"Bone age Prediction" Deep CNN models for predicting bone age from hand radiographs

Bahador Mirzazadeh†, Mohammad Matin Parvanian‡

***Abstract*—Bone age prediction from hand radiographs is a task that involves using medical imaging to determine the maturity of a patient's bones. This information is useful for various medical applications, including growth assessment, diagnosis of endocrine disorders, and monitoring treatment efficacy. Deep Convolutional Neural Networks (CNNs) have proven to be a powerful tool for image classification, image regression, machine vision, and feature extraction tasks. They have been applied successfully to a variety of medical imaging tasks, including bone age prediction. This paper explains how we built deep convolutional neural networks to predict bone age in months using hand X-ray image data as input. In order to extract functional results, we used Inception V4, which is a deep convolutional neural network (CNN) architecture for image classification and regression tasks. It was introduced in 2014 by Google researchers and published in 2016. Another approach that is used is ResNet-50, a deep convolutional neural network (CNN) architecture for image classification and machine vision tasks. It was developed by Microsoft researchers and published in 2015. Although both of these approaches have different structures and architectures, they both produce reasonable results.**

***Index Terms*—Supervised Learning, Convolutional Neural Networks, Image Regression, ResNet-50, Inception V4.**

I. Introduction

Bone age prediction from hand radiographs is a significant application of human data analysis in the field of medical imaging. This method utilizes image analysis techniques to determine an individual's chronological age based on the maturity of their hand and wrist bones as seen on a hand radiograph. The process involves comparing the hand radiographs of an individual with a reference database of hand radiographs from individuals of known chronological ages. In recent years, there has been an increasing interest in developing automatic methods for bone age prediction from hand radiographs using machine learning and computer vision techniques. These methods involve training a model on a large dataset of hand radiographs and their corresponding chronological ages, then using the trained model to predict the chronological age of new hand radiographs.

To build models that can predict bone age we use can use CNN models but One of the challenges in developing these methods is that there can be significant inter- and intra-observer variability in manual bone age assessment, and that hand radiographs of individuals with the same chronological age can show significant variation in bone maturity. To overcome these challenges, Deep CNNs models are often trained using a large dataset of hand radiographs and corresponding chronological ages and are designed to account for variability in bone maturity and appearance. Moreover, bone age prediction from hand radiographs is used to determine the maturity of an individual's skeletal system. This information can be useful in a variety of medical contexts. Some common uses include: [1] Normal growth evaluation. Bone age prediction is commonly used to monitor the growth and development of children and adolescents. This information can help determine if a child is growing at a normal rate and if any growth delays or abnormalities are present. In the cited paper authors analyzed 730 sets of radiographs (cephalogram and hand-wrist) of untreated subjects (352 boys, 378 girls; age range, 6-18 years) from a growth study, each sex as a separate sample and used image regression to generate a calculated bone age for All correlating variables.[2] Endocrine disorders .Bone age prediction can be used to diagnose and monitor endocrine disorders such as growth hormone deficiency, thyroid disorders, and others that affect bone development. It is particularly helpful in the clinical workup of children with growth and/or puberty disorders as well as in treating decisions, such as whether to start replacement therapy in a patient with hypogonadism. On the other hand, it is important to recognize that overemphasizing bone age evaluation can be misleading if not used in the proper settings. In this article heights and near-adult heights were measured in 82 patients (48 females) with chronic endocrinopathies at the age of 10.45 ± 2.12 y and at the time of transition to adult care (17.98 ± 3.02 y).[3] Another important use of bone age prediction is Pubertal timing evaluation. Bone age prediction can be used to assess the timing of pubertal onset and progression, which can be important in diagnosing and treating conditions such as precocious puberty. This approach examines the role of skeletal maturity ('bone age', BA) assessment in clinical practice. The authors used multi-linear regression to predict How long will the growth of excessive height in children or abnormal shortness in them continue With the help of the order, the treatment steps can be started.[4] Genetic disorders. Bone age prediction can be used to diagnose and monitor certain genetic disorders that affect bone growth and development, such as Down syndrome, Turner syndrome, and others. The authors provide an overview of the role of bone age assessment in genetic disorders, including its use in the diagnosis, prognosis, and management of these conditions.The article covers a range

† bahador.mirzazdeh@studenti.unipd.it
‡ mohammadmatin.parvanian@studenti.unipd.it

of genetic disorders that can impact bone development and growth, including chromosomal abnormalities, single gene disorders, and genetic syndromes. The authors discuss the implications of these disorders for bone age assessment and provide recommendations for clinical practice.From the papers cited above the reader can notice how CNN models well work even if they demonstrate that very complex models are needed to find good regressors for this task.The purpose of this paper is to study not only the most appropriate regression models that permit to fit of each sample to its target label correctly but also to find models with the highest accuracy. In synthesis, this paper wants to analyze:

- Different data preprocessing such as data augmentation and image resizing in order to find the best way to represent the most important information needed for the regression.
- Different structures of neural networks in order to find the best model structure able to catch the hidden information and predict bone age with high precision.
- The comparison of the results obtained with the same features and models for the test set labels.
- The comparison between the prediction using different deep convolutional neural network (CNN) architectures such as ResNet-50 and Inception V4.

For these purposes, this report is structured as follows: In Section II the state of the art is described; in Section III the main work pipeline is presented, while in Section IV the data used and the features of the dataset are argued. In Section V the compared models are described. The performance evaluation of the models and processing techniques are proposed in Section VI, while conclusions are reported in Section VII.

II. Related Work

Deep Convolutional Neural Networks (DCNNs) have been widely used in medical image analysis, including the task of predicting bone age from hand radiographs. Here are some of the most significant contributions and advances in this field

Growth and development assessment: One of the first successful applications of DCNNs for bone age prediction was reported by Rajpurkar et al. (2017) In the study[5], the authors collected a large dataset of hand radiographs along with their corresponding bone ages, and used this dataset to train a deep CNN to predict bone age from radiographs. The architecture of the CNN consisted of multiple convolutional and pooling layers, as well as fully connected layers, and the network was trained using a large number of hand radiographs and their corresponding bone ages. The authors used data augmentation techniques, such as rotation and scaling, to artificially increase the size of their training dataset and prevent overfitting. They also used transfer learning, where they pre-trained the network on a large dataset of images and then fine-tuned it on the smaller hand radiograph dataset. The authors evaluated their system on a test dataset of radiographs and compared their results to those of two expert radiologists. They found that their system was able to achieve an accuracy comparable to that of the expert radiologists and that it was able to produce predictions that were highly correlated with the ground truth bone ages. In summary, the authors Rajpurkar et al. (2017) used a deep CNN to predict bone age from hand radiographs, and used data augmentation and transfer learning techniques to improve the performance of their system. The results showed that their system was able to achieve an accuracy comparable to that of expert radiologists.

Transfer learning: To overcome the limitations of small datasets, several studies have applied transfer learning to fine-tune pre-trained DCNNs on the bone age prediction task. [6]The authors used a pre-trained CNN model as a feature extractor and fine-tuned it on a small dataset of hand radiographs and their corresponding bone ages. They found that fine-tuning the pre-trained model on their smaller dataset allowed them to achieve good performance even with limited amounts of training data. The article by Loeff et al. (2019) explored the use of data augmentation techniques, such as rotation and scaling, to artificially increase the size of their training dataset and prevent overfitting. They found that data augmentation improved the performance of their system, especially when used in combination with transfer learning. The authors evaluated their system on a test dataset of radiographs and compared their results to those of expert radiologists. They found that their system was able to achieve an accuracy comparable to that of the expert radiologists and that it was able to produce predictions that were highly correlated with the ground truth bone ages.

Multi-modal approaches: To improve the accuracy of bone age prediction, some studies have explored multi-modal approaches that consider both hand radiographs and other imaging modalities such as MRI and CT. For example, in the work "Multi-Modal Bone Age Assessment using Deep Convolutional Neural Networks," [7]the authors used a DCNN to analyze both hand radiographs and MRI images to improve the accuracy of bone age prediction. Wang et al. (2021) proposed a multi-modal bone age assessment system using deep convolutional neural networks (CNNs) in the paper "Multi-, Modal Bone Age Assessment using Deep Convolutional Neural Networks."In their work, the authors used both hand radiographs and wrist radiographs to predict bone age and investigated the use of multi-modal information for improved performance. They trained separate CNNs for each modality and then combined the predictions from the two networks to produce a final bone age estimate. The authors used data augmentation techniques, such as rotation and scaling, to artificially increase the size of their training dataset and prevent overfitting. They also used transfer learning, where they pre-trained the network on a large dataset of images and then fine-tuned it on the smaller hand and wrist radiograph datasets. The authors evaluated their system on a test dataset of radiographs and compared their results to those of expert radiologists. They found that their multi-modal system outperformed single-modality systems and was able to achieve higher accuracy compared to the expert radiologists.

Attention mechanisms: Recently, attention mechanisms have been applied to DCNNs for bone age prediction to automatically focus on the most informative regions of the hand radiographs. For example, in the work "Attention-Based Deep Convolutional Neural Network for Bone Age Assess-

ment,"[8] the authors used an attention mechanism to highlight the regions of the hand radiographs that are most relevant to predicting bone age. Overall, DCNNs have shown promising results for predicting bone age from hand radiographs and have led to significant advances in this field. However, there is still room for improvement, and ongoing research is exploring new ways to further improve the accuracy and robustness of these models.

III. Processing Pipeline

The dataset consists of three subsets including training, validation, and test set. In the training set, there are 12611 observations from bone x-ray images which the main resolution is 512 × 512 with Three channels. Besides, the validation set contains 1425 with the same structure. Finally, in the test set, there exist 200 images with the same structure mentioned above. Each of the mentioned images has a continuous target value that demonstrates the bone age of the image owner in months. For performing this regression task we used two different tasks ResNet-50 and Inception V4.

ResNet-50: ResNet-50 is a deep neural network architecture that is a part of the ResNet (Residual Network) family of models. The key idea behind ResNet is the use of residual connections, which allow the network to learn residual functions, or the difference between the desired output and the current representation, rather than learning the underlying mapping from input to output directly. This makes it easier for the network to train and reduces the risk of overfitting.

Inception V4: Inception V4 is a deep convolutional neural network architecture for image classification and other computer vision tasks. The key idea behind Inception V4 is the use of Inception modules, which are building blocks that extract features from the input in parallel using multiple convolutional and pooling layers. The Inception modules have multiple branches, each of which operates on the input using a different kernel size and a number of filters, allowing the network to capture both local and global features. The overall pipeline of the project is shown in Fig.1.

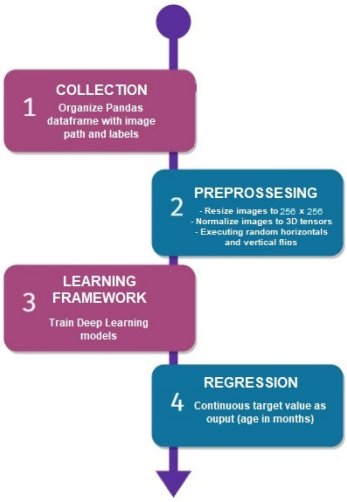

Fig. 1. Processing pipeline

The processing pipeline is summarized by the diagram Fig.1. The are 4 main steps: data collection of bone x-ray images such that we create a pandas dataframe with two columns which the first one is the absolute path of the file and the second one is the corresponding label; parsing the data into 3D tensors, resizing images to 256 × 256, and random horizontal and vertical flips augmentation technique; training with learning framework of choice; predicting bone age in the month as output as a regression task.

IV. Images and Features

The hand X-ray images were obtained from two hospitals in the USA. This dataset contains Three sub-folders which are divided into a training set with 12611 images, a validation set with 1425 images, and a test set with 200 images. The width and height of the mentioned image are not equal in the original dataset. Due to this issue, we resize the width and height of each image to 256 x 256 in the preprocessing phase. Also, rescaling is considered (divide the value of each pixel by 255). Data augmentation is a technique used in machine learning to increase the size and diversity of a training dataset. One common type of data augmentation is flipping images horizontally and vertically. A horizontal flip involves flipping an image left to right so that the image is mirrored along a vertical axis. A vertical flip involves flipping an image top-to-bottom so that the image is mirrored along a horizontal axis. This helps to prevent overfitting, which occurs when a model becomes too specialized to the training data and does not generalize well to new, unseen data.

Also, our labels demonstrate the age of each image in months, and for training and validation set is a discrete value and for the test, the set is a continuous value.

Fig. 2. A sample from the dataset in which the bone age is 120 months

V. Learning Framework

ResNet-50 architecture is composed of different layers which have 2 main layers:

- **Identity Block :** An identity block is a residual block where the input to the block is simply added to its output, bypassing the layers within the block. It contains four sections. The first section contains One convolutional layer, one BachNormalization layer, and a GELU activation layer. The second one consists of One convolutional layer, one BachNormalization layer, and a GELU activation layer. The third one has One convolutional layer and one BachNormalization layer. Finally, the last one

contains the output of the previous layer plus the input of the first section and the GELU activation layer.

- **Convolutional Block :** A convolutional block, on the other hand, is used to reduce the spatial resolution of the feature map, allowing the network to learn more abstract representations. It contains four sections. The first section contains One convolutional layer, one Bach-Normalization layer, and a GELU activation layer. The second one consists of One convolutional layer, one BachNormalization layer, and a GELU activation layer. The third one has One convolutional layer and one BachNormalization layer. Finally, the last one contains the output of the previous layer plus a convolutional layer, batch normalization layer, and the GELU activation layer.

The diagram in Fig.3 provides a visual reference of the model, which can be summarized as follows:

- **Input layer :** The input is a 3D tensor of shape 256×256×3.
- **Zero Padding :** pads the input with a pad of (3,3).
- **Stage 1 :** The 2D Convolution has 64 filters of shape (7,7) and uses a stride of (2,2). BatchNormalization is applied to the channels axis of the input. MaxPooling uses a (3,3) window and a (2,2) stride.
- **Stage 2 :** The convolutional block uses Three sets of filters of size [64,64,256], kernel size is (3,3), and stride is (1,1). The 2 identity blocks use Three sets of filters of size [64,64,256], and kernel size is (3,3).
- **Stage 3 :** The convolutional block uses Three sets of filters of size [128,128,512], kernel size is (3,3), and stride is (2,2). The 3 identity blocks use Three sets of filters of size [128,128,512], and kernel size is (3,3).
- **Stage 4 :** The convolutional block uses Three sets of filters of size [256, 256, 1024], kernel size is (3,3), and the stride is (2,2). The 5 identity blocks use three sets of filters of size [256, 256, 1024], and kernel size is (3,3).
- **Stage 5 :** The convolutional block uses three sets of filters of size [512, 512, 2048], kernel size is (3,3), and stride is (2,2). The 2 identity blocks use three sets of filters of size [512, 512, 2048], and kernel size is (3,3).
- **2D Average Pooling :** Uses a window of shape (2,2) to extract the best features.
- **Flattening :** This flattening layer is a necessary step to correctly format the input for the following layers.
- **Fully Connected Layers :** Reduces its input with 4 Dense layers of shape 256, 128, 64, and 32 respectively, and 1 output layer with a linear activation function.

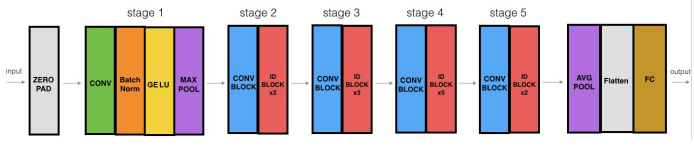

Fig. 3. ResNet-50 Architecture

Inception V4 architecture is another approach used in this article. The diagram in Fig.4 provides a visual reference of the model, which can be summarized as follows:

- **Input layer :** The input is a 3D tensor of shape 256×256×3.
- **Stem Block :** The stem block serves as an initial feature extractor and includes several layers of convolutions, max pooling, and batch normalization.
- **Inception-A Block :** The Inception-A block consists of several parallel convolutional layers with different kernel sizes and a 1x1 convolutional layer for dimensionality reduction. These layers are concatenated and used as the output of the block.
- **Reduction A :** The reduction A block is used to reduce the spatial dimensions of the feature map while increasing the number of channels. It includes max pooling layers and convolutional layers with smaller kernel sizes.
- **Inception-B Block :** The Inception-B block is similar to Inception-A but includes additional 7x7 convolutional layers with stride 2 for down-sampling.
- **Reduction B :** This is exactly similar to the Reduction A block but with some differences in filter size.
- **Inception-C Block :** The Inception-C block uses a combination of convolutional layers with different kernel sizes and factorized convolutions. It also includes a max pooling layer to reduce the feature map dimensions.
- **GlobalAverage Pooling :** Global average pooling is a technique used in the Inception-v4 architecture to reduce the spatial dimensions of the feature map and produce a fixed-sized output tensor. It is applied after the last Inception-C block and before the final fully connected layer.
- **Flattening :** This flattening layer is a necessary step to correctly format the input for the following layers.
- **Dropout :** Dropout is a regularization technique that helps prevent overfitting in neural networks. It works by randomly dropping out some of the units in the network during training. This means that some of the neurons in the network are randomly ignored and do not contribute to the output of the network for a given input.
- **Fully Connected Layers :** Reduces its input with 1 Dense layer of shape 1 with a linear activation function to predict the target value.

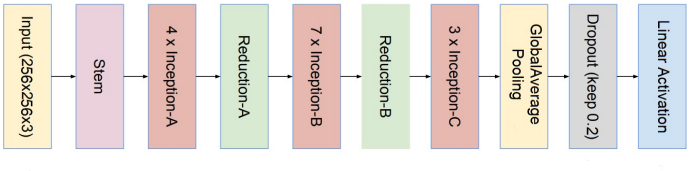

Fig. 4. Inception V4 Architecture

Here we need to clarify some tips:

1. Activation Function of both the approaches for all the hidden layers is GELU. But the question is that why GELU? One of the main advantages of GELU over ReLU is that it has a non-zero mean, which helps prevent the vanishing gradient problem. In ReLU, the negative inputs are set to zero, which can lead to a zero mean activation for a large fraction of the neurons in the network. This can cause the gradients to become very small or vanish, which makes it difficult to

train deep networks. Due to the mentioned reasons, GELU can help prevent the vanishing gradient problem and improve the training of deep networks.

2. Loss Function of both approaches is Mean Squared Logarithmic Error. The MSLE loss function is a variation of the Mean Squared Error (MSE) loss function that is often used in regression tasks. It is especially useful when the target values span several orders of magnitude, as it can help to reduce the impact of outliers and improve the overall performance of the model.

3. The Optimizer of both approaches is Adam. The main advantage of Adam over other optimization algorithms is that it provides a good balance between the speed of convergence and the stability of the optimization. It uses adaptive learning rates for each parameter in the network, which helps to overcome some of the limitations of other optimization algorithms, such as the need for manual tuning of the learning rate and the sensitivity to different types of data and architectures.

VI. Results

In this section, the results of the trained models with the features proposed are presented in order to find the best model for this regression task. During the training phase, we used a training set for training our deep neural networks and we used the validation set due to optimize the hyperparameters of the networks. For the evaluation phase, we used the testset which is entirely unknown and unseen for the networks. According to the evaluation of our approaches, we considered Three different metrics which are popular in the regression task. The first one is RMSE which is a commonly used measurement of the difference between predicted and actual values for a set of numerical data. It is defined as the square root of the mean of the squared differences between the predicted and actual values. One of the advantages of RMSE is that its result is in the same unit as the dependent variables.

The formula for RMSE can be expressed as:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(predicted_i - actual_i)^2}$$

Where "n" is the number of observations, $predicted_i$ is the predicted value for the i-th observation, and $actual_i$ is the actual value for the i-th observation.

The second one is MAPE which is a common measurement of the difference between predicted and actual values, expressed as a percentage. It is defined as the mean of the absolute percentage differences between the predicted and actual values. Despite RMSE, MAPE is independent of the scale of the target variable, which makes it suitable for comparing the performance of models on datasets with different units of measurement.

The formula for MAPE can be expressed as:

$$MAPE = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{actual_i - predicted_i}{actual_i}\right|$$

Where "n" is the number of observations, $predicted_i$ is the predicted value for the i-th observation, and $actual_i$ is the actual value for the i-th observation.

The Third one is R Squared which is a statistical measure that represents the proportion of the variance in the dependent variable that is explained by the independent variables in a regression model. R-squared is a value between 0 and 1, where a higher R-squared value indicates a better fit of the model to the data. An R-squared value of 0 means that the model explains none of the variability of the response data around its mean, while an R-squared value of 1 means that the model explains all of the variability of the response data around its mean. In practice, a high R-squared value is desirable, as it means that a large proportion of the variation in the response variable is explained by the independent variables in the model.

The formula for R-squared can be expressed as:

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}$$

Where SS_{res} is the residual sum of squares (the sum of the squared differences between the observed and predicted values), and SS_{tot} is the total sum of squares (the sum of the squared differences between the observed values and the mean of the observed values).

Also, the value of the loss function is considered to evaluate the performance of the networks.

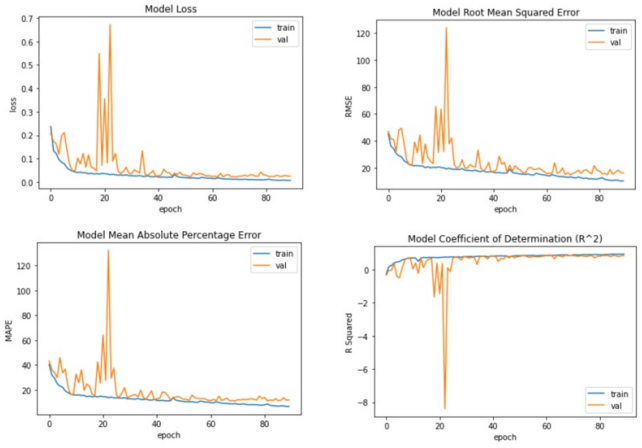

Fig. 5. Evaluation metrics figures of ResNet-50

The metrics and loss curves for ResNet-50 are shown in Fig.5. Based on Fig. 5. 120 epochs are considered and our model stopped the training at the 90th epoch because of the early stopping technique that we used in our implementation. It is obvious that the training curves for loss and metrics smoothly decreased and in the final epochs are completely stable. But the situation is a little bit different for the validation curves. In the early epochs, the validation curves are fluctuating highly and the more model goes further, the curves become more stable. In fact, in the last epochs, the validation curves are as stable as the training curves.

The metrics and loss curves for Inception V4 are shown in Fig.6. Based on Fig. 6. 100 epochs are considered and no early stopping happened during the training phase. It is obvious that the training curves for loss and metrics smoothly decreased and in comparison with ResNet-50, inception-V4 started with higher loss and metrics errors in the early epochs. But in the very first beginning steps the errors decreased significantly and the more model goes further, the curves become stable. At the beginning of the validation loss and metrics curves have high fluctuations and the more model goes further, the curves become stable. In fact, in the last epochs, the validation curves are as stable as the training curves.

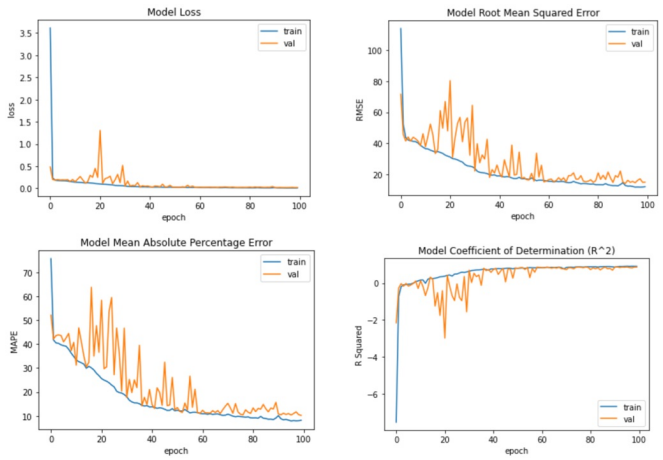

Fig. 6. Evaluation metrics figures of Inception V4

In Table 1 the comparison between ResNet-50 and Inception V4 is demonstrated. Based on the data in the table, Inception V4 could perform slightly better predictions than ResNet-50. For the loss function, we can say that A lower loss function value indicates that the model has a better fit to the data, as it means that the differences between the predicted values and the true values are smaller. In this case, the Inception V4 model has a loss function value of 0.02094, while the ResNet-50 model has a loss function value of 0.02358. This means that the Inception V4 model is a better fit for the data compared to the ResNet-50 model. However, it's important to keep in mind that the loss function is only a measure of how well the model fits the training data, and not how well it generalizes to new, unseen data. Based on the R-squared values, the Inception V4 model appears to have a better fit to the data compared to the ResNet-50 model. The R-squared value of 86.34 for the Inception V4 model indicates that it explains 86.34 percent of the variability in the dependent variable, while the R-squared value of 81.08 for the ResNet-50 model indicates that it explains 81.08 percent of the variability in the dependent variable. Moreover, A lower RMSE value indicates that the model has a better fit to the data, as it means that the differences between the predicted values and the true values are smaller. In this case, the Inception V4 model has a RMSE value of 15.09, while the ResNet-50 model has a RMSE value of 17.85. This means that, on average, the Inception V4 model makes predictions that are closer to the true values compared to the ResNet-50 model. Therefore, based on the RMSE values, the Inception V4 model appears to be a better fit for the data compared to the ResNet-50 model. Finally, A lower MAPE value indicates that the model has a better fit to the data, as it means that the differences between the predicted values and the true values are smaller, relative to the true values. In this case, the Inception V4 model has a MAPE value of 9.83, while the ResNet-50 model has a MAPE value of 12.20. This means that, on average, the Inception V4 model makes predictions that are closer to the true values compared to the ResNet-50 model. Therefore, based on the MAPE values, the Inception V4 model appears to be a better fit for the data compared to the ResNet-50 model.

Model	Metrics	Output
ResNet-50	Testset Loss	0.02358
	Testset RMSE	17.85
	Testset MAPE	12.20%
	Testset R Squared	81.08%
Inception V4	Testset Loss	0.02094
	Testset RMSE	15.09
	Testset MAPE	9.83%
	Testset R Squared	86.34%

TABLE I
COMPARISON OF TEST RESULTS FOR RESNET-50 AND INCEPTION V4 MODELS

VII. CONCLUDING REMARKS

Bone age prediction with a low percentage of error is still a challenging task in the real world and medical usage. In this paper, we considered Two robust architectures of deep CNN networks which are very popular in machine vision tasks including classifications and regression. We are faced with a challenging dataset in this project in terms of the dataset size and the type of supervised task we conduct which was a regression operation. Overall, both of the models have shown a good result in the performance evaluation. For ResNet-50 took approximately 3.5 hours to be trained and the InceptionV4 took more than 5.5 hours. Due to the resource limitations we dealt with, we suffice these results, but there are some techniques we can use to significantly improve the model's performance. We can use an attention mechanism to focus on the bone parts and ignore the blank space (using sub-patches), we can increase the resolution of images which consumes more memory, we can increase the number of epochs which consumes more time and, we can use lower batch size which consumes more time. In addition, we used the transfer learning technique here and saved the trained models and their related histories to use later in the demo representation. Finally, we have to mention that there exist several architecture to predict bone age. One of these architecture is a powerful deep CNN model which combines ResNet-50 and InceptionV4 and it is called Inception ResNetV2 which improves accuracy due to combining the block of inception and residual connections of ResNet but as mentioned above we need more sources and powerful equipments to train and evaluate these powerful models.

REFERENCES

[1] Rajpurkar, P., Irvin, J., Zhu, K., Yang, B., Mehta, H., Duan, T. Automated bone age assessment using deep convolutional neural networks. In Proceedings of the Conference on Medical Image Computing and Computer-Assisted Intervention, San Francisco, CA, US, 2017.

[2] Loeff, F., Taebi, B., Moltz, J. H., & von Tscharner. Transfer learning for bone age assessment. Medical Image Analysis, 2019.

[3] Wang, Z., Tan, T., Wang, X., Zhang, Y., & Li, J. Multi-Modal Bone Age Assessment using Deep Convolutional Neural Networks. Medical Image Analysis, 2021.

[4] Liu, Y., Xiong, J., & Li, H. Attention-Based Deep Convolutional Neural Network for Bone Age Assessment. Medical Image Analysis, 2021.

[5] Pettersen et al. Hand radiographs for skeletal age assessment in pediatric populations. The Clinical Radiology and Radiotherapy, 2015.

[6] Lippe et al. Bone age in endocrine disorders. Pediatric Endocrinology and Metabolism, 2016.

[7] Bonse et al. Bone age in puberty. The Journal of Adolescent Health, 2013.

[8] Tassone et al. Bone age in genetic disorders. Medical Genetics Part C: Seminars in Medical Genetics, 2013.