Detecting Hand Bone Fractures in X-Ray Images

Mahmoud Al-Ayyoub, Ismail Hmeidi, Haya Rababah
Jordan University of Science and Technology
Irbid, Jordan
{maalshbool, hmeidi}@just.edu.jo, hrababah07@cit.just.edu.jo

ABSTRACT: *Computer aided diagnosis is a hot research field. Systems with the ability to provide a highly accurate diagnosis using little resources are highly desirable. One type of such systems depend on medical images to provide instantaneous diagnosis based on some discriminative features extracted from the images after processing them for noise removal and enhancement. In this paper, we propose a system to automatically detect fractures in hand bones using x-ray images. To the best of our knowledge, this problemhave never been addressed before. For a first attempt to tackle such a difficult problem, our system performed incredibly good with a 91.8% accuracy.*

Keywords: Noise Removal, X-Ray Images, Fractures Diagnosis, Medical Images

Received: 28 May 2013, Revised 30 June 2013, Accepted 6 July 2013

1. Introduction

Diagnostic medical imaging tools are invaluable. Computed Tomography (CT), Magnetic Resonance Imaging (MRI), and x-rays are examples of such tools which help physicians in detecting different types of abnormalities [39]. Quick and accurate diagnosis can be crucial to the success of any prescribed treatment. Depending on human experts alone for such a critical matter have caused intolerable errors. Hence, the idea of automating the diagnosis procedure has always been an appealing one.

As with other computer-aided diagnosis systems, the motivations for building this system are: (i) reducing human errors (it is wellknown that the performance of human experts can drop below acceptable levels if they are distracted, stressed, overworked, emotionally unbalanced, etc.) and (ii) reducing the time/effort associated with training and hiring physicians. Eventually, this system can be integrated within the software of the x-ray imaging devices to enable users to produce a quick and highly-accurate diagnosis while generating the image.

Another motivation for our work is to help doctors, patients and researchers look for certain cases for research purposes as follows [15]. In modern hospitals, medical images are stored in the standard DICOM (Digital Imaging and Communications in Medicine) format which includes text into the images. Any attempt to retrieve and display these images must go through PACS (Picture Archives and Communication System) hardware [21]. This requires that the name of the patient or identity card number is provided to find any particular image. Thus, searching for some type of cases (e.g., for research purposes) is usually done manually, which is a very expensive task in terms of time and effort. Providing a tool that can go through a huge database of images and automatically identify the required cases quickly and with high accuracy can save huge amounts of time and effort.

Finally, note that searching through the written reports is not sufficient for this task due to the large number of mistakes in such records. This was observed from personal experience and confirmed by many experienced physicians.

X-ray images (or Radiographs) are among the most common ways to detect problems in bones as well as other organs of the human body. The output image is a shadow-like image. Although CT and MRI images give better quality images for body organs than x-ray images, the latter are faster cheaper, enjoy wider availability and are easier to use with few limitations [4]. Moreover, the level of quality of x-ray images is enough for the purpose of bone fracture detection.

Bones are the solid organs in the human body protecting many vital organs such as brain, heart, lungs, etc. The human body contains 206 bones with various shapes and structures.The largest bones are the femur bones, and the smallest bones are the auditory ossicles. There are five types of bones: long, short, irregular, sesamoid and flat. The focus of this research is on the short and sesamoid bones of the hand as shown in Figure 1. Due to limitations in dataset collection, we focus on two parts of hand bones: metacarpals and phalanges, and ignore carpal bones.

Bones can suffer fractures in spite of their rigidity. Bone fractures can occur due to a simple accident or any other scenario in which a high pressure is applied on the bones. There are many types of bone fractures: simple, oblique, compound, comminuted, spiral, greenstick and transverse [35], [48]. In this work, we will consider the problem of detecting fractures in hand bones without paying attention to the type of fracture. To the best of our knowledge, no prior work have addressed this problem.

This paper consists of four more sections. In the first one (Section 2), a general overview of the literature is presented. The following two sections (Sections 3 and 4) discuss the proposed method and the set of experiments conducted to evaluate its performance. In the last section, conclusion is given and future directions are discussed.

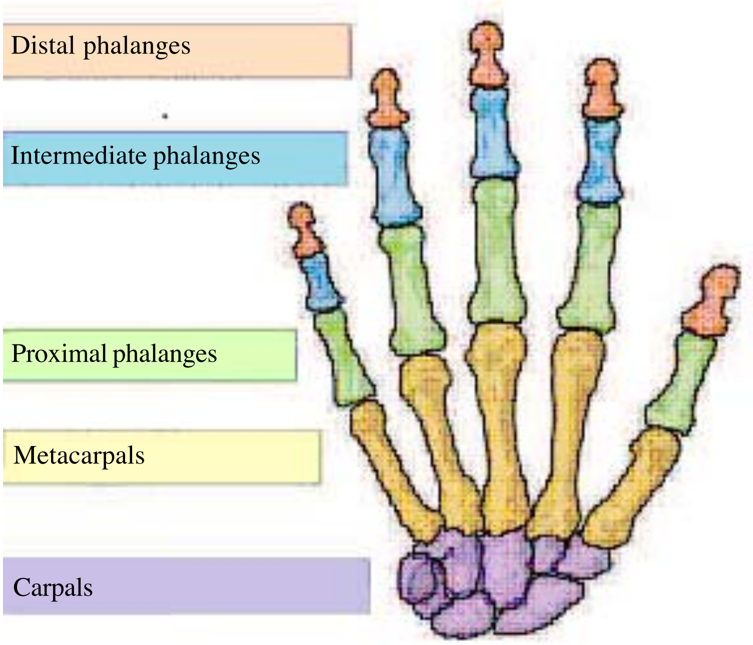


Figure 1. Hand bones scheme [49]

2. Related Works

A broad overview of the literature is presented in this section starting with papers that have a general take on the classification problem on diverse medical datasets and the problems faced therein. Tanwani et al. [43] provide a comparison of six different classifiers on 31 datasets. They follow a general approach consisting of a preprocessing step to remove any redundancy followed by a classification step that may contain enhancements of the classifiers (either individually using bagging tep that may contain enhancements of the classifiers (either individually using baggingstep that may contain enhancements of the classifiers (either individually using bagging and boosting techniques or as a group using stacking and voting techniques.)

Mena et al. [36] consider the problem of imbalanced datasets in medical diagnosis and suggest a rule induction algorithm consisting of three steps: attributes selection, partitions selection and rule construction.

Since we propose a computer aided diagnosis system for which the only sources of information are medical images, it is important to discuss various image preprocessing and enhancement techniques. Specifically, the focus here is on removing different types of noise such as Gaussian, salt and pepper, etc. In [47], the authors present a filtering algorithm for Gaussian noise removal. After estimating the amount of noise corruption from the noise corrupted image, the authors replace the center pixel by the mean value of the sum of the surrounding pixels based on a threshold value. Compared to other filtering algorithms such as mean, alpha-trimmed mean, Wiener, K-means, bilateral and trilateral, this algorithm gives lower *Mean Absolute Error* (*MAE*) and higher *Peak Signal-to-Noise Ratio* (*PSNR*). In [3], the authors propose an extension of the K-fill algorithm to remove salt and pepper noise based on the number of black or white pixels in a 3 × 3 window. In [23], the authors propose an iterative algorithm based on the Expectation Maximization (EM) approach for noise removal.

Assuming that the observations are corrupted by the noise modeled as a sum of two random processes: a Poisson and a Gaussian, this approach allows them to jointly estimate the scale parameter of the Poisson component and the mean and variance of the Gaussian one. Finally, in [52], the authors address the problem of image enhancement and speckle reduction using filtering techniques. Using histogram analysis, they compare different filters: Wiener, average and median filters, and show that the Wiener filter is a better technique for speckle reduction without fully eliminating the image edges.

The following step is feature extraction. Standard edge detection techniques such as Canny [9], Sobel and Laplacian represent an obvious first choice for this step. Other relevant techniques exist. In [46], the authors use the Contourlet transform algorithm for edge detection, and compare it against other edge detection algorithms. In [53], the authors propose a novel multi-scale nonlinear structure tensor based corner detection algorithm to improve the classical Harris corner detector. By considering both the spatial and gradient distances of neighboring pixels, a nonlinear bilateral structure tensor is constructed to examine the image local pattern. Finally, In [12], the authors propose a novel process of feature selection by using three different methods from Wavelet transform to select a subset of the coefficients (features) of each method. They also use the nearest neighbor method to compare between these methods based on the accuracy result. Haar method gives the highest accuracy value compared with other two methods.

To the best of our knowledge, no previous work has considered the problem of diagnosing fractures in hand bones. Relevant research papers can be categorized into two categories. The first one includes the papers that consider fractures in long bones [45], [29], [51], [31], [20], [33], [11] It should be noted here that the techniques used by these papers are not directly transferrable to the problem at hand since detecting fractures in long bones is a much simpler problem than detecting fractures in hand bones due to their complex structure and organization. The second category includes the papers that attempt to study and segment the x-ray images of hand bones for the purpose of diagnosing hand diseases such as rheumatic arthritis as well as for bone age estimation [10], [6], [54], [25], [5], [19], [30]. Papers from both categories are discussed below.

In one of the earliest works on bone fracture detection, Tian [45] propose a system for fracture detection in femur bones based on measuring the neck-shaft angle of the femur. In follow-up works [29], [51], [31], the authors propose to use Gabor, Markov Random Field, and gradient intensity features extracted from the x-ray images and fed into Support Vector Machines (SVM) classifiers. They observe that the combination of three SVM classifiers improves the overall accuracy and sensitivity compared to using individual classifiers. To capitalize on this observation, He at al. [20] propose to use a "*hierarchical*" SVM classifier system for fracture detection in femur bones. To use hierarchical classifiers, the classification problem is divided into smaller sub-problems. This is done in the SVM's kernel space instead of the feature space due to the complexity of the problem and the limited dataset. Each sub-problem is handled by an optimized SVM classifier and to ensure that the hierarchical performs well, lower-level SVMs should complement the performance of higher-level SVMs.

Mahendran and Baboo [33] propose a fusion classification technique for automatic detection of existence of fractures in the Tibia bone (one of the long bones of the leg). The authors start with preprocessing steps of contrast adjustment, edge enhancement, noise removal and segmentation before extracting texture features. For the classification step, the authors propose combining the results of three common classifiers, viz., feedforward backpropagation Neural Networks (NN), Support Vector Machines (SVM) and Naive Bayes (NB), using a simple majority vote technique.

Chai et al. [11] propose a Gray Level Cooccurrence Matrix (GLCM) based algorithm to detect the fracture of femur if it exists. The authors start with image preprocessing steps that include binary conversion, fine particles elimination and bone shaft detection. After applying an edge detection technique, the image goes through texture analysis using Gray Level Co-occurrence Matrix (GLCM) to extract features and perform classification.

In another work by the same group [10], another GLCM based method is proposed to segment the x-ray image of the hand and separate the bone regions from the soft tissue regions. After divide the image into several vertical bands and, subsequently, each vertical band into several horizontal bands, K-means clustering is applied followed by GLCM texture analysis. The purpose of this step is to be able to re-construct the image so that separating the bones from the soft tissues becomes easier.

In both [6], [54], the authors propose an automated algorithm to compute the joint width in the x-ray images of the hand. Such a process is essential in age assessment as well as diagnosis of hand diseases (such as rheumatic arthritis) and their prognosis. Their approach perform dilation of the image followed by a filtering step using Gauss function. Then a thinning procedure is used to define the skeleton of the hand and an analysis of the branches is performed to find the correct branches of the fingers. Based on this joints locations are detected and their widths are computed.

The authors in [25] propose a powerful segmentation approach of x-ray hand images using bottom-up region merging method and similarity measures between regions on four levels: local, regional, global, and hierarchical view.

In [19], the authors proposed an automatic segmentation method of in x-ray hand images. They start with detecting the edges of the image, then automatically determining the region of interest and finally segmenting the image to extract the carpal bones only.

Another segmentation method in x-ray hand images has been proposed in [30] for determining skeletal age. This work consists of two processes, the first one is image preprocessing using diffusion filter and the second process is image segmentation based on the region level.

We finally discuss other related works. This work takes a fully autonomous approach to the diagnosis problem. Other works such as [2] take a semi-autonomous approach in which the user's feedback plays an integral role in determining the system's behavior and accuracy. The AdaAgen system of [42] is an example of such systems that considers the problem of long bone fractures.

The above works focus on diagnostics. Other works consider prognostics and study how the condition of a patient will change. In [24], the authors consider femoral neck fracture and the prognostics of patients' recovery.

3. Proposed Method

In this section, the proposed method is discussed in details. Since the first and most elementary component of a supervised learningsystem such as ours is the labeled dataset, we start our discussion with the dataset collection and labeling. After visiting several hospitals in the two major Jordanian cities of Amman and Irbid, only one hospital (King Hussein Medical Center) has provided us with x-ray images.However, due to the small size of the provided dataset, we make use of the x-ray images available on Internet websites such as [1]. This was a tedious process that took several months.

The aim of this work is to propose an ef-ficient system for a quick and accurate diagnosis of hand bone fractures based on theinformation gained from the x-ray images. The general framework of the proposed system is as follows. It starts by taking a set of labeledx-ray hand images that contain normal as well as fractured hands and enhance them by applyingsome filtering algorithms to remove the noise from them. Then, it detects the edgesin each image using edge detection methods.After that, it converts each image into a set of features using tools such the Wavelet andthe Curvelet transforms. The next step is to build the classification algorithms based on the extracted features. Finally, in the testing phase, the performance and ccuracy of the proposedsystem are evaluated. The following sections discuss these steps in details.

3.1 Image Preprocessing

As typical with computer-aided diagnosis systems that depend on medical images, image processing tools for noise removal, image enhancement and feature extraction play a crucialrole in the success of such systems. In this work, the development environment of choice is MATLAB due to the large number (and diversity) of the image processing tools developed under

MATLAB.The proposed system starts with removing the noise from the x-rayimage after converting it from RGB to greyscale.Edge detection techniques are then used.Below, these steps are discussed in details.

3.1.1 Noise Removal

Noise can be defined as unwanted pixels that affect the quality of the image. Noise can be written as:

$$f(x, y) = g(x, y) + \eta(x, y)$$

where $f(x, y)$ is the original image, $g(x, y)$ is the output image and $\eta(x, y)$ is the noise model.

There are different types of noise. Salt and pepper noise is one of the most common types of noise that can be found in x-ray images. This type of noise is generally caused by a failure in capture or transmission that is appearing in the image as light and black dots. The salt and pepper noise is handled by applying a mathematical transformation T on the x-ray image as follow [22]:

$$g(x, y) = T[f(x, y)]$$

Were $f(x, y)$ is the original x-ray image having salt and pepper noise and $g(x, y)$ is the output image after applying T on it. In our work, we chose to use the median filter as T to reduce the salt and pepper noise while preserving the edges and sharpness of the image. The median filter is also used to reduce the noise from the image while preserving the edges and the sharpness of the image. The median filter takes each pixel in the image and checks how different it is from its neighboring pixels. If it is "*too different*," then its value is replaced with the middle value of its surrounding pixels. Figure 3 shows an example of applying noise removal and image smoothing on an *x*-ray hand image.

3.1.2 Edge Detection

Edge detection is an important operation in image processing that reduce the number of pixels and save the structure ofthe image by determining the boundaries of objects in the image. Two general approaches to edge detection that are commonly used are:gradient and Laplacian. Both approaches use the first and second derivative of the image to find edges, espectively. The gradient method looks for the minimum and maximum in the first derivative of the image, and the Laplacian method looks for the zero crossing in the second derivative of the image to find edges [16], [28].

This work uses the Sobel edge detector, which is a member of gradient method family. The Sobel operator is used to find the absolute value of the gradient magnitude in the image. Because the image is of two dimensions, the Sobel operator apply the 2-D gradient measures on the image, and use 3×3 convolution masks on the *x*-axis of the image and another 3×3 convolution mask on the *y*-axis of the image to estimate the gradient on both of them. There are two masks: horizontal which is used to find the first derivative on the *x*-axis, and vertical which is used to find the first derivative on the *y*-axis. These masks are shown in Figure 2.

In this work, we find the first derivative of both axes, *x* and *y*, of the image using the Sobel edge detector. After that, the gradient magnitude of the image can be calculated using the following formula [16]:

$$|G| = \sqrt{Gx^2 + Gy^2}$$

An approximate magnitude can be calculated using [16]:

$$|G| = |Gx| + |Gy|$$

Figure 3 shows an example of the above steps.

−1	0	+1
−2	0	+2
−1	0	+1

Gx

+1	+ 2	+1
0	0	0
−1	− 2	−1

Gy

Figure 2. Sobel Masks [16]

3.2 Feature Extraction and Selection

After smoothing the image and detecting the edges of the hand bone, the proposed system proceeds with extracting useful and discriminating features of the hand bone image. Feature extraction is the main step in various image processing applications. Since this work is considered a first attempt to address the problem at hand (to the best our knowledge), we focused on using the features that are known to work well in medical diagnosis systems based on image analysis [37], [44]. A combination of different sets of features are used such as features from the Wavelet transform [34], features from the Curvelet transform [8] and other textural features [40]. Note that for the Wavelet and the Curvelet transforms to give the best results, edge detection has to be applied first, whereas other textural features might be negatively affected by edge detection. That is why they are extracted immediately after noise removal. Note also that since both the Wavelet and the Curvelet transforms compute a huge number of coefficients for each image, a technique for feature selection has to be applied.More details will be given on this issue in the following section.

3.2.1 Wavelets Features

The Wavelet transform (or simply Wavelets) is an interesting technique that has been developed to solve problems in physics, mathematics and engineering.This method provides the most modern applications in feature extraction,texture categorization, signal analysis,image processing and finds the abnormality in various medical images [41], [26], [27]. When analyzing an image using Wavelets, the result is a set of coefficients (features) for the analyzed image. Figure 4 shows an example of the application of the Wavelet transform on an x-ray image.

3.2.2 Curvelets Features

The Curvelet transform is a multi-scale method derived from the Wavelet transform that is used in mathematical and signal processingand biological applications. It is an interesting method because it provides a mathematical framework that has an adaptive way to represent objects in smooth curve [7], [32].

It should be noted here that applying the Wavelet or the Curvelet transforms on an xray image would generate tens of thousands of coefficients that can be used as features. Such a huge number of features makes the use of several important classifiers difficult due to their poor handling of high dimensional datasets.

Thus, a feature selection technique must be applied on the dataset. For this work, Weka's supervised attribute filter is used to reduce the number of features by three orders of magnitude with the BestFirst technique which uses hillclimbing with backtracking. This technique selected the best 84 features.

3.2.3 GLCM Features

We use the Gray Level Co-occurrence Matrix (GLCM) method to extract additional texturalfeatures. Below we discuss these features in details [40], [13]:

- **Entropy:** Measures the randomness of an image to use in determining the texture features of the image.

$$-\sum_{i}\sum_{j} p(i,j)\log(p(i,j))$$

- **Contrast:** Measures the difference of contrast among all pixels of the image.

$$\sum_{i,j} |i-j|^2 p(i,j)$$

- **Correlation:** Measures how the pixels are correlated to each other.

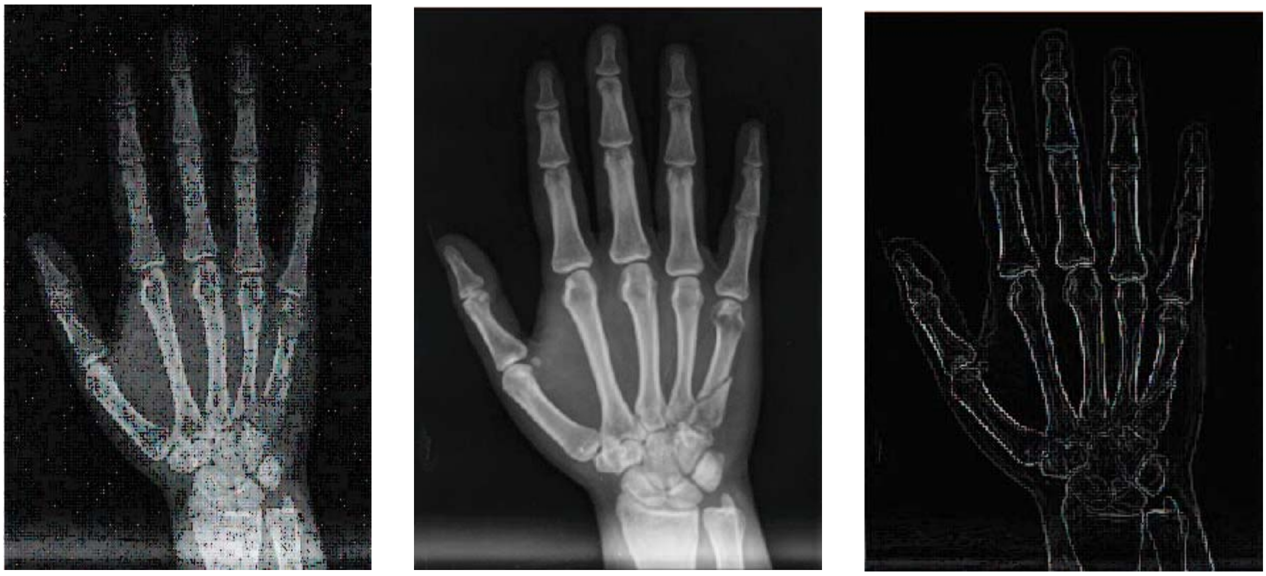
$$\sum_{i,j}\frac{(i-\mu_i)(j-\mu_j)p(i,j)}{\sigma_i\sigma_j}$$

- **Homogeneity:** It is the opposite of contrast feature that measures the closeness among the pixels of the image.

$$\sum_{i,j}\frac{p(i,j)}{1+|i-j|}$$

4. Experimental Results

In this work, we consider the binary classification problem of determining whether a fracture exists in an x-ray image of the hand



(a) Original image (b) The image after noise removal (c) The image after edge detection

Figure 3. Image preprocessing steps

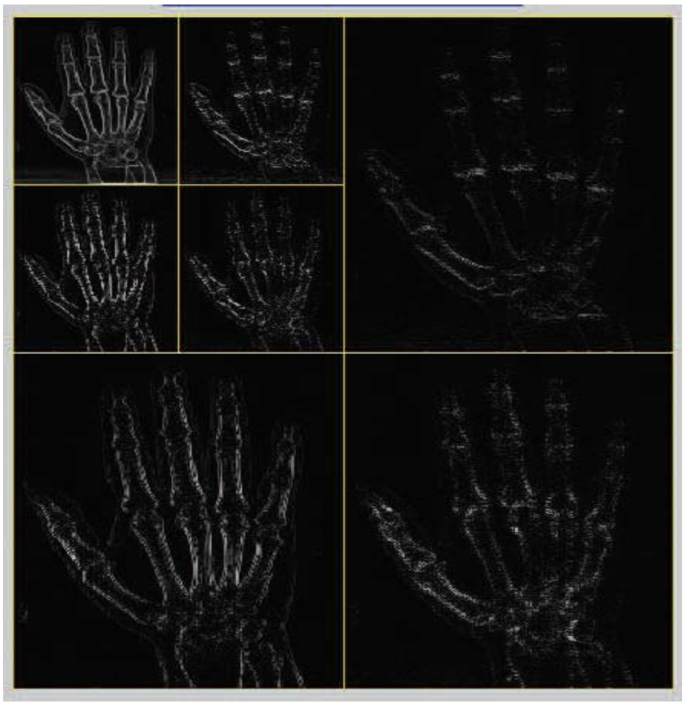


Figure 4. Image analysis using the wavelet transform

or not. The dataset consists of 98 *x*-ray images; half of them are for normal hand bones and the other half contains a fracture in one of the hand's 19 bones.1 Images are collected from Internet websites such as Radiopaedia [1] as well as the Jordanian Royal Medical Services in Amman, Jordan.

Since we have different sets of features, individual experiments must be conducted to determine which one is more suitable. Moreover, an experiment on the combined set of features s is also conducted. These experiments are conducted using Weka [17], one of the most commonly used tools in the machine learning field. Below we discuss the classification algorithms and testing techniques used in these experiments.

4.1 Classification

Classification is a step of data analysis to study a set of data and categorize them into a number of categories. Each category has its own characteristics and the data that belong to such category have the same properties of this category. There are several types of classifiers that are used to classify different sets of data, and also there are different techniques to measure the accuracy of these classifiers. In this section we discuss the classifiers and the techniques we used in this work.

Going through the literature, one can see that there are certain classifiers that work well for problems such as the one at hand, and thus, are more commonly used. We use four of these classifiers in our experiments: Neural Network (NN), Naive Bayes (NB), Bayesian Networks (BN) and Decision Tree (DT).

A *Decision Tree* (*DT*) represents the features of the data as a tree starting by a root and go down to its branches, where each node represents a test on one feature and the following branches of any node are the values associated with this feature. To classify any instance of the data, it should be applied on each node of the tree to test the values of the feature until it reaches the leaf node that represents the class of that instance [38].

The *Naive Bayes* (*NB*) classifier is one of the Bayesian classifiers. NB assumes that a feature value is independent from other features values. The classification of a tuple from the instance works by finding the probability of its conditional dependencies on all the classes, and the maximum probability value of the tuple on a class represents that this tuple belongs to this class. The *Bayesian Network* (*BN*) classifier is another type of Bayesian classifiers. BN differs from NB by finding the probability of the conditional dependencies of features among different subsets of all features [18].

The *Neural Network* (*NN*) classifier simulates the biological nervous systems of the brain. It consists of three layers, namely: input layer, hidden layer, and output layer. The architecture of this classifier is composed of a set of interconnected nodes that are the processing unit, edges that propagate the signal between these neurons, and weights associated with each edge. The neural network learns by adjusting these weights until having an acceptable error rate.

Due to the complexity of the problem at hand, combining a set of "*base classifiers*" into one "*meta-classifier*" to improve the accuracy seems like an appealing option. The most obvious way to combine different classifiers is voting, in which each classifier is applied on a new instance and a majority vote is taken.The voting scheme can be weighted or unweighted.

Unfortunately, voting might not be the best choice for many problems. Two of the most widely-used combination techniques are Bagging and Boosting.

Bootstrap aggregation (or *Bagging*) creates a set of classifiers. To classify a tuple of an instance of the data, each one of the created classifiers predicts the class of the tuple. Then the bagged classifier calculates the most predicted class of this tuple among all classifiers and assigns the tuple to this class [18].

Finally, *Boosting* creates a set of classifiers to classify all tuples of an instance of data and gives a weight for each tuple. To classify a certain tuple, each classifier starts to predict the class of this tuple. If the predicted class of the tuple is incorrect, then increase the weight of this tuple and otherwise decrease the weight of it. Then Boosting gives each classifier a vote weight based on the error rate. The higher voting weight is given to the classifier that has the minimum error rate and this classifier assigned the class of the tuple [18].

4.2 Testing and Evaluation

To give a high level of confidence in the accuracy of the proposed system, the *k*-fold cross validation technique is used. It starts by dividing the data set into *k* folds (subsets) of the same size. It then chooses one of them as a testing set and the remaining as the training set based on which the model is built. This process is repeated for each "*fold* " and the average accuracy is reported [18]. In this work,the 10-fold cross validation is used due to its low variance and bias.

For measuring the performance of our system,we report some of the most commonly used metrics in the literature. Before discussing these metrics, we start by making some definitions. As mentioned before, we consider a binary classification problem in this work. For such problems, there are only four possible outcomes of applying the classifier on any instance. These outcomes are commonly known as [14]:

- *True Positive* (*TP*) which refers to the fractured images that are correctly labeled as fractured.
- *True Negative* (*TN*) which refers to the normal (non-fractured) images that are correctly labeled as normal (nonfractured).
- *False Positive* (*FP*) which refers to the normal (non-fractured) images that are incorrectly labeled as fractured.
- *False Negative* (*FN*) which refers to the fractured images that are incorrectly labeled as normal (non-fractured).

The accuracy measures we use to evaluate the performance of the proposed classifiers are the precision, the recall, the F-measure and the AUC, which is the area under the Receiver Operating Characteristic (ROC) curve. The following equations define the precision, the recall and the F-measure, respectively [50]:

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

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

$$F = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$$

4.3 Results of Base Classifiers

As previously mentioned, we focus on the four most commonly used base classifiers for similar problems of image-based computer-aided diagnosis systems. Specifically, we focus on DT, BN, NB and NN. Moreover, since three sets of features were computed, separate experiments were conducted to evaluate which set is more useful. Tables 1, 2 and 3 show the accuracy measures for each of these classifiers using the Wavelets features, the Curvelets features and the GLCM features, respectively. From these tables, it can be clearly seen that the performance of most classifiers with datasets consisting of the Curvelets features alone or the GLCM features alone is not much better than random guessing. On the other hand, using the Wavelets features produces much better results. Moreover, from Table 1, it can be seen that the best classifier is the Bayesian Network classifier with an accuracy higher than 86%.

Algorithm	Precision	Recall	F-Measure	AUC
BN	86.8%	86.7%	86.7%	93.8%
NB	82.8%	82.7%	82.6%	92.8%
NN	81.7%	81.6%	81.6%	89.4%
DT	63.3%	63.3%	63.3%	63.4%

Table 1. Accuracy measures for the base classifiers using the Wavelets features

Algorithm	Precision	Recall	F-Measure	AUC
DT	46.9%	48%	43%	46.4%
NB	60.3%	59.2%	58.1%	59.7%
BN	47.4%	49%	40%	48.4%
NN	64.7%	60.2%	56.9%	61.5%

Table 2. Accuracy measures for the base classifiers using the Curvelets features

From the above tables, it is obvious that using each set of features individually, the results are far from perfect. To improve them, a natural way is to combine the feature set. Table 4 shows the accuracy measures for the base classifiers using the combined feature set. As with case of using the Wavelets features alone, the best results are obtained though the Bayesian Network classifier. Moreover, the table shows that both the Bayesian Network classifier and the decision tree classifier performed similarly compared to the case of using the Wavelets features alone. On the other hand, the accuracy of the Naive Bayes classifier slightly increased whereas the accuracy of the neural networks classifiers slightly decreased. Since the combined feature set is generating better results, it will be the one used for the remaining experiments.

4.4 Results of Meta-Classifiers

In the previous section, we discuss the performance of base classifiers. One way to improve them is to use meta-classifiers as

Algorithm	Precision	Recall	F-Measure	AUC
BN	47.8%	49%	41%	48.4%
NB	60.6%	59.2%	57.8%	60%
NN	55.6%	55.1%	54.2%	56.6%
DT	46.9%	48%	43%	46.6%

Table 3. Accuracy measures for the base classifiers using the GLCM features

Algorithm	Precision	Recall	F-Measure	AUC
BN	86.8%	86.7%	86.7%	93.4%
NB	84.8%	84.7%	84.7%	92.7%
NN	78.9%	78.6%	78.5%	87.9%
DT	63.3%	63.3%	63.3%	62.1%

Table 4. Accuracy measures for the base classifiers using the combined feature set

discussed earlier this section. Following is a discussion of the results of applying three famous metaclassifiers (voting, bagging a and boosting). Since meta-classifiers can be applied to almost any classifier, there is no reason stopping us from applying a meta-classifier on another metaclassifier. We call this technique *two-level metaclassification*. Incidentally, it turns out that the best results were obtained through a two-level meta-classifier.

4.4.1 Voting

We start with voting, which is perhapsthe simplest meta-classifiers. For a binary classification like ours, voting simply consults an odd number of different classifiers and takes a majority vote between them. Since only four base classifiers are under consideration, it is natural to apply voting on different 3-element subsets of them. Table 5 shows the accuracy measures resulting from applying the majority voting scheme on different sets of classifiers. The best accuracy result is obtained by applying voting on BN, NB and NN classifiers with a value of 88.7%, which is better than the result any of base classifier.

Voting On	Precision	Recall	F-Measure	AUC
BN + NB + NN	88.9%	88.8%	88.8 %	94.9%
BN + NB + NN	86.8%	86.7%	86.7%	91.2%
BN + NB + NN	87.3%	85.7%	85.6%	94%
BN + NB + NN	85.9%	84.7%	84.6%	92.3%

Table 5. Accuracy measures for the voting scheme

Bagging on	Precision	Recall	F-Measure	AUC
BN	86.8%	86.7%	86.7%	94.5%
NB	86%	85.7%	85.7%	95%
NN	50%	50%	45.2 %	52.2%
DT	78.6%	78.6%	78.6%	86.7%

Table 6. Accuracy measures for the bagging scheme

4.4.2 Bagging

The bagging technique can be applied to individual classifiers to improve their performance.It involves a very simple technique of replicating the dataset and building a model for each replicate. These models are used as an ensemble of classifiers to generate a decision on any new instance. Table 6 shows the accuracy measures resulting from applying the bagging scheme on the base classifiers. From this table,it can be seen that the best accuracy result is obtained by the Bayesian Network classifier is the same with and without applying bagging. On the other hand, bagging has a slight positive effect on the Naive Bayes classifier, a notable positive effect on the decision tree classifier and a significant negative effect on the neuralnetworks classifier. The latter observation might be due to the small sizes of the replicated datasets, which lead to poor performance by each of the neural network models created.

Algorithm	Precision	Recall	F-Measure	AUC
BN	86.9 %	86.7 %	86.7%	96%
NB	84.7%	84.7%	84.7%	91.8%
NN	78.9%	78.6%	78.5%	87.6%
DT	76.6%	76.5%	76.5%	76%

Table 7. Accuracy measures for the boosting scheme

4.4.3 Boosting

The last meta classifiers we consider is the adaptive boosting algorithm. Table 7 shows the accuracy measures resulting from applying the boosting scheme on the base classifiers. From this table, it can be seen that the best accuracy result is obtained by the Bayesian Network classifier has slightly improved with the use of boosting. Moreover, boosting has no effect on the neural networks classifier, a slight negative effect on the Naive Bayes classifier and a notable positive effect on the decision tree classifier (but less positive than the effect of bagging).

Voting On	Precision	Recall	F-Measure	AUC
BN + NB + NN	83.7%	83.7%	83.7 %	90.7%
BN + NB + DT	91%	90.8%	90.8%	95.8%
BN + NN + DT	75%	62.2%	56.7%	84.4%
NB + NN + DT	76.8%	76.5%	76.5%	89%

Table 8. Accuracy measures for the boosting then voting scheme

4.4.4 Two-level Meta-Classifiers: Boosting and Voting

Since both voting and boosting generate promising results, combining them may produce even better results. This is achieved by applying boosting on each of the four individual classifiers we consider and then apply voting on them. Table 8 shows the accuracy measures resulting from applying this approach on different sets of classifiers. The results are satisfying as we finally manage to cross the 90% accuracy barrier. However, they are still not good enough. In the following experiment, we try other combinations of two-level metaclassifiers.

Algorithm	Precision	Recall	F-Measure	AUC
BN	90.8 %	90.8 %	90.8%	94.5%
NB	87.8%	87.8%	87.8%	94.4%
NN	53.4%	53.1%	51.8%	51.6%
DT	81.8%	81.6%	81.6%	90.3%

Table 9. Accuracy measures for the bagging then boosting scheme

Algorithm	Precision	Recall	F-Measure	AUC
BN	91.8 %	91.8 %	91.8%	97.1%
NB	86.8%	87.8%	86.7%	88.6%
NN	53.7%	52%	45.9%	54.6%
DT	77.6%	77.6%	77.5%	87.5%

Table 10. Accuracy measures for the boosting then bagging scheme

4.4.5 Two-level Meta-Classifiers: Bagging and Boosting

It is common to hear the phrase "*you may apply bagging and boosting to improve the performance of your classifier*" being thrown around in machine learning seminars and workshops. We followed this sentence literally. However, we were not sure which technique should be applied first. So, we report the results for both options. Tables 9 and 10 show the accuracy measures for the two possible ways of combining bagging and boosting. From these tables, it can be seen that the Bayesian Network classifier has a 90+% accuracy in both cases with the best being of 91.8% when applying boosting followed by bagging.

5. Conclusion and Future Work

This work address the problem of detecting hand bone fractures from x-ray images. Several techniques are tested for the image preprocessing phase. Moreover, different sets of features are computed and tested. Finally, base as well as multi-level meta-classifiers are tested. An accuracy level of 91.8% is obtained by applying boosting and then bagging on the Bayesian Network classifiers were the feature set include features computed using Wavelets, Curvelets and GLCM. For a first attempt at this problem, such results are extremely encouraging. Nonetheless, there are still room for improvement such as experimenting with larger and more diverse dataset and different feature sets.

References

[1] Radiopaedia. (2013). http://radiopaedia.org/ [Online; accessed February-2013].

[2] Agapie, E. (2008). Second opinion, a collaborative online game for medical diagnosis. Technical report, University of California, Berkeley.

[3] Al-Khaffaf, H., Talib, A. Z., Salam, R. A. (2008). Removing salt-and-pepper noise from binary images of engineering drawings. *In*: *Pattern Recognition*. ICPR. 19th International Conference on, p. 1–4. IEEE.

[4] American Cancer Society. (2013). Imaging (radiology) tests, http://www.cancer.org/acs/groups/cid/documents/webcontent/003177-pdf.pdf [Online; accessed June-2013].

[5] Bandyopadhyay, O., Chanda, B., Bhattacharya, B. B. (2011). Entropy-based automatic segmentation of bones in digital x-ray images. *In*: *Pattern Recognition and Machine Intelligence*, p. 122–129. Springer.

[6] Bielecki, A., Korkosz, M., Zieliński, B. Hand radiographs preprocessing, image representation in the finger regions and joint space width measurements for image interpretation. *Pattern Recognition*, 41(12) 3786–3798, 2008.

[7] Candès, E. J. (2003). What is... a curvelet? *Notices of the American Mathematical Society*, 50 (11) 1402–1403.

[8] Candès, E. J., Donoho, D. L. (2000). Curvelets: A surprisingly effective nonadaptive representation for objects with edges. Technical report, DTIC Document.

[9] Canny, J. (1986). A computational approach to edge detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-8 (6) 679–698.

[10] Chai, H. Y., Wee, L. K., Swee, T. T., Hussain, S. (2011). Glcm based adaptive crossed reconstructed (acr) k-mean clustering hand bone segmentation. *Book GLCM based adaptive crossed reconstructed (ACR) k-mean clustering hand bone segmentation*, p. 192–197.

[11] Chai, H. Y., Wee, L. K., Swee, T. T., Hussain, S. (2011). Gray-level co-occurrence matrix bone fracture detection. *WSEAS TRANSACTIONS on SYSTEMS*, 10 (1).

[12] Chan, K.-P., Fu, A. W.-C. (1999). Efficient time series matching by wavelets. *Data Engineering*. *In*: Proceedings, 15*th* International Conference on, p. 126–133. IEEE.

[13] Clausi, D. A. (2002). An analysis of co-occurrence texture statistics as a function of grey level quantization. *Canadian Journal of remote sensing*, 28 (1) 45–62.

[14] Davis, J., Goadrich, M. (2006). The relationship between precision-recall and roc curves. *In*: Proceedings of the 23*rd* International Conference on Machine Learning, p. 233–240. ACM.

[15] Gong, T., Liu, R., Tan, C. L., Farzad, N., Lee, C. K., Pang, B. C., Tian, Q., Tang, S., Zhang, Z. (2007). Classification of ct brain images of head trauma. In *Pattern Recognition in Bioinformatics*, p. 401–408. Springer.

[16] Green, B. Edge detection tutorial. http://dasl.mem.drexel.edu/alumni/bGreen/www. pages.drexel.edu/ weg22/edge.html [Online; accessed [June-2013].

[17] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I. H. (2009). The weka data mining software: an update. *ACM SIGKDD Explorations Newsletter*, 11 (1) 10–18.

[18] Han, J., Kamber, M., Pei, J. (2011). *Data Mining: Concepts and Techniques*. Morgan Kaufmann.

[19] Hao, S., Han, Y., Zhang, J., Ji, Z. (2013). Automatic isolation of carpal-bone in hand *x*-ray medical image. In *Informatics and Management Science I*, p. 657–662. Springer.

[20] He, J. C., Leow, W. K., Howe, T. S. (2007). Hierarchical classifiers for detection of fractures in x-ray images. In *Computer Analysis of Images and Patterns*, p. 962–969. Springer.

[21] Huang, H. (2000). Pacs: basic principles and applications. *European Radiology*, 10 (2) 303–303.

[22] Jassim, F. A. (2003). Kriging interpolation filter to reduce high density salt and pepper noise. *World of Computer Science and Information Technology*, 3 (1).

[23] Jezierska, A., Chaux, C., Pesquet, J.-C., Talbot, H., Engler, G. (2011). An EM approach for poisson-gaussian noise modeling. *In*: *European Signal Processing Conference*, p. 2244–2248.

[24] Kukar, M., Kononenko, I., Silvester, T. (1996). Machine learningin prognosis of the femoral neck fracture recovery. *Artificial Intelligence in Medicine*, 8 (5) 431–451.

[25] Lehmann, T. M., Beier, D., Thies, C., Seidl, T. (2005). Segmentation of medical images combining local, regional, global, and hierarchical distances into a bottom-up region merging scheme. *In*: *Proc. of SPIE,* 5747, p. 547.

[26] Li, J., Gray, R. M. (2000). Context-based multiscale classification of document images using wavelet coefficient distributions. *Image Processing, IEEE Transactions on*, 9 (9) 1604–1616.

[27] Li, T., Li, Q., Zhu, S., Ogihara, M. (2002). A survey on wavelet applications in data mining. *ACM SIGKDD Explorations Newsletter*, 4 (2) 49–68.

[28] Lim, J. S. (1989). *Two-Dimensional Signal and Image Processing*. Prentice Hall PTR.

[29] Lim, S. E., Xing, Y., Chen, Y., Leow, W. K., Howe, T. S., Png, M. A. (2004). Detection of femur and radius fractures in xray images. *In*: Proc. 2*nd* Int. Conf. on Advances in Medical Signal and Info. Proc.

[30] Lin, P., Zheng, C., Zhang, F., Yang, Y. (2005). X-ray carpal-bone image boundary feature analysis using region statistical feature based level set method for skeletal age assessment application. *Optica Applicata*, 35 (2) 283.

[31] Lum, V. L. F., Leow, W. K., Chen, Y., Howe, T. S., Png, M. A. (2005). Combining classifiers for bone fracture detection in x-ray iimages. *In*: *Image Processing*. ICIP. IEEE International Conference on, 1, p. I–1149. IEEE.

[32] Ma, J., Plonka, G. (2010). The curvelet transform. *Signal Processing Magazine, IEEE*, 27 (2) 118–133.

[33] Mahendran, S., Baboo, S. S. (2011). An enhanced tibia fracture detection tool using image processing and classification fusion techniques in X-ray images. *Global Journal of Computer Science and Technology (GJCST)*, 11 (14) 23–28.

[34] Mallat, S. G. (1989). A theory for multiresolution signal decomposition: the wavelet representation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 11 (7) 674–6939.

[35] MedicineNet. Bone fracture. (2011). http://www. medicinenet.com/fracture/article.htm [Online; accessed November-2011].

[36] Mena, L., Gonzalez, J. (2006). Machine learning for imbalanced datasets: Application in medical diagnostic. *In*: Proceedings of the 19th International FLAIRS Conference.

[37] Misiti, M., Misiti, Y., Oppenheim, G., Poggi, J.-M. (2007). *Wavelets and their Applications*. Wiley Online Library.

[38] Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill.

[39] Pham, D. L., Xu, C., Prince, J. L. (2000). Current methods in medical image segmentation. *Annual review of biomedical engineering*, 2 (1) 315–337.

[40] Rad, A. E., Rahim, M. S. M., Kumoi, R., Norouzi, A. (2013). Dental x-ray image segmentation and multiple feature extraction. *AWERProcedia Information Technology and Computer Science*, p. 2.

[41] Sifuzzaman, M., Islam, M., Ali, M. (2009). Application of wavelet transform and its advantages compared to fourier transform. *Journal of Physical Sciences*, 13, p. 121–134.

[42] Syiam, M., El-Aziem, M. A., El-Menshawy, M. (2004). Adagen: Adaptive interface agent for x-ray fracture detection. *International Journal of Computing & Information Sciences*, 2 (3).

[43] Tanwani, A. K., Afridi, J., Shafiq, M. Z., Farooq, M. (2009). Guidelines to select machine learning scheme for classi- fication of biomedical datasets. *Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics*, p.128–139.

[44] Tian, D.-Z., Ha, M.-H. (2004). Applications of wavelet transform in medical image processing. *Machine Learning and Cybernetics*. *In*: Proceedings of 2004 International Conference on, 3, p. 1816–1821.

[45] Tian, T. (2002). Detection of femur fractures in x-ray images. Master's thesis, National University of Singapore, Singapore.

[46] Tsai, W.-S. (2008). Contourlet transforms for feature detection.

[47] Vijaykumar, V., Vanathi, P., Kanagasabapathy, P. (2010). Fast and efficient algorithm to remove gaussian noise in digital images. *IAENG International Journal of Computer Science*, 37 (1).

[48] Wikipedia. (2013). Bone fracture-wikipedia, the free encyclopedia. http://en.wikipedia.org/w/index.php? title=Bone fracture&oldid=544914255 [Online: accessed January-2013].

[49] Wikipedia. (2013). Hand — wikipedia, the free encyclopedia.

[50] Witten, I. H., Frank, E. (2005). *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann.

[51] Yap, D. W.-H., Chen, Y., Leow, W. K., Howe, T. S., Png, M. A. (2004). Detecting femur fractures by texture analysis of trabeculae. In *Pattern Recognition*. *ICPR*. *In*: Proceedings of the 17th International Conference on, 3, p. 730–733. IEEE.

[52] Zain, M. L. M., Elamvazuthi, I., Begam, M. (2009). Enhancement of bone fracture image using filtering techniques. *The International Journal of Video and Image Processing and Network Security*, 9 (10).

[53] Zhang, L., Zhang, L., Zhang, D. (2010). A multi-scale bilateral structure tensor based corner detector. *Computer Vision–ACCV*, p. 618–627.

[54] Zielinski, B. A fully-automated algorithm dedicated to computing metacarpophalangeal and interphalangeal joint cavity widths. *Schedae Informaticae*, 16, 47–67.