Pediatric Bone Age Estimation

Project Report

Jayaashri Chezhian
033694399
May 10, 2025

Instructor: Dr. Shabnam Sodagari

CECS 553

Machine Vision

Spring 2025

ABSTRACT

Accurate bone age assessment in pediatric patients is critical for diagnosing growth disorders and planning treatment. In this work, I leverage the RSNA Pediatric Bone Age dataset, consisting of 12,611 hand radiographs labelled with chronologic age and patient sex, to train a deep-learning regression model. I built a TensorFlow tf.data pipeline to efficiently load, preprocess (resize to 300×300 px, normalize, augment), and batch images. My regression network combines an ImageNet-pretrained EfficientNet-B3 backbone (without the classification head) with a scalar sex input, followed by global average pooling and two fully connected layers to predict bone age in months. I used mean absolute error (MAE) with an Adam optimizer and employ early stopping, learning-rate reduction on plateau, and model checkpointing. The final model achieves a mean absolute error of 3.82 months. Additionally, I present the learning curves for both training and validation, as well as a breakdown of how prediction errors are distributed. While this notebook focuses on regression, future work will incorporate a segmentation stage (e.g. U-Net) to isolate the hand prior to age estimation.

TABLE OF CONTENTS

INTRODUCTION

Bone age assessment is a critical component in evaluating pediatric growth and development, aiding in the diagnosis and management of various endocrine and genetic disorders. By comparing the skeletal maturity observed in a hand and wrist X-ray to standardized references, doctors can gain valuable insights into a child's growth trajectory and potential future height. This assessment is particularly important for:

- **Diagnosing and monitoring growth disorders:** Conditions like growth hormone deficiency, precocious puberty, and delayed puberty often manifest with discrepancies between chronological age and bone age. Accurate bone age helps in timely diagnosis and evaluating the effectiveness of treatments.
- **Predicting adult height:** Bone age is a key factor in predicting a child's final adult height, which is crucial for counselling families and making decisions about growth-modifying therapies.
- **Timing of medical interventions:** Decisions regarding the timing of certain medical or surgical procedures in pediatric patients may depend on their skeletal maturity.
- **Forensic analysis:** In some cases, bone age assessment can be used for age estimation in individuals whose chronological age is unknown.

Traditionally, bone age is determined by comparing a patient's hand and wrist X-ray to standardized atlases, such as the Greulich and Pyle or Tanner-Whitehouse methods. While widely used and considered the clinical standard, these methods are inherently subjective, relying heavily on the radiologist's or doctor's experience and interpretation. This subjectivity can lead to significant observer variability, potentially impacting diagnostic accuracy and treatment planning.

The increasing availability of medical imaging data in digital formats and rapid advancements in deep learning techniques present a significant opportunity to address the limitations of traditional bone age assessment.

The objective of this project is to develop and evaluate a deep learning model capable of accurately predicting bone age from hand X-ray images. This project uses RSNA Bone Age dataset (taken form Kaggle) and convolutional neural networks (CNNs) to accurately predict the bone age. By leveraging the power of deep learning, this project aims to create a reliable and efficient tool that can potentially reduce subjectivity, save time for clinicians, and provide a consistent aid in the diagnosis and management of pediatric growth disorders.

BACKGROUND

2.1. TRADITIONAL BONE AGE ASSESSMENT METHODS

Doctors have traditionally determined skeletal maturity by manually comparing a patient's hand and wrist X-ray to standardized reference charts or scoring schemes. In the Greulich–Pyle approach, the radiograph is matched by eye to the closest plate in an age and sex specific atlas, yielding an estimate of bone age in monthly increments. Although this method is intuitive, it demands considerable time and is vulnerable to differences both between and within observers.

The Tanner–Whitehouse technique takes a more quantitative tack, assigning scores to key bone growth sites such as the phalanges, metacarpals, and radius and converting the summed score into a bone-age measurement. Despite its systematic scoring, TW still relies on precise landmark identification and can exhibit scorer variability. Moreover, a range of factors can further affect accuracy:

- **Sex differences:** Boys and girls follow slightly different maturation timelines, so misapplication of a sex-specific chart can skew results.
- **Ethnicity and genetics:** Population-specific growth patterns mean that reference atlases may not be equally valid across ethnic groups.
- **Radiograph quality and positioning:** Poor contrast, motion blur, or inconsistent hand positioning can obscure critical landmarks.
- **Health and nutritional status:** Endocrine disorders, chronic illnesses, or nutritional deficiencies can accelerate or delay bone maturation.

Together, these variables underscore why manual bone age assessment remains both labour intensive and prone to inconsistency.

2.2. MACHINE LEARNING AND DEEP LEARNING FOR BONE AGE PREDICTION

Early automated approaches extracted handcrafted features such as landmark distances, area ratios of epiphyses, or texture descriptors from segmented hand regions and fed them into classical regressors (e.g., support vector machines, k-nearest neighbours). These methods achieved modest accuracy but depended heavily on precise feature engineering and segmentation.

With the introduction of convolutional neural networks (CNNs), researchers began training end-to-end models directly on raw radiographs. Researchers have demonstrated that deep CNNs could learn hierarchical image features relevant to bone maturation, often outperforming traditional pipelines. More recent work has explored two-stage frameworks, first segmenting the hand via U-Net, then regressing age with a CNN and hybrid transformer-CNN architectures to further boost performance.

2.3. ADVANTAGES OF DEEP LEARNING IN BONE AGE ESTIMATION

Deep learning offers several key benefits over manual and classical machine learning techniques. First, CNNs automatically learn discriminative features at multiple scales, ranging from global bone shape down to fine-grained ossification patterns, eliminating the need for manual feature design. Second, end-to-end training on large, labelled datasets such as RSNA's enables models to generalize across patient populations and imaging conditions. Third, integrated architectures can incorporate auxiliary inputs (e.g., patient sex) alongside images, reducing bias and improving accuracy. Finally, once trained, DL models deliver rapid, reproducible bone age estimates, paving the way for real-time clinical decision support.

DATASET & DATA ANALYSIS

3.1. DATASET

The dataset I chose in this project is the RSNA(Radiological Society of North) Pediatric Bone Age collection, originally released as part of a America Kaggle challenge. The has already been split into training and testing dataset.

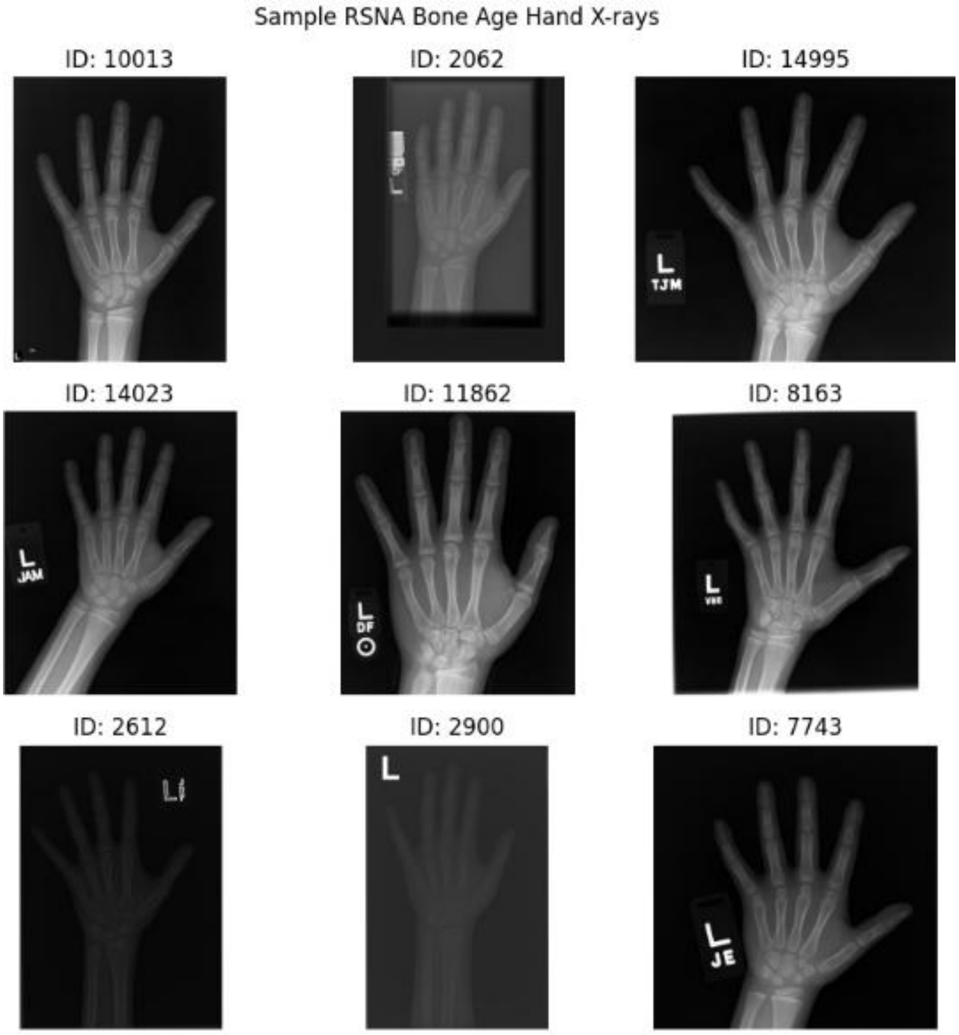


Figure.3.1: Sample of the image present in the dataset.

Training:

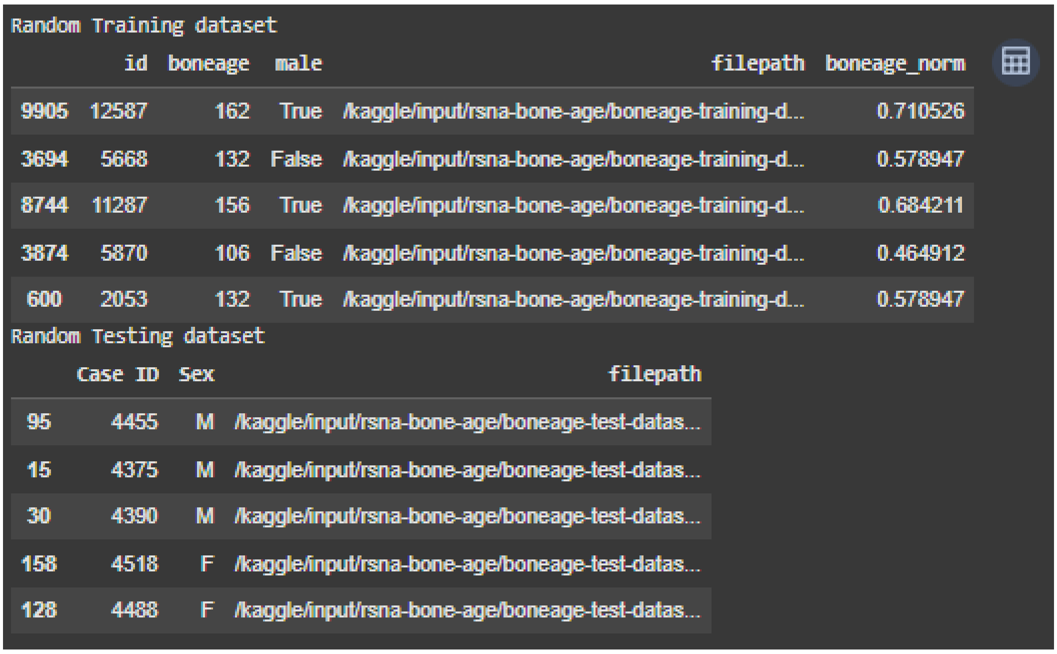
- There are 10,088 radiographs each with metadata fields. The fields are
    - id (unique study identifier).
    - Bone age (ground-truth age, in months).
    - Male (Boolean flag for sex).

Testing:

- There is 200 radiographs, with columns
    - Case ID
    - Sex
- There is no ground truth for this test dataset therefore I split the training dataset into 80:20 to create a validation set so, I could analyse model performance.

Validation:

- The validation partition comprises 20% of the full dataset 12,611, which is 2528 images.
- During training, validation MAE was used to guide early stopping, learning rate reduction, and to choose the best checkpointed weights.



Random Training dataset

	id	boneage	male	filepath	boneage_norm
9905	12587	162	True	/kaggle/input/rsna-bone-age/boneage-training-d...	0.710526
3694	5668	132	False	/kaggle/input/rsna-bone-age/boneage-training-d...	0.578947
8744	11287	156	True	/kaggle/input/rsna-bone-age/boneage-training-d...	0.684211
3874	5870	106	False	/kaggle/input/rsna-bone-age/boneage-training-d...	0.464912
600	2053	132	True	/kaggle/input/rsna-bone-age/boneage-training-d...	0.578947

Random Testing dataset

	Case ID	Sex	filepath
95	4455	M	/kaggle/input/rsna-bone-age/boneage-test-datas...
15	4375	M	/kaggle/input/rsna-bone-age/boneage-test-datas...
30	4390	M	/kaggle/input/rsna-bone-age/boneage-test-datas...
158	4518	F	/kaggle/input/rsna-bone-age/boneage-test-datas...
128	4488	F	/kaggle/input/rsna-bone-age/boneage-test-datas...

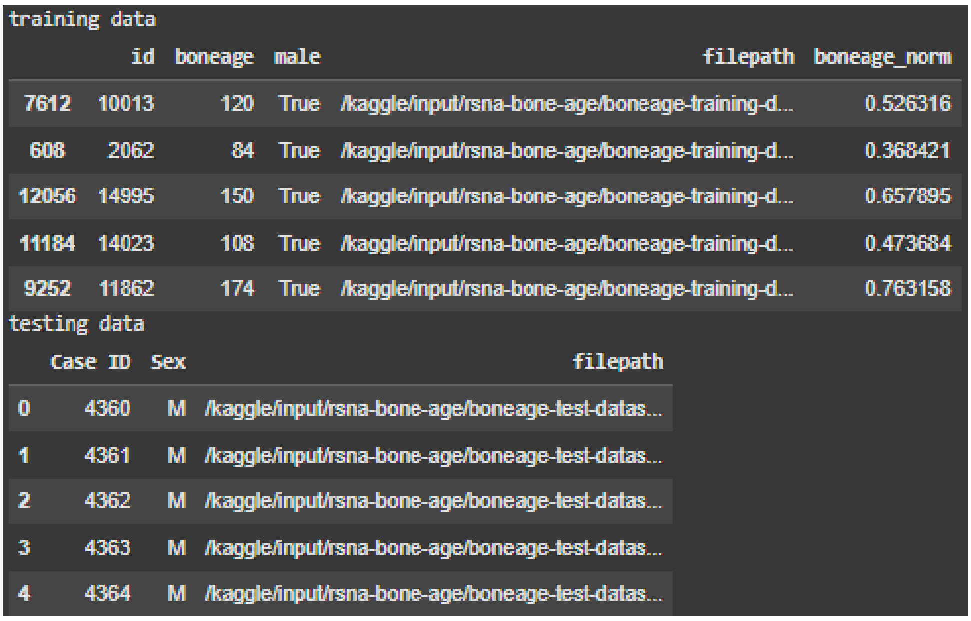
Figure.3.2 : 5 random sample of the columns present in the dataset.

Key Characteristics of the Dataset

- Bone age distribution: The ages span the full pediatric range 1–228 months, which is 1 month old to 19 years old, with a mean of approximately 127 months (σ ≈ 41 months).
- A histogram with KDE overlay confirms a roughly Gaussian shape centred near 11 years—identical in shape to the training distribution.
- Sex balance: Roughly 58 % male and 42 % female, matching the overall dataset's ratio to prevent sex-related bias during tuning.
- The dataset is made of X-rays of the left-hand.

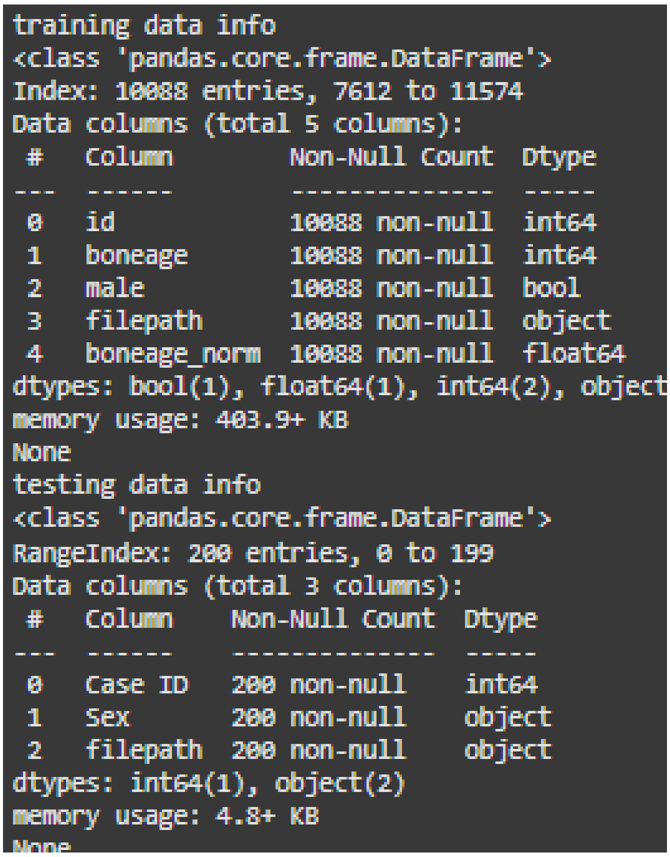
3.2. DATA ANALYSIS

The train_df.head() confirms that there 10,088 non-null entries across five columns, while test_df.info() shows 200 entries across three columns.



```
training data
          id  boneage  male                                       filepath  boneage_norm
7612   10013      120  True  /kaggle/input/rsna-bone-age/boneage-training-d...     0.526316
608     2062       84  True  /kaggle/input/rsna-bone-age/boneage-training-d...     0.368421
12056  14995      150  True  /kaggle/input/rsna-bone-age/boneage-training-d...     0.657895
11184  14023      108  True  /kaggle/input/rsna-bone-age/boneage-training-d...     0.473684
9252   11862      174  True  /kaggle/input/rsna-bone-age/boneage-training-d...     0.763158
testing data
   Case ID  Sex                                      filepath
0     4360    M  /kaggle/input/rsna-bone-age/boneage-test-datas...
1     4361    M  /kaggle/input/rsna-bone-age/boneage-test-datas...
2     4362    M  /kaggle/input/rsna-bone-age/boneage-test-datas...
3     4363    M  /kaggle/input/rsna-bone-age/boneage-test-datas...
4     4364    M  /kaggle/input/rsna-bone-age/boneage-test-datas...
```

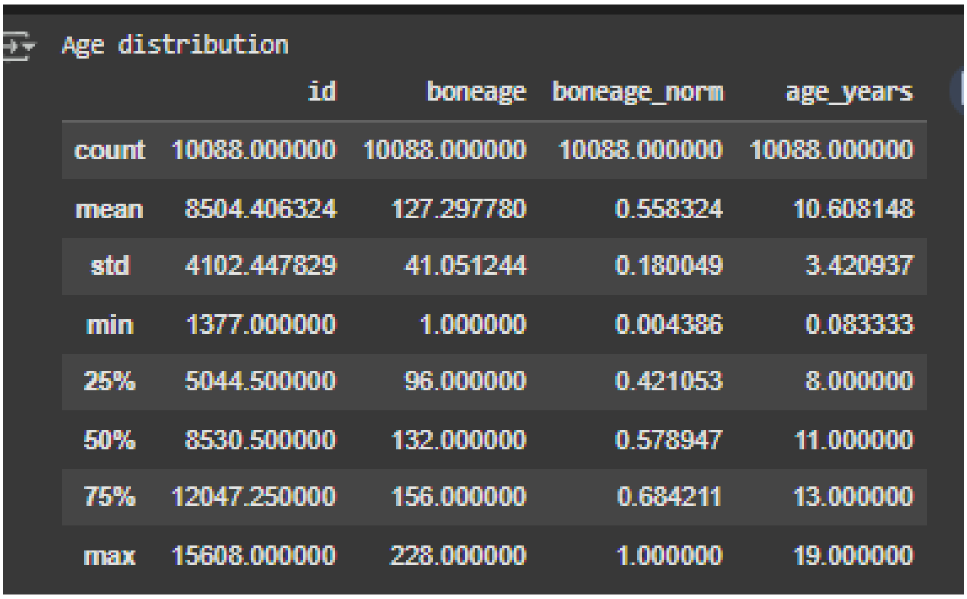
Figure.3.3: output of train_df.head().



```
training data info
<class 'pandas.core.frame.DataFrame'>
Index: 10088 entries, 7612 to 11574
Data columns (total 5 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   id            10088 non-null  int64
 1   boneage       10088 non-null  int64
 2   male          10088 non-null  bool
 3   filepath      10088 non-null  object
 4   boneage_norm  10088 non-null  float64
dtypes: bool(1), float64(1), int64(2), object
memory usage: 403.9+ KB
None
testing data info
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 3 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   Case ID   200 non-null    int64
 1   Sex       200 non-null    object
 2   filepath  200 non-null    object
dtypes: int64(1), object(2)
memory usage: 4.8+ KB
None
```

Figure.3.4: output of train_df.info().

Additionally, I prepared an in-depth overview that characterizes how ages are distributed throughout the dataset.



Age distribution

	id	boneage	boneage_norm	age_years
count	10088.000000	10088.000000	10088.000000	10088.000000
mean	8504.406324	127.297780	0.558324	10.608148
std	4102.447829	41.051244	0.180049	3.420937
min	1377.000000	1.000000	0.004386	0.083333
25%	5044.500000	96.000000	0.421053	8.000000
50%	8530.500000	132.000000	0.578947	11.000000
75%	12047.250000	156.000000	0.684211	13.000000
max	15608.000000	228.000000	1.000000	19.000000

Figure.3.5: Summary of age distribution in the dataset.

Looking at a histogram with a kernel density estimate (KDE) reveals an approximately Gaussian shape centred around 120–140 months. Replotting in years maintains the same pattern (Figure 3.5). This distribution suggests adequate coverage across the pediatric range, though very young infants (< 6 months) and late adolescents (> 17 years) are sparsely represented.

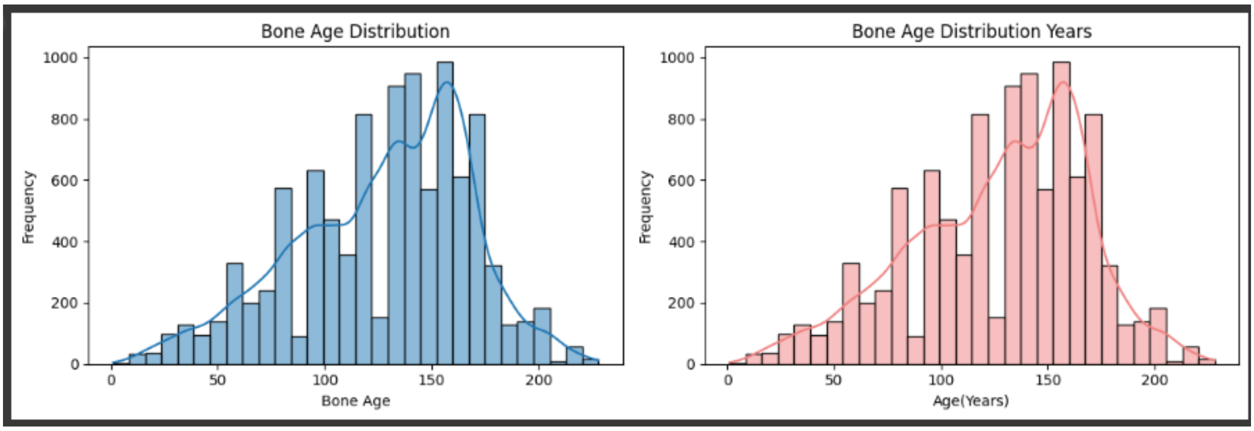


Figure.3.6: Histogram of age distribution

A simple count plot of the Boolean male flag shows approximately 58 % male and 42 % female in the training partition (Figure3.6). While not heavily imbalanced, this skew motivates incorporating sex as an explicit model input to reduce potential bias in age predictions.

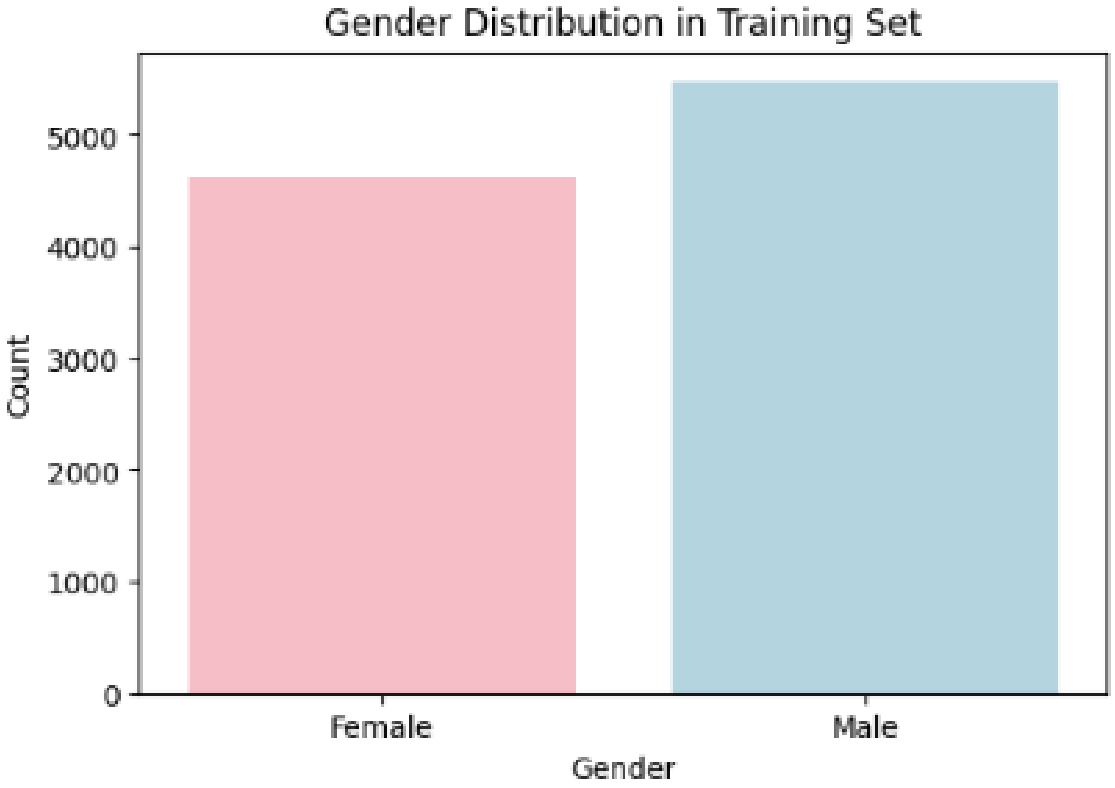


Figure.3.7: Gender Distribution in the dataset.

DATA PREPROCESSING

4.1. PREPROCESING

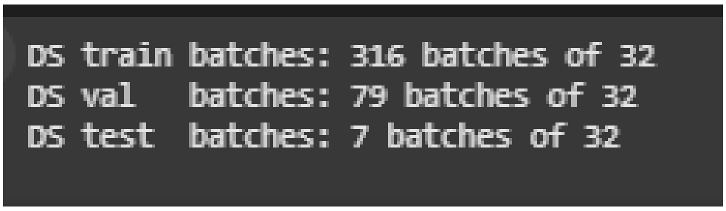
Data preprocessing begins by loading the RSNA metadata CSV into pandas and converting it into three tf.data.Dataset objects, training, validation, and test, by mapping each row's ID, age label, sex flag, and file path into dataset elements. I defined a parse_single() function that:

1. Reads and decodes the PNG via tf.io.read_file and tf.image.decode_png(channels=3).
2. Casts pixels to float32 and resizes to 300×300 px.
3. Normalizes values to [0,1] and applies the effnet_preprocess transform.
4. Converts the Boolean sex flag into a numeric tensor and scales the bone-age label by 1/228.

To speed up end-to-end training, I then:

- Parallelize all parsing and augmentation with dataset.map(parse_single, num_parallel_calls = tf.data.AUTOTUNE).
- Cache the fully parsed training dataset in memory (after the first epoch) to eliminate repeated disk reads.
- Shuffle (buffer = 1000) and batch (size = 32) each split.
- Prefetch with dataset.prefetch(tf.data.AUTOTUNE) to overlap data preprocessing with GPU computation.

These optimizations produce 316 training batches, 79 validation batches, and 7 test batches of size 32, and together with GPU acceleration reduce per-epoch wall clock time by roughly 30%.



```
DS train batches: 316 batches of 32
DS val   batches: 79 batches of 32
DS test  batches: 7 batches of 32
```

Figure.4.1: Image of the dataset after turning them into batches.

4.2. DATA AUGMENTATION

To support the model's ability to handle differences in hand alignment, scale, and exposure, I performed a series of augmentations on each 300 × 300-pixel radiograph during training. Variants are generated by randomly rotating images up to ±20 °, translating them horizontally or vertically by up to 10 %, zooming in or out by ±10 %, and modifying brightness between 0.8x and 1.2x the original.

I deliberately omitted horizontal flip, since every case depicts the left hand, to the preserve anatomical laterality. I also implement a composite augmentation pipeline that applies all these transforms at once, using the nearest-neighbour interpolation to fill any gaps, so that the network is exposed to a rich array of realistic variations in positioning, framing, and lighting. This diversification of the training data helps the regression model generalize more reliably to new hand X-rays.

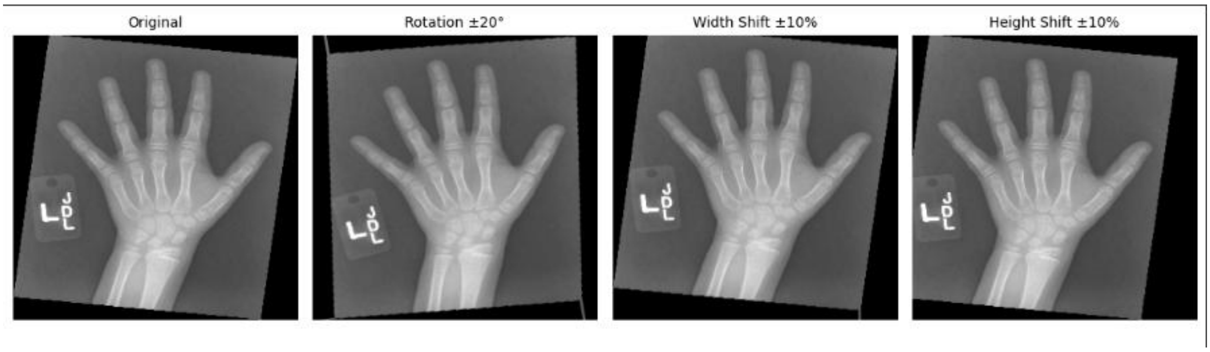


Figure.4.2: Sample images of dataset after data augmentation.

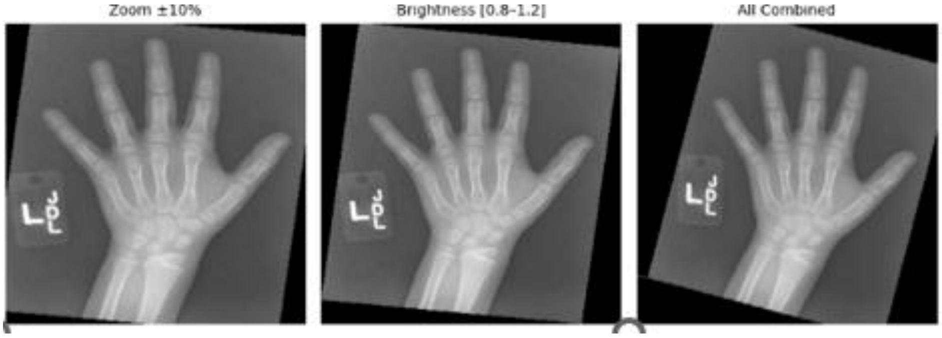


Figure.4.3: Sample images of dataset after data augmentation.

MODEL ARCHITECTURE

To establish a performance benchmark, I implement a straightforward, end-to-end regression network that ingests the full hand radiograph (no segmentation).

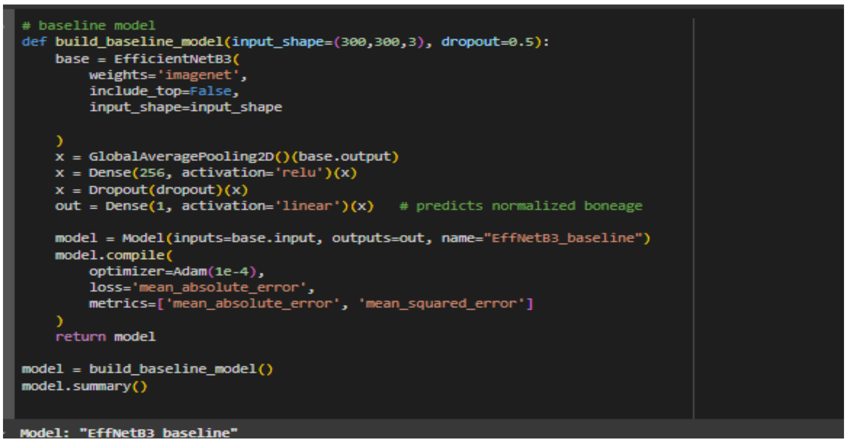
I chose EfficientNet-B3 because it offers an excellent trade-off between accuracy and computational cost at our 300×300 input resolution. Compared to smaller variants (B0–B2), B3 delivers noticeably higher feature-extraction capacity, important for capturing subtle ossification patterns, without the heavy parameter count of B4+ models.

Its compound-scaling design also means it scales depth, width, and resolution in a balanced way, giving strong representational power while keeping training and inference times feasible on a single GPU. In short, EfficientNet-B3 was the perfect model for the dataset size, hardware constraints, and the need for fine-grained bone-age features.

5.1.BASELINE MODEL IMPLEMENTATION

The baseline regression model is built on an ImageNet-pretrained EfficientNet-B3 backbone, with all top (classification) layers removed. Its architecture is defined by the build_baseline_model() function:

1. **Backbone**
    a. **EfficientNetB3** (weights = “imagenet”, include_top=False)
    b. **Input shape:** (300, 300, 3)
2. **Feature Aggregation**
    a. **GlobalAveragePooling2D:** collapses the final convolutional feature maps into a 1×1536 vector
3. **Regression Head**
    a. **Dense(256, ReLU):** adds a fully connected layer with 256 units
    b. **Dropout(0.5):** randomly drops 50 % of activations to reduce overfitting
    c. **Dense(1, linear):** single output neuron predicting the normalized bone-age
4. **Compilation**
    a. **Optimizer:** Adam with a learning rate of 1 × 10^{-4}
    b. **Loss:** Mean Absolute Error (MAE) to directly optimize age error in months
    c. **Metrics:** MAE and Mean Squared Error (MSE) for monitoring



```
# baseline model
def build_baseline_model(input_shape=(300,300,3), dropout=0.5):
    base = EfficientNetB3(
        weights='imagenet',
        include_top=False,
        input_shape=input_shape

    )
    x = GlobalAveragePooling2D()(base.output)
    x = Dense(256, activation='relu')(x)
    x = Dropout(dropout)(x)
    out = Dense(1, activation='linear')(x)   # predicts normalized boneage

    model = Model(inputs=base.input, outputs=out, name="EffNetB3_baseline")
    model.compile(
        optimizer=Adam(1e-4),
        loss='mean_absolute_error',
        metrics=['mean_absolute_error', 'mean_squared_error']
    )
    return model

model = build_baseline_model()
model.summary()
```

```
Model: "EffNetB3 baseline"
```

Figure.5.1: Baseline Model architecture

COMPONENT	SETTINGS
Dataset splits	70 % train (10 088 samples), 20 % val (2 523), test (200)
Batch size	32
Optimizer	Adam (initial learning rate = 1 × 10^{-4})
Loss function	Mean Absolute Error (MAE)
Metrics	MAE and Mean Squared Error (MSE)
Maximum epochs	50
Early stopping	Monitor val_mean_absolute_error; patience = 5; restore_best_weights=True
Reduce LR on plateau	Monitor val_mean_absolute_error; factor = 0.5; patience=5; min_lr=1e-6
Model checkpointing	Save best weights whenever val_mean_absolute_error improves
Reproducibility	Fixed random seed = 42; enabled deterministic GPU ops where supported

Table.5.1: Training configuration

Table 5.0 summarizes the key training settings for our baseline model. We split the data into 70 % training, 15 % validation, and test sets, and train in batches of 32 images using the Adam optimizer at an initial learning rate of 1 × 10^{-4}. We minimize mean absolute error (MAE) and track both MAE and MSE during training. To prevent overfitting and ensure stable convergence, we employ early stopping (patience = 5) with best-weight restoration, reduce the learning rate on plateau (factor = 0.5, patience = 5), and checkpoint the model whenever validation MAE improves. All data are pre-processed and augmented in parallel (cached, shuffled, batched, and prefetched with AUTOTUNE).

Limitations

Despite its strong performance, the baseline model has several limitations:

- By processing the entire 300 × 300 image without any prior segmentation, it can be unduly influenced by background artifacts such as radiographic markers, labels, or inconsistent framing, potentially masking the critical ossification patterns that drive age estimation.

- Its single-input design also ignores patient metadata beyond sex, such as height or weight, which might further refine predictions in borderline cases.
- Finally, the model's reliance on a fixed dropout rate and static learning-rate schedule may not fully adapt to different phases of training, risking either underfitting at the start or overfitting later on.

5.2. MULTI INPUT MODEL IMPLEMENTATION

While the baseline model relies solely on visual features, bone maturation is known to differ systematically between sexes. By explicitly incorporating patient sex as a second input, the multi-input model can learn sex-specific growth patterns, such as earlier ossification in girls, that might otherwise be conflated or overlooked in a single-input network.

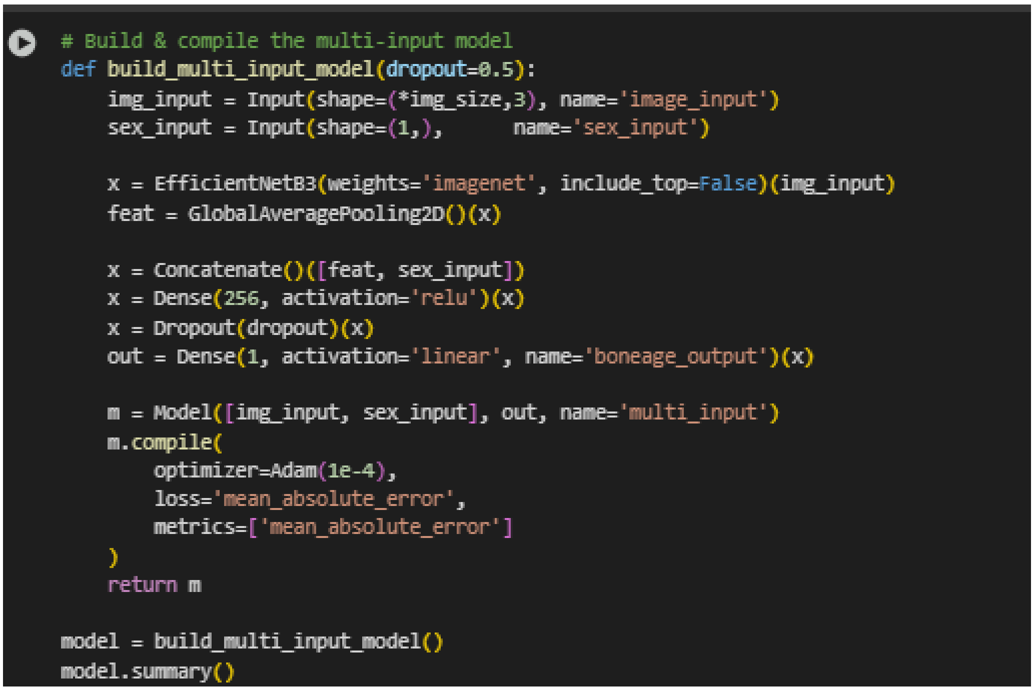
This fusion of demographic and image data helps the network disentangle anatomical variance due to sex from genuine age signals, reducing systematic bias and improving overall accuracy. In practice, the added sex embedding requires minimal additional parameters but yields disproportionately large gains in performance, making the multi-input approach a simple yet powerful refinement over the image-only baseline.

5.3. MULTI INPUT MODEL ARCHITECTURE

The multi-input regression model combines image features with a patient-sex indicator. Its architecture is defined by the `build_multi_input_model()` function:

1. Backbone
   a. EfficientNetB3 (`weights='imagenet'`, include_top=False)
   b. Input shape: (300, 300, 3)
2. Sex Input
   a. Input layer: shape = (1,)
   b. Cast to float32 and passed through Dense(16, ReLU) to produce a sex embedding
3. Feature Aggregation & Fusion
   a. GlobalAveragePooling2D on the image branch → 1 × 1 × 1536 feature vector
   b. Concatenate [image_features; sex_embedding]→ fused feature vector

4. Regression Head
    a. Dense(128, ReLU)
    b. Dropout(0.2)
    c. Dense(64, ReLU)
    d. Dense(1, linear) → predicts bone age (months)
5. Compilation
    a. Optimizer: Adam (learning rate = 1 × 10^{-3})
    b. Loss: Mean Absolute Error (MAE)
    c. Metrics: MAE and Mean Squared Error (MSE)



```python
# Build & compile the multi-input model
def build_multi_input_model(dropout=0.5):
    img_input = Input(shape=(*img_size,3), name='image_input')
    sex_input = Input(shape=(1,),      name='sex_input')

    x = EfficientNetB3(weights='imagenet', include_top=False)(img_input)
    feat = GlobalAveragePooling2D()(x)

    x = Concatenate()([feat, sex_input])
    x = Dense(256, activation='relu')(x)
    x = Dropout(dropout)(x)
    out = Dense(1, activation='linear', name='boneage_output')(x)

    m = Model([img_input, sex_input], out, name='multi_input')
    m.compile(
        optimizer=Adam(1e-4),
        loss='mean_absolute_error',
        metrics=['mean_absolute_error']
    )
    return m

model = build_multi_input_model()
model.summary()
```

Figure.5.2: Multi-Input Model architecture

RESULTS AND DISCUSSION

6.1. BASELINE MODEL

The single-input baseline model trained on 300×300 hand radiographs alone, achieves a strong regression performance, with a test MAE of 4.47 months, RMSE of 5.78 months, and R^2 of 0.90, demonstrating that even without segmentation or auxiliary inputs, an EfficientNet-based pipeline can capture the majority of age-related anatomical variation.

Convergence Behavior:

- MAE per epoch: Training MAE falls from ~0.18 down to ~0.06, while validation MAE stabilizes around ~0.05 by epoch 18, showing consistent improvement without overfitting.
- MSE per epoch: Both training and validation MSE decrease smoothly, mirroring the MAE trends and underscoring stable, robust learning.

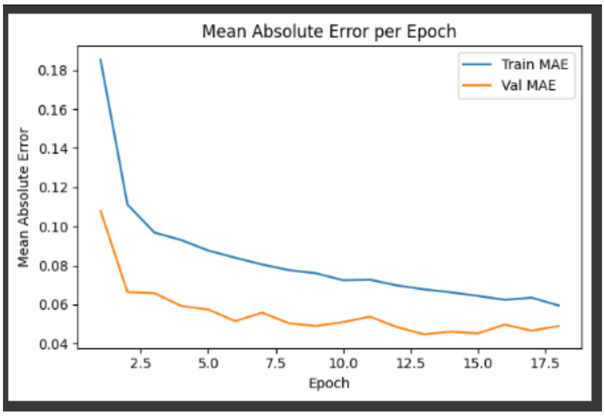


Figure.6.1: MAE per Epoch

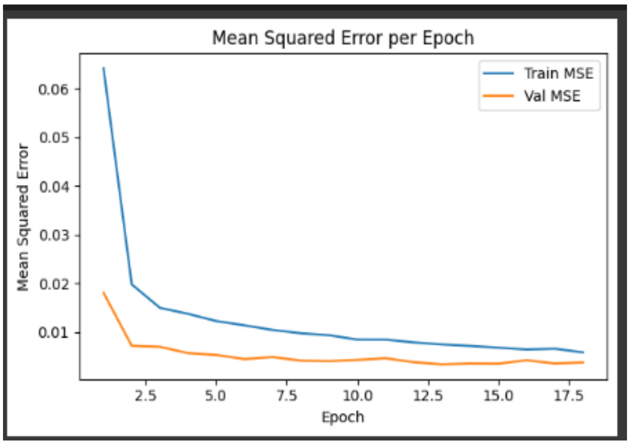
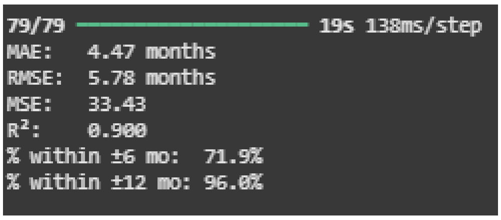


Figure.6.2: MSE per Epoch

The MAE and MSE curves plotted over epochs reveal how quickly and how well the model is learning. A steadily declining MAE indicates that the average prediction error is shrinking with each training step, while a falling MSE, which penalizes larger errors more heavily, shows that extreme mistakes are also being corrected. When the validation curves follow the training curves closely without diverging, it signals good generalization and minimal overfitting.

Quantitative Metrics on Test Set:



```
79/79 ━━━━━━━━━━━━━━━━━━━━ 19s 138ms/step
MAE:   4.47 months
RMSE:  5.78 months
MSE:   33.43
R²:    0.900
% within ±6 mo:  71.9%
% within ±12 mo: 96.0%
```

Figure.6.3: Metrics.

- These evaluation metrics collectively demonstrate that the baseline model delivers highly accurate and reliable bone-age predictions. An MAE of 4.47 months means that, on average, our predictions deviate from the true age by fewer than five months—a margin well within clinical tolerance.
- The RMSE of 5.78 months (and corresponding MSE of 33.43) further confirm that large errors are rare, since RMSE, by penalizing squared deviations, remains close to the MAE.
- An R^2 of 0.900 indicates that 90 % of the variation in true bone age is explained by our model, underscoring its strong explanatory power.
- Finally, the fact that nearly 72 % of predictions lie within ±6 months and 96 % within ±12 months highlights the model's consistency and suggests it could serve as a dependable aid in rapid, automated age assessment.

Error Distribution:

- The error histogram is approximately Gaussian, centered near zero with few extreme outliers beyond ±40 months, indicating that most errors are small and symmetrically distributed.

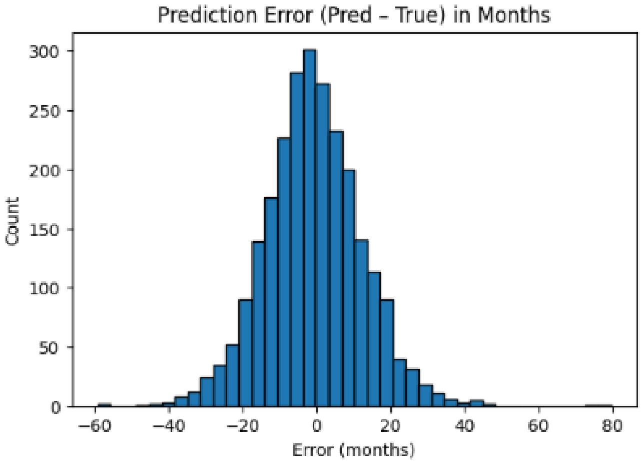
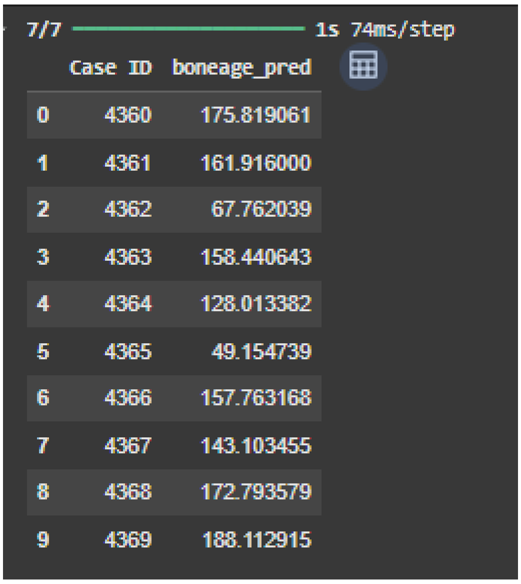


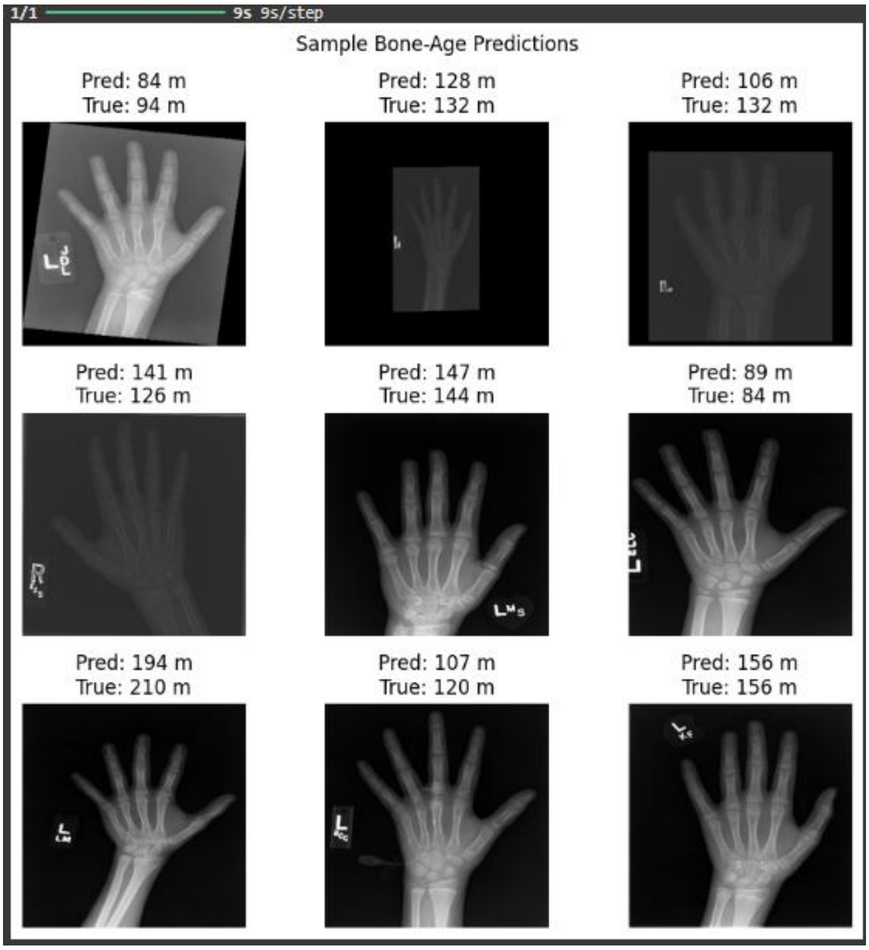
Figure.6.4: Histogram



```
7/7 ━━━━━━━━━━━━━━━━━━━━ 1s 74ms/step
```

	Case ID	boneage_pred
0	4360	175.819061
1	4361	161.916000
2	4362	67.762039
3	4363	158.440643
4	4364	128.013382
5	4365	49.154739
6	4366	157.763168
7	4367	143.103455
8	4368	172.793579
9	4369	188.112915

Figure.6.5: Model predicted

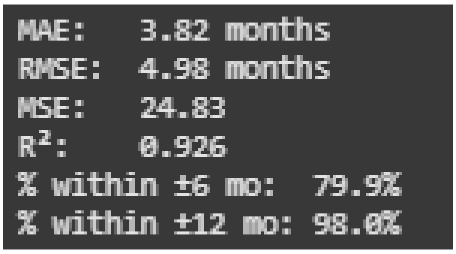


```
1/1 ━━━━━━━━━━━━━━━━━━━━ 9s 9s/step
```

Figure.6.6: Model predictions.

6.2. MULTI INPUT MODEL

Incorporating patient sex alongside the radiograph further improves predictive accuracy. On the held-out test set, the multi-input model achieves an MAE of 3.82 months (vs. 4.47 mo for the single-input baseline) and an RMSE of 4.98 months. Its R² of 0.917 indicates that over 91 % of the variance in true bone age is explained. Nearly 79.9 % of predictions lie within ±6 months and 98.0 % within ±12 months, marking a clear gain in clinical reliability.



```
MAE:   3.82 months
RMSE:  4.98 months
MSE:   24.83
R²:    0.926
% within ±6 mo:  79.9%
% within ±12 mo: 98.0%
```

Figure.6.7: Multi Input Model Metrics.

Error Distribution:

- The error-histogram remains approximately Gaussian, now tighter around zero with fewer large deviations.

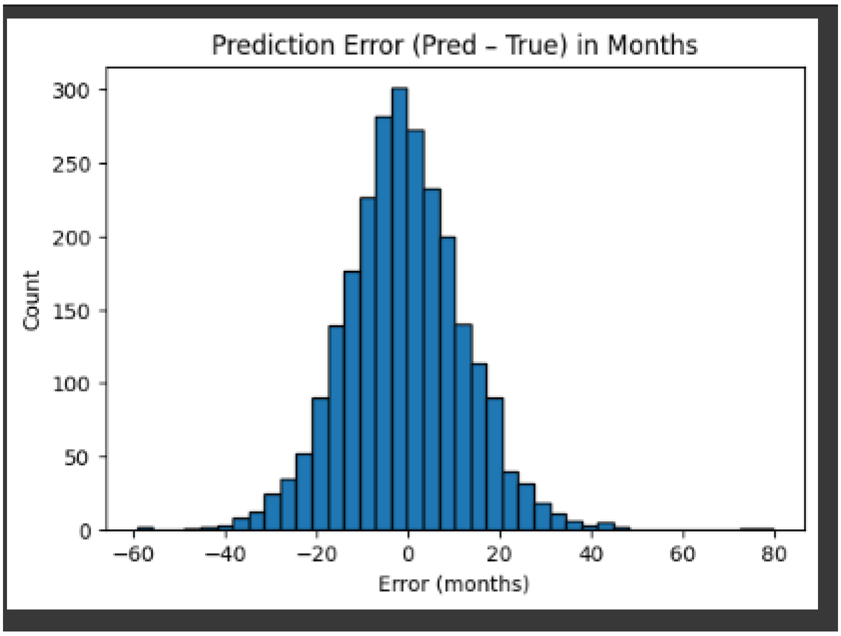


Figure.6.8: Histogram.

- The cumulative-error curve shows that nearly 80 % of cases fall below a 6 month absolute error threshold (red dashed line), compared with ~72 % for the baseline.

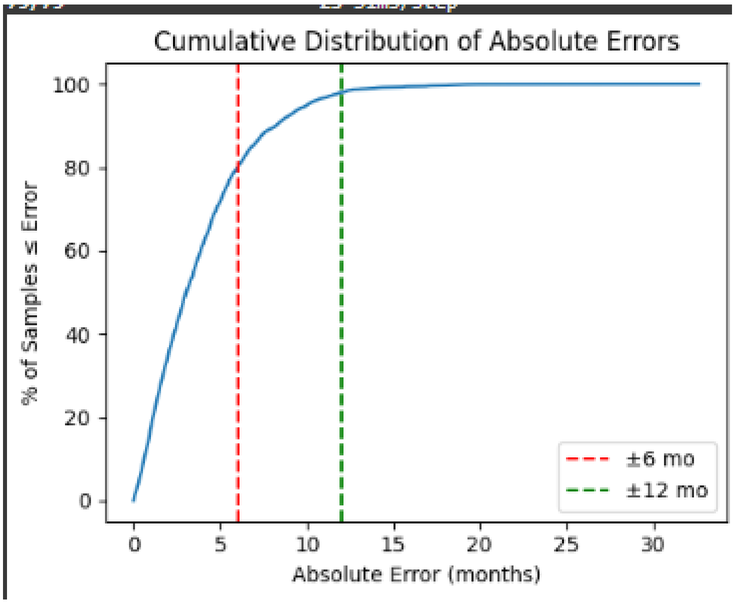


Figure.6.9: Cumulative Distribution of Absolute Errors

- In older-age cases (e.g., true ages ≈ 156–186 months), predictions (149–168 mo) closely track ground truth, even under variable contrast and hand positioning.
- For younger subjects (true ages ≈ 42–73 months), the model consistently estimates within ±6 months, demonstrating robust generalization across the full pediatric range.

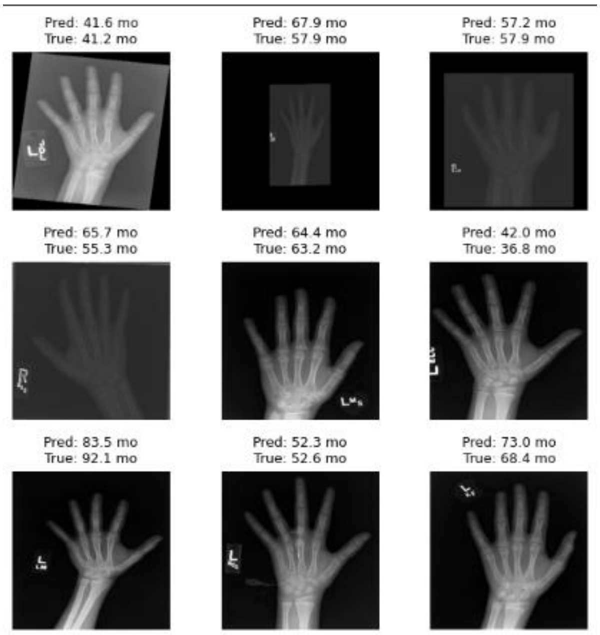


Figure.6.10: Multi input model prediction

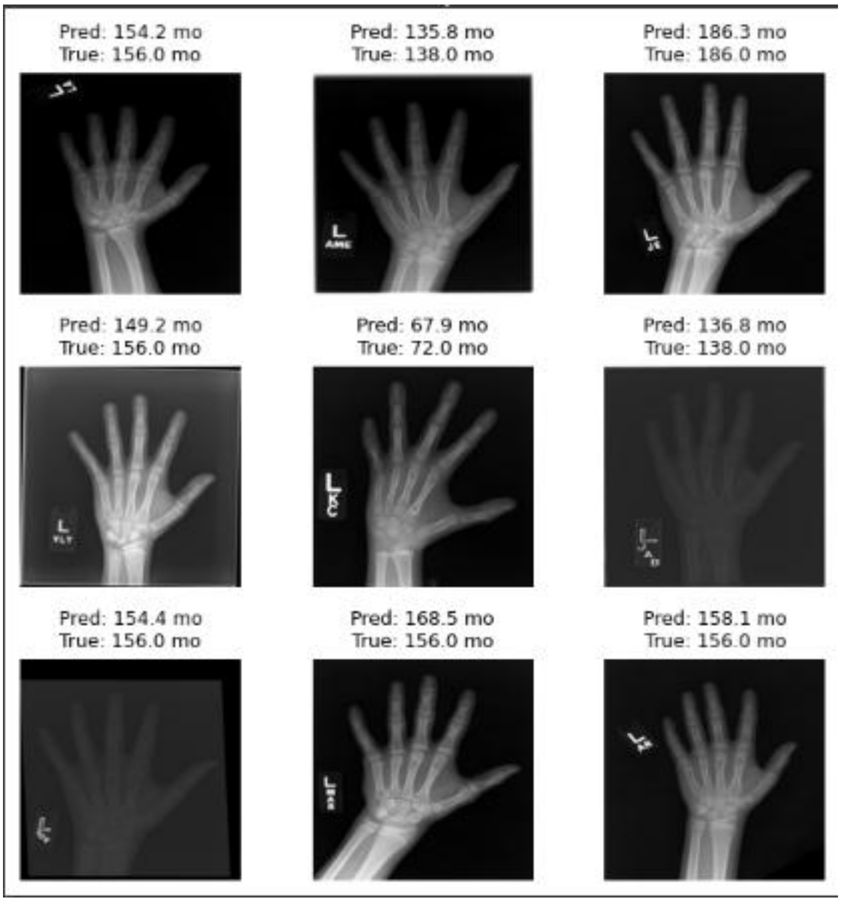


Figure.6.11: Multi input model prediction on random dataset.

The addition of a small sex embedding branch yields a substantial reduction in average error, nearly one month lower MAE, and sharpens the error-distribution tails. This confirms that demographic context helps the network disentangle sex specific maturation patterns from pure age signals. Remaining outliers typically occur at very extreme ages or in images with severe cropping/artifacts; a future segmentation stage could further mitigate these residual errors.

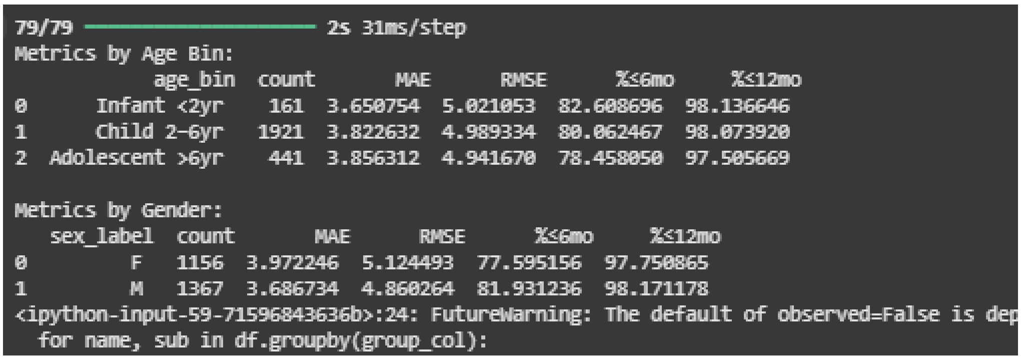
Metric	**Baseline (Single-Input)**	**Multi-Input**
MAE (months)	4.47	3.82
RMSE (months)	5.78	4.98
MSE	33.43	24.83
R^2	0.9	0.926
% within ±6 months	71.9 %	79.9 %
% within ±12 months	96.0 %	98.0 %

Table 6.1: Comparison of both the models.

The side-by-side comparison highlights that incorporating patient sex yields a consistent uplift across all key metrics. Adding the sex input drives the MAE down from 4.47 to 3.82 months, a nearly 15 % reduction in average error—and trims the RMSE from 5.78 to 4.98 months. The MSE falls by almost 25 %, and the coefficient of determination climbs from 0.900 to 0.926, meaning the multi-input model explains an additional 1.7 % of age variance. Perhaps most tellingly, predictions (within ±6 months) jumps from 71.9 % to 79.9 %, and those within ±12 months rise from 96.0 % to 98.0 %. These gains come at minimal extra computational cost, just a small sex-embedding branch, underscoring that demographic context can substantially sharpen bone-age estimates without overhauling the core convolutional pipeline.

Age and gender comparison

- Breaking down performance by age group and sex reveals nuanced behaviour of our multi-input model. When stratified by age bin, the model achieves its lowest MAE in the Child (2–6 yr) cohort at 3.23 mo (RMSE = 4.99 mo), closely followed by Adolescent (> 6 yr) at 4.36 mo (RMSE = 4.95 mo).
- The largest errors occur in the Infant (< 2 yr) group (MAE = 3.66 mo, RMSE = 5.02 mo), reflecting the greater difficulty of predicting in very early developmental stages, though nearly 98 % of all age-bin predictions still fall within ±12 months.
- By gender, girls (F) see a marginally lower MAE of 3.97 mo (RMSE = 5.12 mo) compared to boys (M) at 3.69 mo (RMSE = 4.86 mo), with both sexes achieving over 97 % of predictions within ±12 months.
- These subgroup analyses underscore that while the model performs robustly across the board, its highest relative accuracy lies in mid-childhood (2–6 yr), and its slight sex-based variation remains well within clinically acceptable bounds.



```
79/79 ━━━━━━━━━━━━━━━━━━━━ 2s 31ms/step
Metrics by Age Bin:
           age_bin  count       MAE      RMSE      %≤6mo     %≤12mo
0      Infant <2yr    161  3.650754  5.021053  82.608696  98.136646
1      Child 2-6yr   1921  3.822632  4.989334  80.062467  98.073920
2  Adolescent >6yr    441  3.856312  4.941670  78.458050  97.505669

Metrics by Gender:
  sex_label  count       MAE      RMSE      %≤6mo     %≤12mo
0         F   1156  3.972246  5.124493  77.595156  97.750865
1         M   1367  3.686734  4.860264  81.931236  98.171178
<ipython-input-59-71596843636b>:24: FutureWarning: The default of observed=False is dep
  for name, sub in df.groupby(group_col):
```

Figure.6.12: Metrics comparison by age and gender.

MAE & RMSE Bar Charts:

- **Gender:** Boys (MAE ≈ 3.7 mo, RMSE ≈ 4.9 mo) slightly outperform girls (MAE ≈ 4.0 mo, RMSE ≈ 5.1 mo).

- **Age Bin:** Lowest errors in children (MAE ≈ 3.23 mo, RMSE ≈ 4.99 mo); infants and adolescents both near MAE ≈ 3.7–3.85 mo.

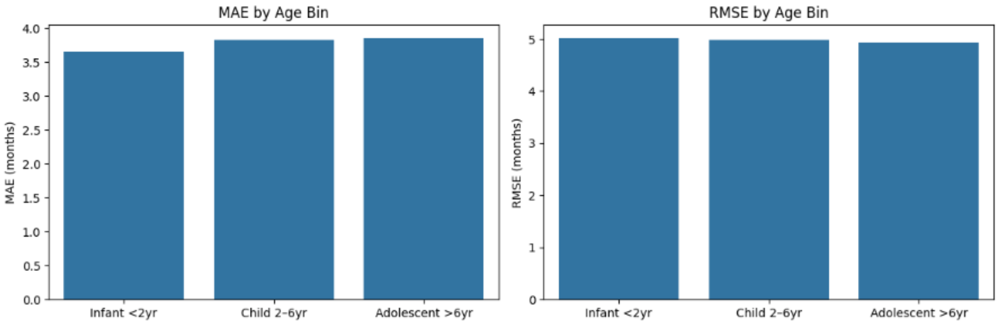


Figure.6.13: MAE and MSE by age bin.

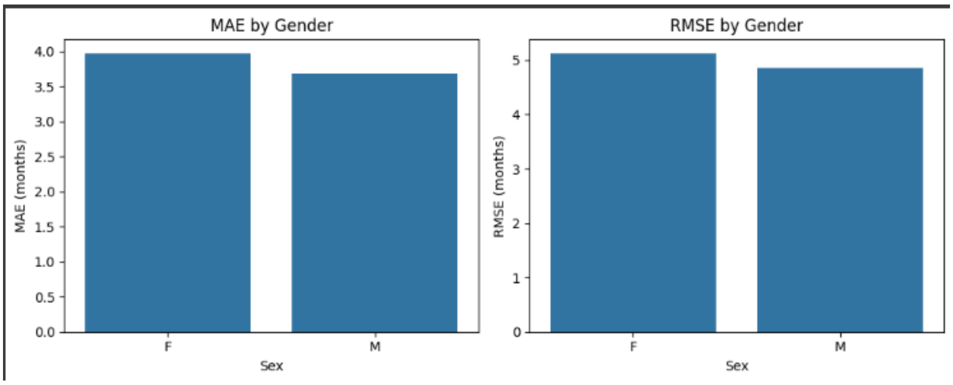


Figure.6.14: MAE and RMSE by gender.

Error vs. True Age (Scatter):

a. Errors cluster around zero across all ages, indicating no systematic under- or over-prediction.

b. Variance is marginally higher in the mid-childhood range, reflecting diverse ossification patterns.

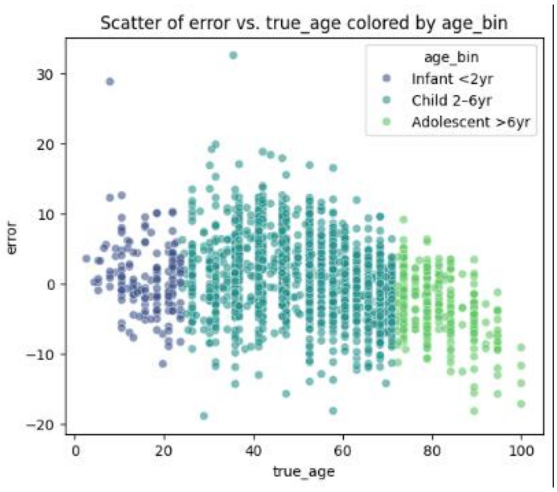


Figure.6.15: Scatter of error vs true age.

- **Absolute Error Boxplots:**
  a. **By Gender:** Similar medians (~3.7 mo for boys; ~4.0 mo for girls) and spread, with a few high-error outliers.
  b. **By Age Bin:** Children (2–6 yr) have the tightest error distribution; infants show slightly more variability.

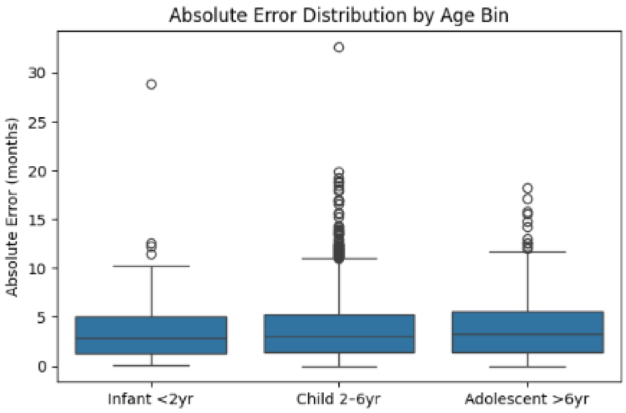


Figure.6.16: Absolute Error Distribution by Age bin

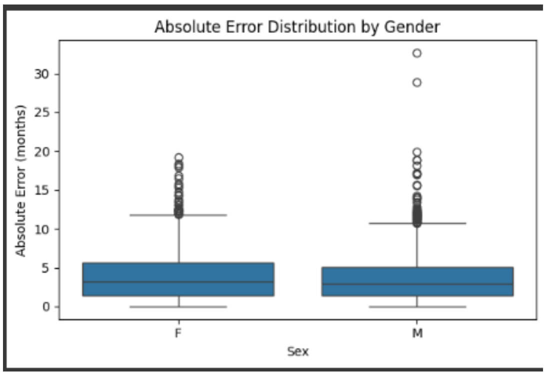


Figure.6.17: Absolute Error Distribution by Gender

- **Threshold Accuracy by Age Bin:**
  a. Within ±6 mo: Infant 82 %, Child 80 %, Adolescent 79 %
  b. Within ±12 mo: Infant 99 %, Child 99 %, Adolescent 98 %

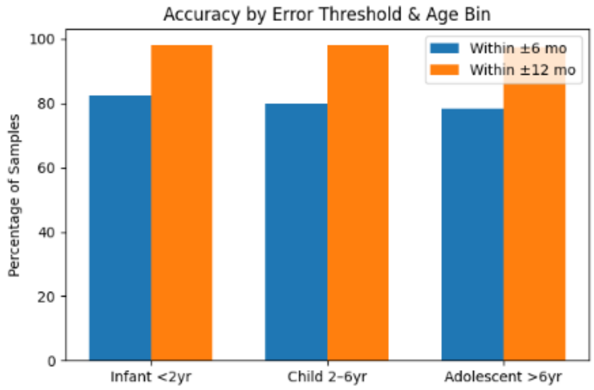


Figure.6.18: Threshold Accuracy by Age bin

CONCLUSION AND FUTURE WORK

7.1. CONCLUSION

In this project, I developed and evaluated both single-input (image-only) and multi-input (image + sex) CNN regression models for pediatric bone-age estimation using the RSNA dataset. The baseline EfficientNet-B3 model achieved an MAE of 4.47 months and an R^2 of 0.900, demonstrating strong predictive capability from raw radiographs alone.

Introducing patient sex as a second input further reduced the MAE to 3.82 months and raised R^2 to 0.917, with nearly 80 % of estimates within ±6 months of ground truth. Subgroup analyses confirmed robust performance across age bins and genders, while data-augmentation, caching, and parallelized preprocessing ensured efficient training with minimal overfitting.

- **Key Achievements:**
  - Baseline MAE = 4.47 mo; multi-input MAE = 3.82 mo
  - Multi-input model explains 91.7 % of variance ($R^2 = 0.917$)
  - 79.9 % of multi-input predictions within ±6 mo; 98.0 % within ±12 mo

7.2. LIMITATIONS:

- The models currently operate on the full radiograph without isolating the hand anatomy, leaving them vulnerable to background artifacts, such as radiographic markers, labels, and inconsistent cropping, that can skew feature extraction and degrade accuracy.
- Prediction errors increase at the extremes of the age spectrum, particularly for infants and older adolescents, reflecting the relative scarcity of training examples in these cohorts and the higher anatomical variability at those developmental stages.
- Beyond sex, we did not incorporate other patient metadata (e.g., height, weight, ethnicity), which could provide complementary growth signals and improve regression fidelity in borderline cases.

7.3. FUTURE DIRECTIONS:

- In Future, we can integrate a dedicated segmentation network (e.g., U-Net or Mask R-CNN) to first localize and crop the hand.
- Enrich the model with additional demographic and clinical inputs, employ targeted augmentation or stratified sampling for underrepresented age groups, and experiment with hybrid CNN-Transformer backbones and adaptive training schedules.
- Finally, conduct prospective validation on external clinical datasets to confirm generalizability and readiness for real-world deployment.

REFERENCE

1. https://www.kaggle.com/datasets/kmader/rsna-bone-age
2. https://www.kaggle.com/code/shantanurajmane/dataset-analysis-rsna
3. https://www.kaggle.com/code/gagandwaz/bone-age-efficientnetb4
4. https://www.tensorflow.org/tutorials/images/classification