" Predict bone age from hand radiographs "

Hamrouni Nawres Atidel[†], Hanen Hassen[‡]

***Abstract*—With the progress of medical technology, people need to extract the effective information contained in big biomedical data to promote the development of precision medicine. Deep learning plays an increasingly important role in the field of medical health and has a broad prospect of application. Bone Age prediction is an important medical phenomenon that is used to monitor the growth of children and adolescents. This helps in the detection of genetic growth disorders, and orthopedic problems in the rudimentary stages and helps in early treatment. The rapid advancements in technology now assist to identify bone age from hand radiographs using Deep learning techniques. In this pipeline, we will evaluate the performance of various models and produce the highest-quality radiographic images. We evaluated the accuracy and selected the optimal pipeline and initialization to improve precision by experimenting with different model architectures on a large dataset of pre-processed hand radiographs and adjusting the parameters of pre-trained models.**

***Index Terms*—Neural Networks, Recurrent Neural Networks, Deep Learning, Bone Age Assessment (BAA), Multiple-instance learning, Hand radiograph.**

I. Introduction

Bone age assessment (BAA) is a medical imaging technique used to evaluate the maturity of a child's bones, typically their hand and wrist radiographs. BAA is used to determine the skeletal maturity of a child, which can be helpful in diagnosing conditions such as growth disorders and endocrine disorders, as well as monitoring the progression of these conditions. The assessment is typically performed by a radiologist or pediatric endocrinologist, who will compare the child's radiograph to a set of standardized radiographs of children of known ages. BAA is commonly used in endocrinology, pediatrics, and rheumatology [1]. The main objective is to create an automated bone age assessment system that is accurate, precise, and efficient enough to be of practical use in clinical settings. Since deep learning neural networks have been shown to be effective for image classification tasks, this technique has become increasingly popular in the field of medical image analysis. Tasks related to medical images that involve cognitive and inferential processes include identifying and grouping images, determining numerical values, dividing images into segments, and identifying specific features. Specifically, using convolutional neural networks has been shown to be effective in identifying and measuring features within images [2].

The organization of this report is as follows. In Section II, we review and analyze the methods and results of previous research related to our project. Then, in Section III, we go through the technical tools and procedures and clarify the structures chosen for the project. Section IV provides a comprehensive overview of the input data and the pre-processing techniques employed. Section V is the key component of the report, as it delves into the specific details of the learning framework, including the parameters and structures. Finally, Section VI presents the final results and discusses the impact of the different architectures used.
In this study, we aim to investigate the use of AI for bone age estimation, improve upon current state-of-the-art deep learning techniques, and evaluate the accuracy of various experiments. Our process for understanding bone age prediction from radiographs involves selecting the appropriate input features by manipulating images, comparing the capabilities of multiple models, and choosing the most suitable ones based on previous studies and our own experience. Our project not only provides a solution for the bone age prediction problem but it can also be extended to other image classification tasks and optimized by adding data and adjusting parameters. First, we will design and construct the image data loader and determine an appropriate distribution of the number of classes and their ranges. Pre-processing the radiographs is a crucial step as it decreases training time and increases inference speed. We will then experiment with various architectures such as CNN, Inception, and EfficientNet, and adjust their parameters to achieve optimal results. Additionally, we will analyze the impact of gender information on the classification task and assess the outcomes.

II. Related Work

Deep neural network models have been very successful in classifying age based on facial images. The outstanding and effective performance of deep learning using Convolutional Neural Networks (CNN) has made image recognition highly responsive and accessible. The authors looked into using radiological images to evaluate and identify medically relevant information such as diseases and skeletal maturity. The study [3] showed the application of deep learning techniques based on CNNs by outlining the process of collecting data, implementing CNN, and carrying out the training and testing phases. The variability in the training radiographs (input images that vary greatly in intensity, contrast, and grayscale) makes it difficult for algorithms to identify key features. As a result, they proposed a novel pre-processing engine that uses a detection CNN to locate the hand and create a corresponding mask, followed by a vision pipeline to standardize and enhance the invariant features of images. They utilized transfer learning, which involves utilizing well-trained low-level knowledge from a large-scale dataset and

[†] Department of Information Engineering, University of Padova, email: {naouresatidel.hamrouni}@studenti.unipd.it

[†] Department of Information Engineering, University of Padova, email: {hanen.hassen}@studenti.unipd.it

then adjusting the weights to make the network specific to the target application. The three tested architectures were AlexNet, GoogleNet, and VGG-16. In terms of accuracy, VGG-16 was the best performer, AlexNet was the weakest, and GoogLeNet was the most efficient neural network. The held-out test images [3] showed 57.32% and 61.40% accuracy for the female and male cohorts. Compared to qualified radiologists and current automated systems for assessing skeletal maturity on hand radiographs, the Deep-learning technique performed similarly and efficiently. To measure the overall model performance, the root means square (RMS) and mean absolute difference (MAD) were used to compare the model accuracy with the given standard bone ages. In research [4], CaffeNet was used due to its low complexity, making it an ideal choice for illustrating hand X-ray age estimation. CaffeNet has many edges (weights) connecting its neurons that needed to be configured, so a pre-trained weight set on ImageNet was used. The results showed a slight overfitting, likely due to the small size of the training data set or the shallow network architecture. Also, in the research, [5] a deep residual network with 50 layers was used after a basic preprocessing method was applied to the RSNA dataset. The results were compared to medical reports and observations from three reviewers. RSNA used the Inception V3 model on pixels and combined it with a dense network that took age information as input. We can see in the research [6], they dealt with a smaller dataset, but the high quality of the radiographs and the clear separation of bones made it an important study. The classification was based on a model using the same basic rules as a CNN.

The researchers also investigated the number of layers and characteristics of neural networks. After filtering and denoising as part of the preprocessing step, the ResNet network was deemed the best model for the prediction task. The number of layers was increased from 18 to 152 and the parameters of the layers, such as the optimizer and activation function, were altered. The study showed that not only the choice of model is important, but also the tuning of its parameters [7].

In a recent study [8], Bone Age Assessment (BAA) using whole-body CT Scans was addressed. The model was based on VGGNet, with additional networks or layers added to balance the features generated and prevent over-fitting. The study also provided insights into the ROI and the impact and importance of gender classification in BAA. There have been many efforts in previous studies [3] to simplify the process and improve the results of bone age assessment.

In this study, we will build upon the work done in the paper [5] which includes important elements such as preprocessing, cropping, and gender information. After selecting a suitable feature space from images, we will experiment with various architectures and fine-tune their hyper-parameters. Instead of viewing the BAA problem as a regression task, we will treat it as a classification problem, as most previous works have done. Both regression and classification are supervised learning methods, but regression outputs a continuous value within a range, while classification assigns a fixed class label to each component to make predictions, which is more appropriate for grouping age ranges into intervals.

III. Processing Pipeline

To understand the stages of the work, we should present the processing pipeline. This will simplify the data processing tasks and reduces manual intervention. Figure 1 shows the processing pipeline of our work.

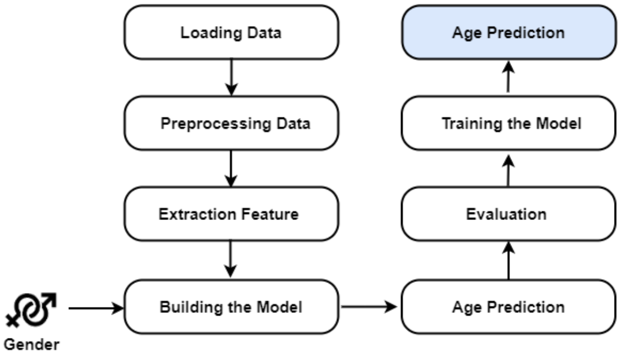

Fig. 1: Processing pipeline

- The initial step is to load and visualize the data provided to understand it by extracting the needed information from it.
- Pre-processing, this second step is used to clean, transform, and prepare data before it is fed into our model. It ensures that the data is in a suitable format for training and that it provides the model with the information it needs to make accurate predictions.
- The next step is to extract features. This is a significant stage in our process pipeline as it can help to improve the model performance, increase the interpretability of the model, and reduce the risk of overfitting.
- The subsequent step is to build the model with the radiographs data. We will later incorporate the gender information and added it to the input of the model.
- After training the model, we must fine-tune the parameters to enhance accuracy and evaluate its prediction capabilities. Four distinct models will be trained and their performance will be analyzed and presented in the results section.

IV. Signals and Features

- **Data-set Presentation**

In this project, we are using data provided by the Radiological Society of North America (RSNA). [1] This Data set consisting of 14 236 hand radio-graphs was made available to the participants in the RSNA Pediatric Bone Age Machine Learning Challenge in 2017 [9]. Images for the training and validation sets were obtained from the Children's Hospital in Colorado and the Lucile Packard Children's Hospital at Stanford. The images were labeled, with skeletal age estimates and sex from

[1] RSNA: a professional organization for radiologists, medical physicists, and medical imaging professionals.

the accompanying clinical radiology report provided at the time of imaging. This information will be added as input as shown in figure 1 Our data is divided into three sections:

1) The training data set, which contained 12 611 hand radiographs. These pieces of information are utilized to train the model and fit the network's weights.
2) Secondly, we utilized a validation data set consisting of 800 images to evaluate the effectiveness of the training process. By adjusting the parameters, we were able to determine the optimal set that produced the best results.
3) The final component is the test set, with approximately 200 samples, which is employed to confirm the accuracy of the predictions by estimating the ages of novel images and comparing the predictions to their actual labels.

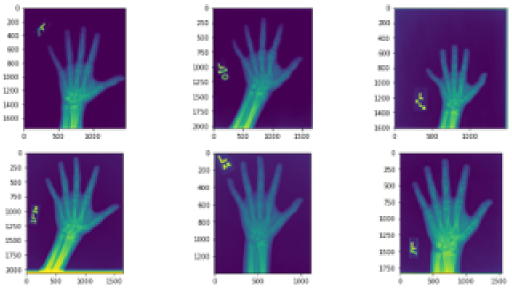

Fig. 2: Example of Radio-graphs from the data-set

- **Pre-processing Data**

There are two different approaches available : The first approach is to provide the model with the full image, while the second involves dividing the original image into smaller sections and determining the label based on the output from these sections. Both approaches require data pre-processing for the training phase. Foremost, let's start to establish the data loaders for the three datasets to obtain a data frame containing the image and its related information. Following that, we must resize images to have uniform dimensions. The package Bilateral Filter smooths the images based on the spatial domain and the range domain. Also, we need to enhance the image quality by increasing brightness and, critically, adjusting the contrast of the image, this is by using the ImageEnhance function from the PIL library. Thirdly, normalization adjusts the range of intensities in the image, making the result more standard and beneficial for training the model. By limiting the values of pixel intensities, this approach helps to eliminate some of the noise in the image. This technique is frequently used in AI applications and has been shown to improve model performance.

To improve the image's structure and remove unwanted details from it, the Gaussian blur was used. This software tackles the problem of information loss in global equalization by utilizing an adaptive method. It splits the image into smaller segments, performs histogram equalization on each segment independently, then combines nearby segments using bi-linear interpolation to minimize noise. The age labels for each image range from 1 to 228 months. Regression learning can handle this type of data easily, but with classification, adjustments need to be done. A large number of possible ages makes classification challenging and results in poor predictions, as the model struggles to distinguish between similar labels. To address this issue, we decided to categorize the ages into a set number of classes, each containing ages that are close to one another. The key factors to consider are the number of classes and the range of each class. We plotted the number of instances for each age to determine the best division of the class ranges and determine the number of classes based on the peaks in figure 3.

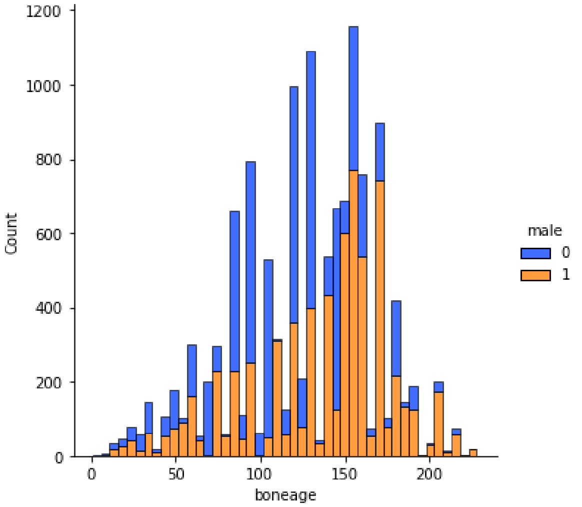

Fig. 3: Number of counts for each age and gender

The first approach to solving the age classification problem was to divide the age range into a smaller number of intervals (2 or 3) and assign each image to a class based on its age. This method had decent accuracy but was limited because having fewer classes made it difficult to distinguish between ages even if they weren't in the same range, leading to a loss of data. To address the limitations of the first method, the second option was to increase the number of classes. However, having too many classes (more than 10) would result in reduced accuracy, so the number of classes had to be carefully selected. After experimenting with the different number of classes, the optimal solution was found to be dividing the age range into 5 intervals and using Pandas' cut function to convert the continuous age variable into a discrete categorical variable. The results showed that dividing the data into 5 groups was the best choice.

V. Learning Framework

This section focuses on outlining the approach to solving the problem, which will be broken down into three main subsections for clarity in comprehending the learning model. 1) Preparing Data

In the context of diverse forms of input data, neural networks, and deep learning models are capable of handling a large volume of inputs. To prepare the data for training, we can either input the entire image into the model or divide the image into a fixed number of N sub-patches and pass each patch through training and prediction. The final class of the input would be determined by combining the results of all N patches based on the maximum number of votes. However, due to the computational complexity and burden of the latter option, we opted to work with the whole image. The gender of the patient, which is a valuable piece of information provided by the dataset, is also taken into consideration. Male and female bones differ in terms of shape, size, and density, especially as they age, with males having larger bones. This gender information, along with the associated picture, can be supplied to the model to improve the classification task. First, we will focus on classifying the pictures and then try to add the gender feature as parallel input to enhance the prediction.

2) The selection of parameters

- **Learning rate :**

The learning rate is a configurable hyperparameter utilized in training neural networks, which has a small positive value, typically between 0.0 and 1.0. It regulates the speed at which the model adjusts to the problem. After experimenting with different values of the learning rate and using a stable optimizer so we settled on a final value of 0.001.

- **Number of epochs**

This is a hyperparameter that defines the number of times that the learning algorithm will work through the entire training dataset. One epoch means that each sample in the training dataset has had an opportunity to update the internal model parameters. An epoch is comprised of one or more batches.

- **Batch size**

Batch size refers to the number of samples that are processed by the network at once. It is also commonly referred to as a mini-batch. After adjusting the batch size manually, we settled on a value of 64 samples.

- **Loss Function**

known as the cost function or objective function, is what the model aims to minimize during optimization. It converts the complex components and values of the model into a single scalar value that can be used to evaluate and compare performance. In a supervised learning task, the loss function assesses the model's ability to predict at a given step by comparing the actual output of the model with the predicted output. We will use cross entropy as the loss function because it minimizes the difference between the probabilities of reality and prediction and is commonly used in classification tasks. The equation 1 shows the loss function.

$$L = -1/N \sum_{C=1}^{N} P_c \log(1 - P_c) \tag{1}$$

While Pc is the probability of the true label.

- **Optimizer**

The Optimizer is a crucial aspect of the training process. It adjusts the weights based on the selected algorithm, gradient values, and the result of the loss function (which is cross entropy in this case). The chosen optimizer for this scenario is the Adam (Adaptive Moment Estimation) which utilizes an exponentially decaying average of past gradients to determine the direction of parameter updates, similar to the momentum optimizer.

- **Activation function**

Activation functions play a vital role in neural networks by breaking linearity and guiding the learning process. It defines the mapping of the weighted input sum to an output. The activation function used in the output layer is different from the function used in hidden layers. The hidden layers will utilize the rectified linear activation function (ReLu), which is widely used in neural networks, especially CNN, because it is computationally efficient and allows for fast convergence. The exponential linear activation function (ELu) may also be used, as it extends ReLu to handle negative values and has shown improved results. For the output layer, the softmax activation function (normalized exponential function) will be used to solve problems related to multiclass classification. The softmax calculates the relative probabilities and outputs the probability of each class to determine the final prediction. The softmax extends the sigmoid function, which is used for binary classification tasks. The following equation shows the activation function for $K = 1, 2, ..., N$

$$\sigma(Z_K) = \frac{e^{Z_1}}{\sum_{i=1}^{N} e^{Z_i}} \tag{2}$$

- **Batch Normalization**

When we normalize a dataset and start the training process, the weights in our model become updated over each epoch. So what will happen if, during training, one of the weights ends up becoming drastically larger than the other weight? This large weight can again cause the output from its corresponding neuron to be extremely large. Then, this imbalance will again continue to cascade through the neural network causing the problem where features with larger values will have a bigger impact on the learning process compared with the features with smaller values.

That is the reason why we need to normalize not just the input data, but also the data in the individual layers of the network. When applying batch norm to a layer we first normalize the output from the activation function.

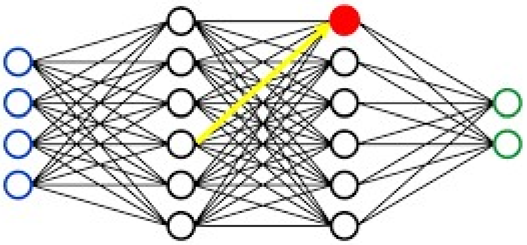

Fig. 4: Batch Normalization

- **Dropout**

Dropout is a technique used to reduce complexity in neural networks and prevent overfitting. The dropout rate determines the proportion of neurons that will be deactivated and set to 0. During training, the model randomly ignores a portion of neurons, creating a smaller network. The dropout layer can either be kept or replaced with the batch normalization technique as shown in figure 5.

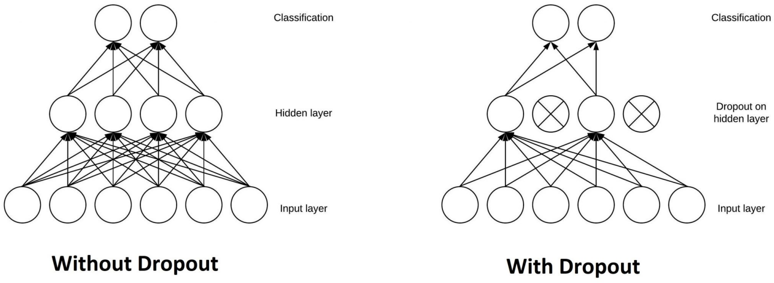

Fig. 5: Dropout Layers

- **Early stopping**

To accomplish regularization and minimize overfitting, we need to use early stopping as a solution. The method is straightforward: we track the loss function values using the validation set. When the value on the validation set shows no progress after a specific number of epochs, we can stop training and reduce the number of epochs.
The next figure 6 shows the early stopping condition.

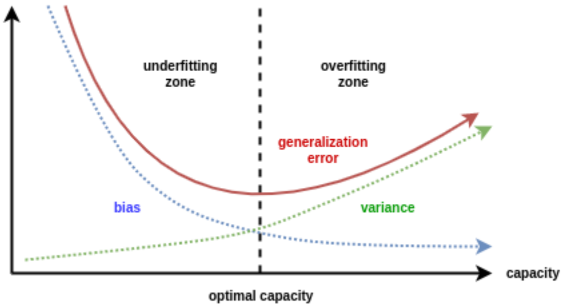

Fig. 6: Early stopping

3) Models

- **CNN**

A Convolutional Neural Network (CNN) is composed of an input layer, an output layer and a hidden layer that comprises multiple layers for convolution, pooling, fully connected, and normalization. The convolution layer is the crucial component of a CNN and performs the bulk of computation. It requires input data, a filter, and a feature map and is usually of type Conv2D when using images as inputs. The pooling layer reduces the spatial size of the representation to simplify the network and reduce computational cost, using either max pooling or average pooling. The fully connected layer (FC) connects neurons between two different layers and is typically placed before the output layer and forms the final layers of a CNN architecture.

- **EfficientNet**

EfficientNet is a Convolutional Neural Network (CNN) architecture that uses a compound coefficient to uniformly scale all aspects of depth, width, and resolution. The uniform scaling of width, depth, and resolution is performed with set scaling coefficients that maintain balance, leading to better capture of patterns in large images such as radiography. Compared to previous models, EfficientNet is not only faster but also more accurate. The EfficientNetB0 model will be used as a starting point.

- **Inception**

The Inception model is a variation of the Convolutional Neural Network (CNN) used for image analysis and named after the movie Inception. It consists of an inception layer made up of 1x1, 3x3, and 5x5 convolution layers and a max pooling layer. The inception layer reduces dimensionality while enabling deeper extraction, leading to more efficient computation. The final outputs from this layer are combined and passed on as a single unit to the next layer.

- **ResNet**

ResNet is a sophisticated and advanced neural network first introduced in 2015, composed of residual blocks with varying constructions. It is used to address complex problems by incorporating extra layers to extract more data. The blocks in ResNet feature direct linking, which bypasses some connections that could alter the output value. This allows the model to learn functions and ensures that the high-level layers produce results comparable to those from the lower levels.

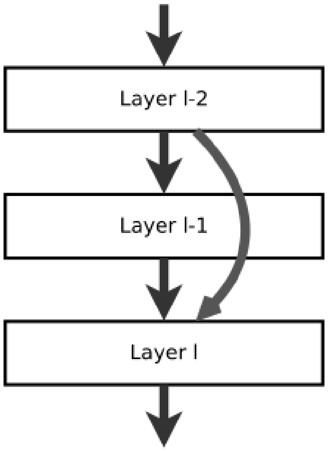

Fig. 7: Canonical form of a residual neural network. A layer l-1 is skipped on the activation of l-2

VI. Results

As the number of classes grows, it becomes more difficult for the model to accurately determine the correct class. This is demonstrated by the decreased performance when

Classes	Model	Training accuracy	Validation accuracy
3	Customized model	78.2	76.1
	EfficientNet	84.2	82.3
	Inception	85.5	83.6
	Resnet	85.72	84.36
5	Customized model	69	65.3
	EfficientNet	75.4	71.25
	Inception	71.52	70.63
	Resnet	75.6	72.3
10	Customized model	55.32	51.2
	EfficientNet	57.4	55.9
	Inception	56.33	54.87
	Resnet	57.36	54.63

using five classes compared to three classes, and even worse performance when using ten classes, which aligns with previous discussions.

The validation accuracy was improved when gender information was included, showing its significance. After a thorough investigation, it was found that this information is particularly valuable at the edges of classes. In other words, when classifying becomes more challenging near the class boundaries, the added information can help solve the issue, which is why gender information can help reclassify some samples. The results of training and validation accuracy and loss using three fine-tuned models, EfficientNet, Inception, and ResNet, are presented below.

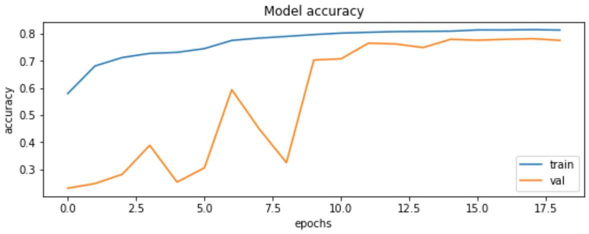

Fig. 8: Training and validation accuracy using EfficientNet

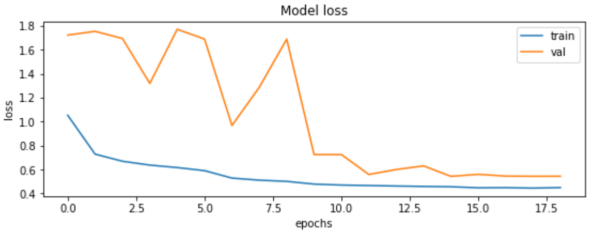

Fig. 9: Training and validation loss using EfficientNet

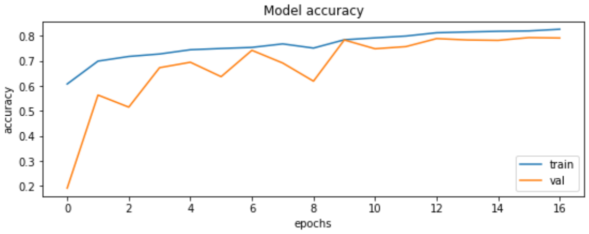

Fig. 10: Training and validation accuracy using Inception

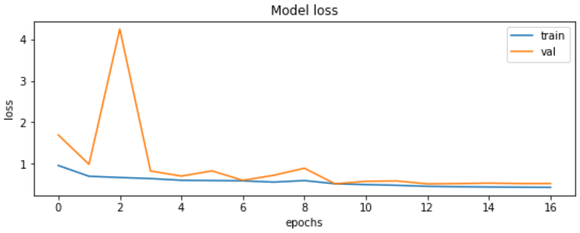

Fig. 11: Training and validation loss using Inception

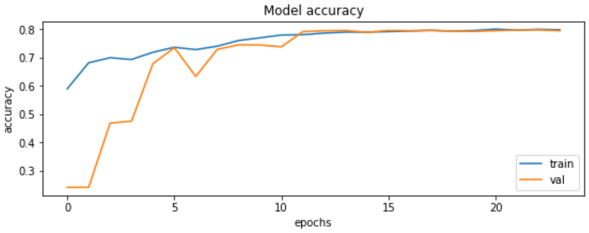

Fig. 12: Training and validation accuracy using ResNet

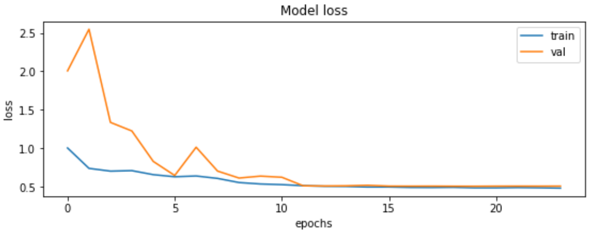

Fig. 13: Training and validation loss using ResNet

VII. Concluding Remarks

In essence, this project allowed us to put our theoretical understanding into practical use by working with deep images. We utilized several pre-processing techniques to enhance the image and make the bones more visible. We discovered the significance of data division and category choice and attempted to add a new layer to the network to include gender information, which led to an improvement in the accuracy of the model. Our future plans include applying the models to different types of data that involve dividing the image into small segments, and exploring different techniques and architectures. Finally, the project provided us with an opportunity to work on a real-world problem utilizing classification instead of regression, which has proven useful in the field of medicine and holds potential for use in other areas.

References

[1] C. Wang, Y. Wu, C. Wang, X. Zhou, Y. Niu, Y. Zhu, X. Gao, C. Wang, and Y. Yu, "Attention-based multiple-instance learning for pediatric bone age assessment with efficient and interpretable," *Biomedical Signal Processing and Control*, vol. 79, p. 104028, 2023.

[2] M. Mansourvar, M. A. Ismail, T. Herawan, R. Gopal Raj, S. Abdul Kareem, and F. H. Nasaruddin, "Automated bone age assessment: motivation, taxonomies, and challenges," *Computational and mathematical methods in medicine*, vol. 2013, 2013.

[3] H. Lee, S. Tajmir, J. Lee, M. Zissen, B. A. Yeshiwas, T. K. Alkasab, G. Choy, and S. Do, "Fully automated deep learning system for bone age assessment," *Journal of digital imaging*, vol. 30, pp. 427–441, 2017.

[4] J. H. Lee and K. G. Kim, "Applying deep learning in medical images: the case of bone age estimation," *Healthcare informatics research*, vol. 24, no. 1, pp. 86–92, 2018.

[5] D. B. Larson, M. C. Chen, M. P. Lungren, S. S. Halabi, N. V. Stence, and C. P. Langlotz, "Performance of a deep-learning neural network model in assessing skeletal maturity on pediatric hand radiographs," *Radiology*, vol. 287, no. 1, pp. 313–322, 2018.

[6] J. R. Kim, W. H. Shim, H. M. Yoon, S. H. Hong, J. S. Lee, Y. A. Cho, and S. Kim, "Computerized bone age estimation using deep learning based program: evaluation of the accuracy and efficiency," *American Journal of Roentgenology*, vol. 209, no. 6, pp. 1374–1380, 2017.

[7] X. Chen, J. Li, Y. Zhang, Y. Lu, and S. Liu, "Automatic feature extraction in x-ray image based on deep learning approach for determination of bone age," *Future Generation Computer Systems*, vol. 110, pp. 795–801, 2020.

[8] M. Duan, K. Li, C. Yang, and K. Li, "A hybrid deep learning cnn–elm for age and gender classification," *Neurocomputing*, vol. 275, pp. 448–461, 2018.

[9] S. S. Halabi, L. M. Prevedello, J. Kalpathy-Cramer, A. B. Mamonov, A. Bilbily, M. Cicero, I. Pan, L. A. Pereira, R. T. Sousa, N. Abdala, *et al.*, "The rsna pediatric bone age machine learning challenge," *Radiology*, vol. 290, no. 2, pp. 498–503, 2019.