response time
Nancy Johari et al. January 2018 [4]	More accurate results with less effort and time	Value of edge detection not perfect so as the edges shown
Pranav Rajpurkar et al. May 2018 [5]	Accurate results similar to that of radiologists	Bad alignments and imperfect orientation of images
Mariam M Saii et al. August 2018 [6]	Experimented on 155 bone images with 93% accuracy	Less quality of image results in less accuracy
Wint Wah Myint et al. October 2018 [7]	Accurate and reliable results as well as performance	Less accuracy resulted due to less quality
Dennis Banga et al. August 2019 [8]	High performance on finger studies as compared to DenseNet model	Less performance on wrist and hand studies as compared to DenseNet model

3. PROPOSED WORK

The X-Ray images are obtained from a dataset known as MURA. Stanford University has a huge amount of X-Ray images. These are bone X-Ray images for many people. The bones are of forearms, humerus, femur, wrist, etc. These X-Ray images are first converted to black and white. The black and white images may contain noise in them. So firstly, the noises in images are removed. Then we apply an Edge Detection Algorithm for obtaining the edges of the images. When we have the edges of the bone then it could be easy to find the abnormal region of the bone. When we get the abnormal region, we get the accuracy as high. Accuracy tends to be low as sometimes it could not find the abnormality. Here we use the Convolutional Neural Network (CNN) for classifying the images between abnormal and normal sets of output. Due to edged image input the model finds it easy to classify the images. After this process we proceed towards DenseNet and ResNet. These methods have been adopted in order to detect the abnormality of the bone which means the fractures occurred in bones. The input is given which is an X-ray image. The system converts the normal image into a negative image using Image Augmentation. After which the system could detect and highlight the abnormal region of the bone on the image. Once detected the system tends to calculate the percent of abnormality found on the bone. These are computer generated results which provides more accuracy rather than the doctor manually looks up and predicts the fracture. The final procedure is shaping the system in an interface using Flask which is a python web framework. The design UI is kept simple so that any user could use it easily.

3.1 SYSTEM ARCHITECTURE

Figure 1 shows the system architecture for classification of normal and abnormal conditions of the bones in the images.

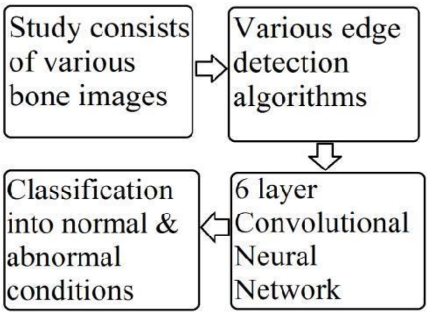

Figure 1. Proposed System Architecture

A. Input Block Description:

The first block consists of the input which we have to give to the system. The input consists of various bone x-ray images. In which some images may contain normal or abnormal conditions of bones. MURA dataset provides more than 14,000 images. The images are of high quality. Then these images are processed further.

B. Edge Detection Algorithms:

The second part is to detect the edges of the bones. Canny, Sobel and Prewitt are the edge detection algorithms used in our system. Each provides different accuracy levels in detecting the edges. We use the edges so that our model is able to classify easily between the normal and the abnormal conditions of the bone. Different methods for extracting edges provide different accuracy results. The algorithm containing higher accuracy will be feasible to use.

C. 6 Layer Convolutional Neural Network:

The input to this model is the edge detected images. Convolutional networks (ConvNets) are inherently efficient when we apply in a sliding fashion as they naturally share computations common to overlapping regions. While applying a network to larger images at test time, we simply apply each convolution over the extent of full image. This extends the output of each layer which covers the new size of image, therefore producing a map of output class predictions, with one spatial location for each “window” (field of view) of input.

D. Output Block Description:

At the end we get the final output in which the images are classified into normal and abnormal conditions. Referring to these images we could get an idea of which is a normal image, which means the bone shown in the image is free of fracture and the other one the abnormal image image, which means the bone shown in the image contains a fracture. Figure 2 shows the architecture of the Convolutional Neural network.

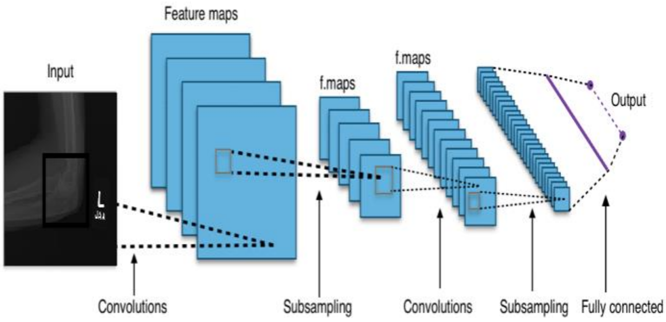

Figure 2. Convolutional Neural Network

Image Augmentation: In order to achieve good performance, deep networks require a high amount of training data. To boost the performance of deep networks, image augmentation is required. Multiple processing is carried onto the training images. In our case we need image augmentation for converting the normal image to a negative image. Due to which highlighting the abnormality becomes easy.

ResNet and DenseNet: Another architecture we come across is DenseNet and ResNet. Residual Neural Network is a kind of Artificial Neural Network. This network is capable of training hundreds or even thousands of layers and still achieves compelling performance. The base block of ResNet is a residual block. When we proceed further through the large number of layers, the computation becomes more complex. Every layer tries to learn some underlying mapping of the desired function and instead of having these blocks; we try and fit a residual mapping. We can get an idea of ResNet by looking at Figure 3 which is architecture of ResNet 101. In the case of DenseNet, the network proposes concatenating the outputs from the previous layers instead of using the summation. Here the reuse of the residuals are high, which creates a deep supervision because every layer receives more supervision from the previous layer and thus loss function will react accordingly and due to this methodology, it makes DenseNet a more powerful network. In our project we use these networks for training of multiple images and get the abnormalities of the bones in the end. We get a percentage value on how much abnormality of the bone is detected. Figure 4 shows the architecture of DenseNet 121.

Flask Python: At the end of all the procedures we shape them for user interaction. We have used Flask as GUI in our system. It is a web framework which is developed using python. Figure 5 shows architecture of flask.

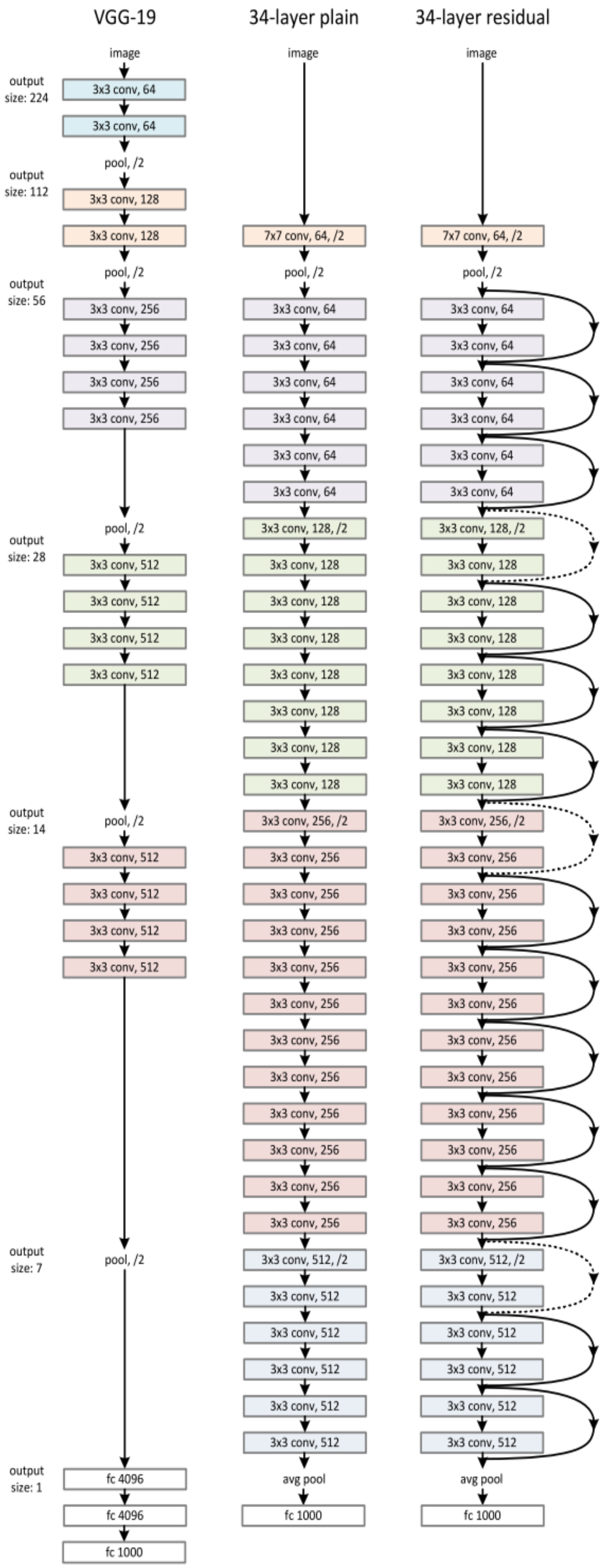

Figure 3. ResNet 101 architecture

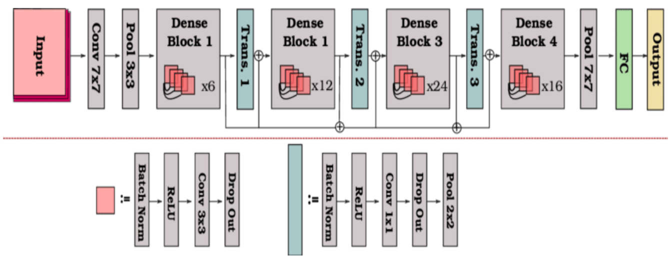

Figure 4. DenseNet 121 Architecture

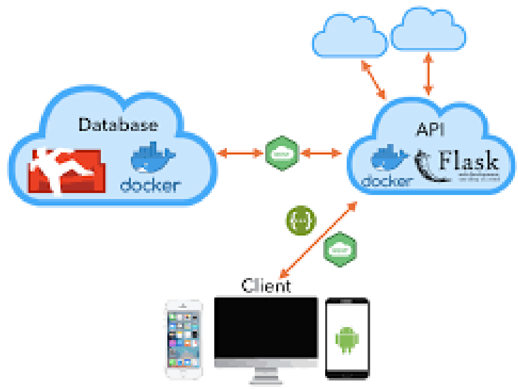

Figure 5. Flask Architecture

Dataset and Parameters

A dataset of x-rays consisting of a total of 40,561 multi-view radiographic images. Around 14,863 studies from 12,173 patients. With each belonging to one of seven standard radiographic study types: elbow, finger, forearm, hand, humerus, shoulder, and wrist. The MURA abnormality detection task is a binary classification task, where the input is an upper extremity radiograph study where each study contains one or more images and the expected output is a binary label either 0 or 1 indicating whether the study is normal or abnormal, respectively. [5]

Table 2. Dataset used for project [5]

Study	Train		Validation		Total
	Normal	Abnormal	Normal	Abnormal	
Elbow	1094	660	92	66	1912
Finger	1280	655	92	83	2110
Hand	1497	521	101	66	2185
Humerus	321	271	68	67	727
Forearm	590	287	69	64	1010
Shoulder	1364	1457	99	95	3015
Wrist	2134	1326	140	97	3697
Total No. of Studies	8280	5177	661	538	14656

Table 2 shows the total number of images in the dataset by MURA. These images are collected from Stanford University. These are x-rays of various bones of various patients. Containing x-ray images of both normal bone as well as the fractured bone.

4. RESULTS

At the very first we look for the results of classification of the images. The images are processed which means edge detected and then sent for training for classification. The classifications of images are done through the Convolutional Neural Network. At the end we get some set of results. The network provides the training results from which we get the Accuracy and Loss for classifying the images in normal and abnormal conditions.

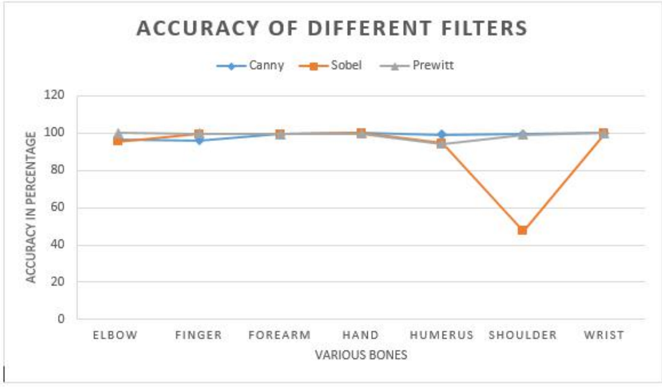

Figure 6. Graph showing accuracy in percentage of various bones

As we can see in Figure 6 the accuracy which we get for 3 different filters for the various bones when trained in Convolutional Neural Network. This accuracy is dependent on how clean the images are. If the images are not of high quality, the accuracy tends to be very low automatically. The graph in Figure 7 shows the loss of classification by the Convolutional Neural Network.

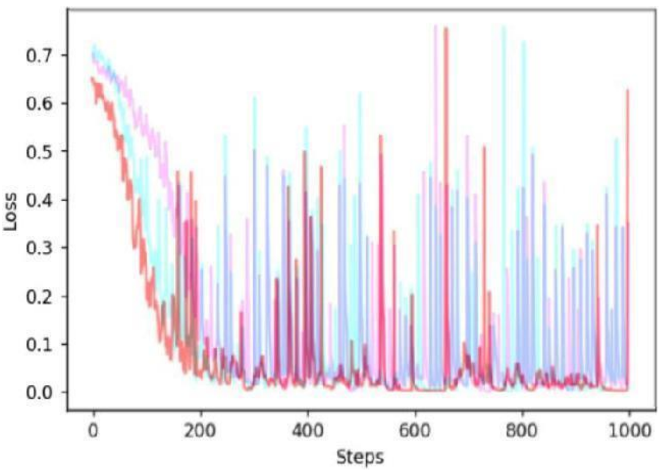

Figure 7. Loss graph with Canny, Sobel and Prewitt filter represented by color Red, Magenta and Cyan respectively.

Now let us look for the results of ResNet 101 and DenseNet 121 which is used for detecting the abnormalities in the bone. All the various bone images are given as input and sent for training through ResNet 101 and DenseNet 121. As we can see in Table 3 the results obtained from both the networks.

Table 3. Results of trained model

Model	Single View Accuracy	Multi View Accuracy	Single View Kappa	Multi View Kappa
ResNet	81.1	83.1	0.619	0.653
ResNet (w test aug)	81.9	84.1	0.637	0.675
DenseNet	83.2	84.4	0.663	0.682
DenseNet (w test aug)	82.5	84.8	0.649	0.692

We have around 36800 train images. All the images are classified in normal and abnormal classes. We can see in Figure 8 how much images we get in normal class and abnormal class respectively

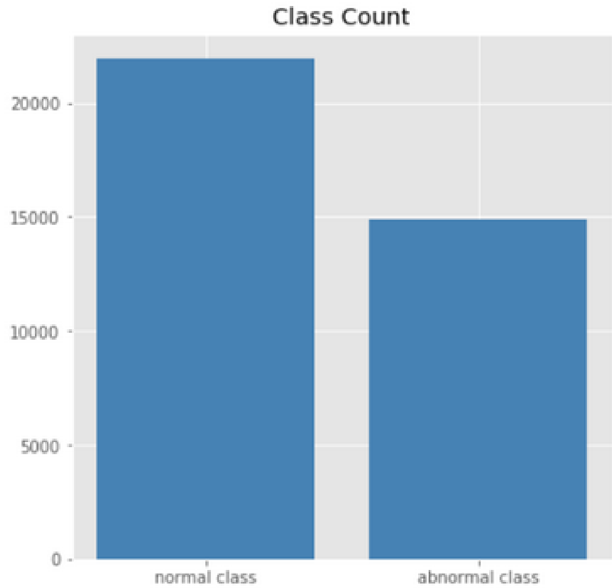

Figure 8. Normal and abnormal classification of the trained images

In the same way we get classification of various bones in normal and abnormal classes. We can see the same in the graph shown in Figure 9.

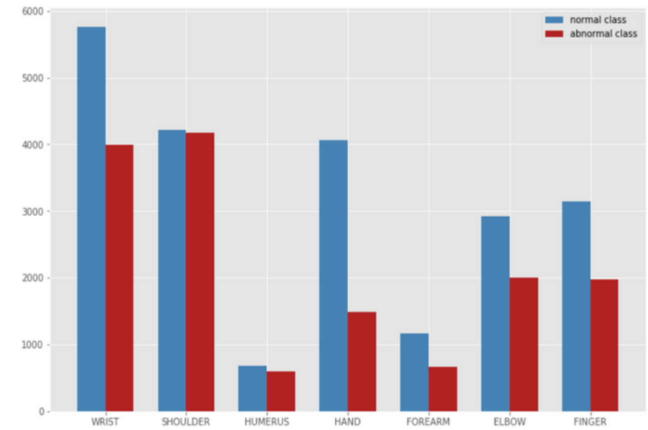

Figure 9. Class count by bone type for train data

Classification of the train images makes it easy for further procedures and gets the results accordingly.

Output / Screenshots:

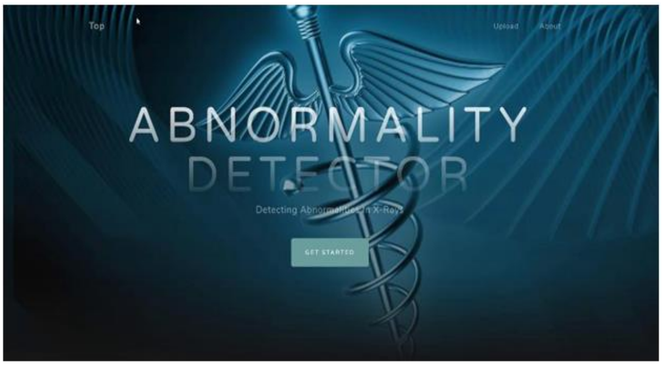

Figure 10. System on startup

The figure 10 shows the first view of the system. It will take x ray as input that wants to be detected. On proceeding we get another image as output in which the system is able to show where the fracture has occurred and also the percentage of confidence on its detection.

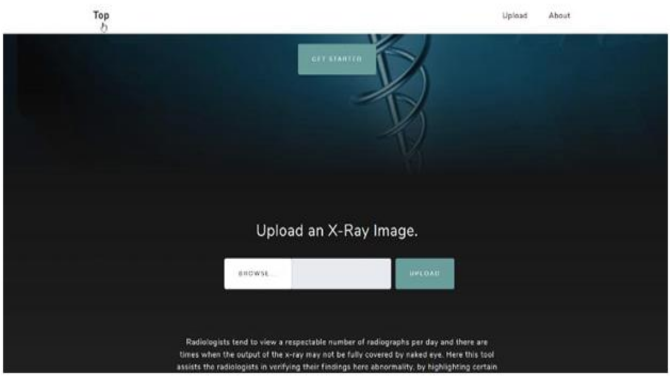

Figure 11. Section for uploading an image

Figure 11 shows the area where the user can browse for required x-ray images and upload to the system in order to detect any abnormality.

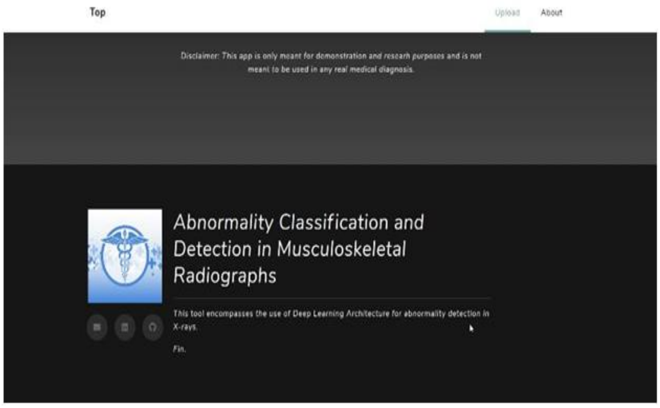

Figure 12. About me section of the system.

Figure 12 shows the developers of the system and any message if they need to convey to the users.

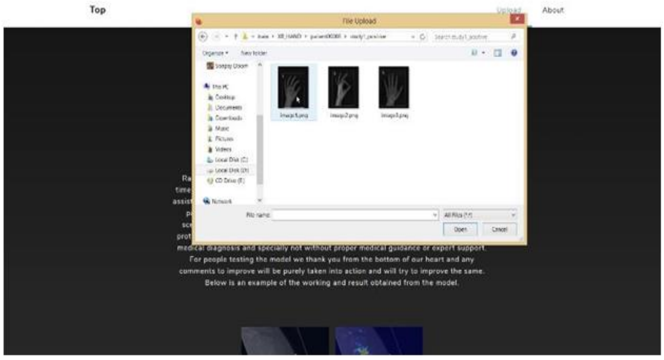

Figure 13. Files upload section.

The user can upload a required image from their device system as seen in Figure 13.

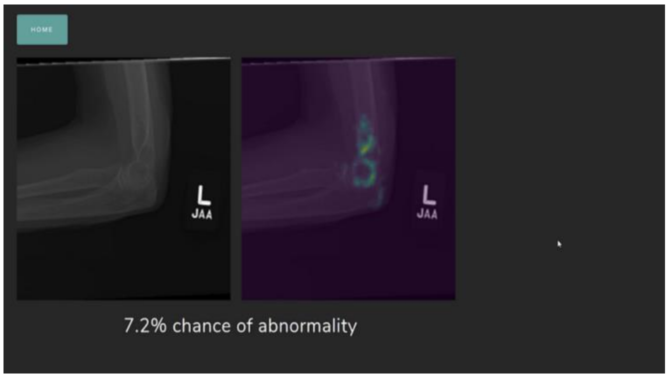

Figure 14. Area where the detected image result is seen.

Once the image is selected and clicked on upload, some time is taken in order to get the result. Then we get the result. This resultant image is in negative form. It also contains the highlighted area of fracture. In Figure 14 the result can be seen of a normal bone x-ray image. Therefore the percentage of abnormality detected is very less.

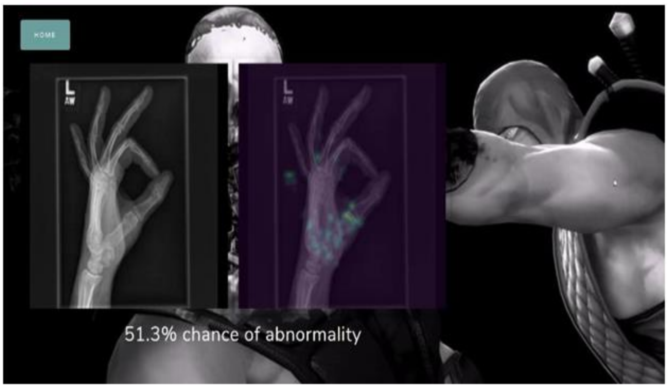

Figure 15. The resultant image of normal bone.

We can see in Figure 15 that the result comes with a high percent of abnormality. As we can see some problems in the bone which causes high percent of abnormality detection.

5. REQUIREMENT ANALYSIS

We have used some set of software in order to develop and train our system. They mainly are Tensorfow, Anaconda and Jupyter Notebook.

~Tensorflow is a symbolic math library and is used for ML applications such as neural networks. We use this for the classification of bones whether abnormal or normal.

~Anaconda is a free and open-source distribution of Python as well as R programming languages for scientific computing, that aims to simplify package management and deployment.

~Jupyter notebook is created to develop open-source software, open-standards, and services for interactive computing across dozens of programming languages.

CONCLUSION

The project is intended to make software or a web framework that would help people. This system would be able to detect the abnormalities of bones in X-Ray images. The exact location of the wound could be known easily. With prior knowledge of the wound early medication could be taken without any much delay. This would help the doctors to identify the fracture of bone and could suggest proper medication accordingly. Hence we have learnt new technologies and algorithms while making this project. We have learnt about image processing which are the edge detection methods from an image, its steps and its implementation. Then we have used the Convolutional Neural Network for Image Classification, analyzing how the model works, how output is obtained. We have detected the abnormality using DenseNet and ResNet which are deep learning methods. We have used Flask Python which is a web framework for GUI of our system. The framework would be hosted, therefore making all the users to access the system.

REFERENCES

[1] Mahmoud Al-Ayyoub, Ismail Hmeidi, Haya Rababah Detecting Hand Bone Fractures in X- Ray images (September 2013).

[2] Pierre Sermanet and Xiang Zhang OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks arXiv: 1312.6229v4 (February 2014)

[3] S. Febrianto Kurniawan, I.K.G. Darma Putra and A.A. Kompiang Sudana Bone Fracture Detection using OpenCV (June 2014)

[4] Nancy Johari and Natthan Singh Bone Fracture Detection using Edge Detection Technique (January 2018)

[5] Pranav Rajpurkar1, Jeremy Irvin1, Aarti Bagul, Daisy Ding, Tony Duan, Hershel Mehta, Brandon Yang, Kaylie Zhu, Dillon Laird, Robyn L. Ball, Curtis Langlotz, Katie Shpanskaya, Matthew P. Lungren, Andrew Y. Ng MURA: Large Dataset in Musculoskeletal Radiographs arXiv:1712.06957v4 (May 2018)

[6] Mariam M Saii and Ali M Mayya Hand and Finger Fracture Detection and Type Classification Using Image Processing (August 2018)

[7] Wint Wah Myint, Khin Sandar Tun, Hla My Tun Analysis on Leg Bone Fracture Detection and Classification Using X-ray Images (January 2018)

[8] Dennis Banga, Peter Waiganjo Abnormality Detection in Musculoskeletal Radiographs with Convolutional Neural Networks (Ensembles) and Performance Optimization. arXiv:1908.02170v1 [cs.CY] **(**August 2019)

[9] Jianan Li, Xiaodan Liang, Yunchao Wei, Tingfa Xu, Jiashi Feng, Shuicheng Yan Perceptual Generative Adversarial Networks for Small Object Detection.arXiv:1706.05274 (June 2017)

[10] Thomas Schlegl, Philipp Seeb¨ock, Sebastian M. Waldstein, Ursula Schmidt-Erfurth, and Georg Langs Unsupervised Anomaly Detection with Generative Adversarial Networks to Guide Marker Discovery. ArXiv: 1703.05921 (March 2017)

[11] Jie Hu, Li Shen, Samuel Albanie, Gang Sun, Enhua Wu Squeeze-and-Excitation Networks. ArXiv: 1709.01507v4 [cs.CV] (May 2019)

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun Deep Residual Learning for Image Recognition. ArXiv: 1512.03385v1 [cs.CV] (December 2015)