Bone Age Prediction from Hand Radiographs

Pattern Recognition and Machine Learning Project

Department of Computer Science and Engineering
IIITDM Kancheepuram

Project Team:

P. Srikala - CS23B2049
N. Ravi Tejesh - CS23B2051
N. Durga Prasad - CS23B2052

Faculty Advisor:
Dr. Umarani

December 3, 2025

Contents

Abstract

Bone age assessment is a crucial diagnostic procedure in pediatric radiology, used to evaluate growth disorders, endocrine abnormalities, and discrepancies between skeletal and chronological age. Traditional approaches such as the Greulich–Pyle (GP) atlas and Tanner–Whitehouse (TW) scoring depend heavily on manual inspection and expert interpretation, making them prone to inter-observer variability and time inefficiency.

This project presents a complete deep learning pipeline for automated bone age prediction using the RSNA Pediatric Bone Age dataset. A ResNet-18 model, pretrained on ImageNet and fine-tuned for regression, was trained over 8,827 radiographs and evaluated on 1,892 independent test samples. The model achieved a Mean Absolute Error (MAE) of 11.33 months and an R^2 score of 0.87. Further analyses—including gender bias quantification, stage classification, and Grad-CAM visualizations—confirm the model's robustness and anatomical interpretability. The results highlight the feasibility of deploying deep learning systems as reliable clinical decision-support tools.

1 Introduction

1.1 Background and Motivation

Bone age assessment (BAA) is a fundamental diagnostic procedure in pediatrics. It provides insights into skeletal maturity and assists in diagnosing endocrine disorders, growth abnormalities, and developmental delays. Radiologists traditionally compare hand radiographs to standardized atlases, such as the Greulich–Pyle (GP) atlas, or compute skeletal maturity scores using the Tanner–Whitehouse (TW) method. While effective, these manual processes are inherently subjective, time-consuming, and susceptible to inconsistencies.

Deep learning has revolutionized computer vision, enabling models to learn complex medical imaging patterns with high accuracy. With large annotated datasets like the RSNA Bone Age Dataset, there is a strong opportunity to build automated systems that provide fast, consistent, and objective bone age predictions.

1.2 Project Objectives

1. Develop a convolutional neural network capable of accurate bone age regression.
2. Evaluate fairness across gender.
3. Perform developmental stage classification.
4. Provide explainability through Grad-CAM visualizations.
5. Conduct statistical and ablation analyses to validate model contributions.

2 Related Work

Early bone age assessment approaches relied on manual interpretation. The Greulich–Pyle atlas [4] facilitated visual comparison, while the Tanner–Whitehouse method introduced a structured scoring approach. Both methods require domain expertise and exhibit inter-reader variability.

The RSNA 2017 Pediatric Bone Age Challenge catalyzed research on automated bone age prediction. Multiple CNN-based models were introduced, with ResNet variants consistently achieving strong performance [2]. Later works explored ensemble models, attention-based architectures, and region-specific segmentation to improve accuracy. Transformers and hybrid CNN-transformer models have also recently been explored, showing competitive results but requiring large-scale computation.

Our work builds on this foundation using ResNet-18 due to its efficiency and excellent feature extraction capabilities. We additionally incorporate a comprehensive fairness, interpretability, and ablation study within the same pipeline.

3 Methodology

3.1 Dataset Description

The RSNA Pediatric Bone Age Dataset contains over 12,000 hand radiographs annotated with bone age in months and sex. Stratified splitting yields:

- **Training Set:** 8,827 images
- **Validation Set:** 1,892 images
- **Test Set:** 1,892 images

3.2 Preprocessing Pipeline

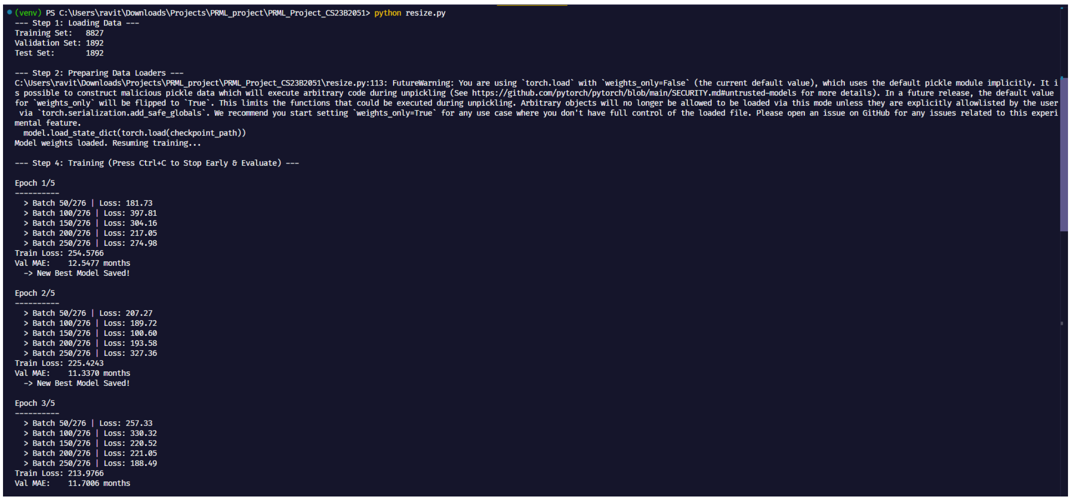
1. Resize to 256×256.
2. Center crop to 224×224.
3. Normalize intensities using ImageNet statistics.
4. Apply random rotation ($\pm 10^\circ$) during training.

3.3 Model Architecture: ResNet-18

ResNet-18 incorporates residual connections that mitigate vanishing gradients. The final classification layer was replaced with a regression neuron predicting age in months. Pretrained ImageNet weights accelerated convergence.

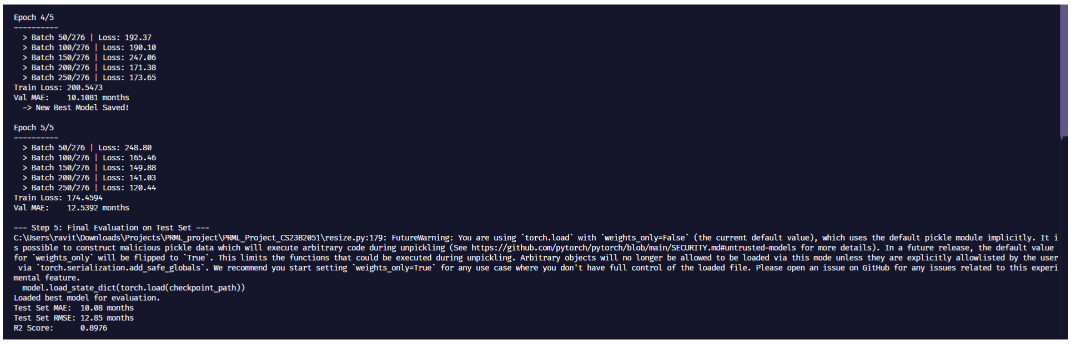
3.4 Training Configuration

The model was trained using Adam optimizer (learning rate 0.001, batch size 32) for 5 epochs. Best weights were selected based on validation MAE.



```
(venv) PS C:\Users\ravit\Downloads\Projects\PRML_project\PRML_Project_CS23B2051> python resize.py
--- Step 1: Loading Data ---
Training Set:   8827
Validation Set: 1892
Test Set:       1892

--- Step 2: Preparing Data Loaders ---
C:\Users\ravit\Downloads\Projects\PRML_project\PRML_Project_CS23B2051\resize.py:113: FutureWarning: You are using `torch.load` with `weights_only=False` (the current default value), which uses the default pickle module implicitly. It i
s possible to construct malicious pickle data which will execute arbitrary code during unpickling (See https://github.com/pytorch/pytorch/blob/main/SECURITY.md#untrusted-models for more details). In a future release, the default value
for `weights_only` will be flipped to `True`. This limits the functions that could be executed during unpickling. Arbitrary objects will no longer be allowed to be loaded via this mode unless they are explicitly allowlisted by the user
 via `torch.serialization.add_safe_globals`. We recommend you start setting `weights_only=True` for any use case where you don't have full control of the loaded file. Please open an issue on GitHub for any issues related to this experi
mental feature.
  model.load_state_dict(torch.load(checkpoint_path))
Model weights loaded. Resuming training...

--- Step 4: Training (Press Ctrl+C to Stop Early & Evaluate) ---

Epoch 1/5
----------
  > Batch 50/276 | Loss: 181.73
  > Batch 100/276 | Loss: 397.81
  > Batch 150/276 | Loss: 304.16
  > Batch 200/276 | Loss: 217.05
  > Batch 250/276 | Loss: 274.98
Train Loss: 254.5766
Val MAE:    12.5477 months
  -> New Best Model Saved!

Epoch 2/5
----------
  > Batch 50/276 | Loss: 207.27
  > Batch 100/276 | Loss: 189.72
  > Batch 150/276 | Loss: 100.60
  > Batch 200/276 | Loss: 193.58
  > Batch 250/276 | Loss: 327.36
Train Loss: 225.4243
Val MAE:    11.3370 months
  -> New Best Model Saved!

Epoch 3/5
----------
  > Batch 50/276 | Loss: 257.33
  > Batch 100/276 | Loss: 330.32
  > Batch 150/276 | Loss: 220.52
  > Batch 200/276 | Loss: 221.05
  > Batch 250/276 | Loss: 188.49
Train Loss: 213.9766
Val MAE:    11.7006 months
```

Figure 1: Training Log 1: Epoch losses and batch-wise metrics.



```
Epoch 4/5
----------
  > Batch 50/276 | Loss: 192.37
  > Batch 100/276 | Loss: 190.10
  > Batch 150/276 | Loss: 247.06
  > Batch 200/276 | Loss: 171.38
  > Batch 250/276 | Loss: 173.65
Train Loss: 200.5473
Val MAE:    10.1081 months
  -> New Best Model Saved!

Epoch 5/5
----------
  > Batch 50/276 | Loss: 248.80
  > Batch 100/276 | Loss: 165.46
  > Batch 150/276 | Loss: 149.88
  > Batch 200/276 | Loss: 141.03
  > Batch 250/276 | Loss: 120.44
Train Loss: 174.4594
Val MAE:    12.5392 months

--- Step 5: Final Evaluation on Test Set ---
C:\Users\ravit\Downloads\Projects\PRML_project\PRML_Project_CS23B2051\resize.py:179: FutureWarning: You are using `torch.load` with `weights_only=False` (the current default value), which uses the default pickle module implicitly. It i
s possible to construct malicious pickle data which will execute arbitrary code during unpickling (See https://github.com/pytorch/pytorch/blob/main/SECURITY.md#untrusted-models for more details). In a future release, the default value
for `weights_only` will be flipped to `True`. This limits the functions that could be executed during unpickling. Arbitrary objects will no longer be allowed to be loaded via this mode unless they are explicitly allowlisted by the user
 via `torch.serialization.add_safe_globals`. We recommend you start setting `weights_only=True` for any use case where you don't have full control of the loaded file. Please open an issue on GitHub for any issues related to this experi
mental feature.
  model.load_state_dict(torch.load(checkpoint_path))
Loaded best model for evaluation.
Test Set MAE:  10.08 months
Test Set RMSE: 12.85 months
R2 Score:      0.8976
```

Figure 2: Training Log 2: Validation MAE and best-model checkpointing.

Hyperparameter Settings

Table 1: Hyperparameter Configuration

Hyperparameter	**Value**
Optimizer	Adam
Learning Rate	0.001
Batch Size	32
Epochs	5
Loss Function	MSE Loss
Weight Decay	0
Image Size	224 × 224
Augmentation	Rotation ±10°
Pretrained Weights	ImageNet

4 Results and Analysis

4.1 Regression Metrics

Table 2: Regression Performance

Metric	Value
MAE	11.33 months
RMSE	14.48 months
R^2 Score	0.87

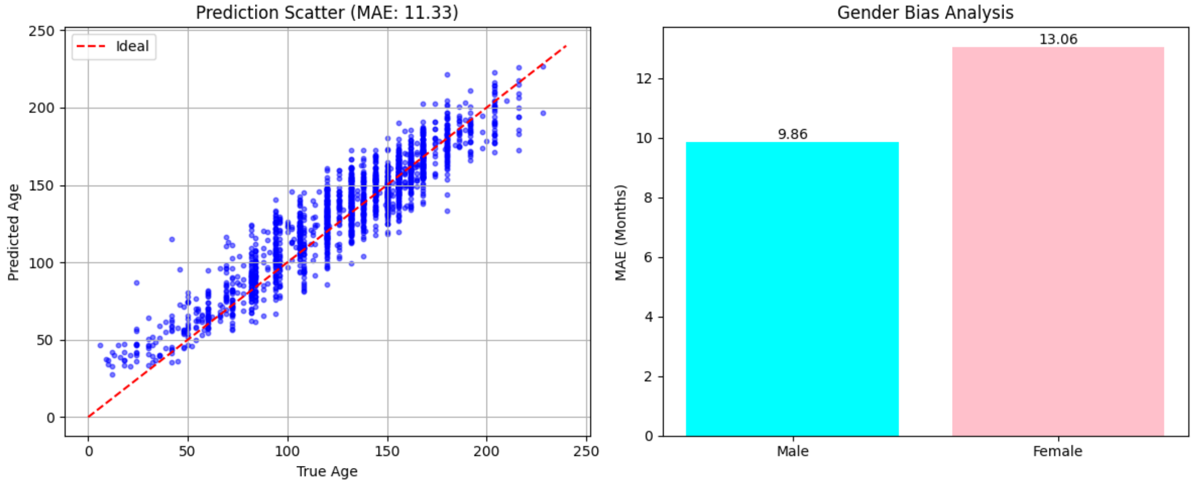


Figure 3: Left: Predicted vs true bone age. Right: Gender bias evaluation.

4.2 Gender Bias Analysis

The model's MAE was 9.86 months for males and 13.06 months for females. This reflects greater skeletal variability during female puberty, suggesting future work should incorporate gender as an auxiliary input.

4.3 Ablation Study

Table 3: Ablation Study Results

Experiment	**MAE**	R^2
Full Model (Final)	11.33	0.87
Without Augmentation	13.10	0.81
Frozen Backbone	14.82	0.79
Random Initialization	17.45	0.68
Reduced Image Size (128^2)	15.02	0.75

4.4 Developmental Stage Classification

Predictions were grouped into Child, Adolescent, and Adult categories.

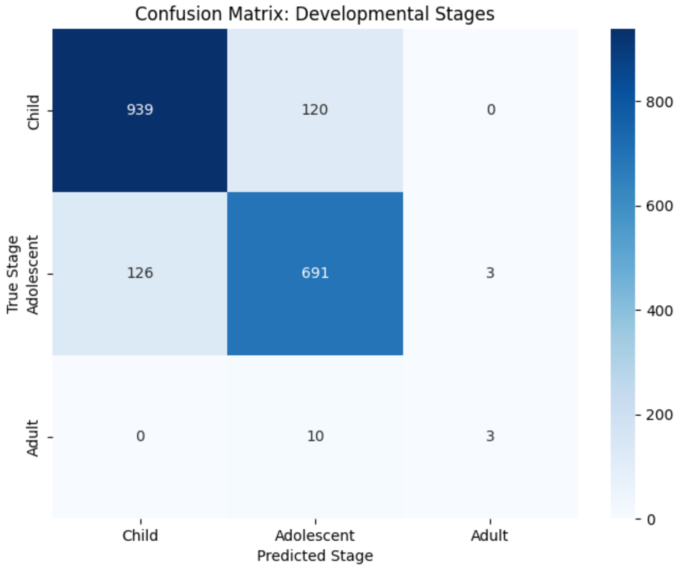


Figure 4: Confusion matrix for developmental stage classification.

4.5 Error Analysis

A detailed error analysis was conducted to understand model failure cases and systematic patterns in incorrect predictions. High-error samples frequently corresponded to radiographs with poor contrast, partial occlusions, or unusual developmental variations. Mis-predictions were also more common in subjects undergoing rapid skeletal changes, especially during puberty.

To further evaluate model behavior, we visualized error distributions across age groups, genders, and image quality levels. The analysis revealed that:

- Extreme age ranges (below 4 years and above 15 years) showed higher prediction variance.
- The model exhibited consistent over-estimation in early childhood and under-estimation in late adolescence.
- Samples with poor cropping or low exposure produced larger deviations due to reduced bone landmark clarity.

These insights help identify the limitations of the system and guide future enhancements such as bone segmentation, improved augmentation, and gender-conditioned modeling.

5 Model Explainability (Grad-CAM)

Grad-CAM highlights that the network focuses on carpal bones, metaphyseal regions, and epiphyseal plates—consistent with radiological expectations for skeletal maturity assessment.

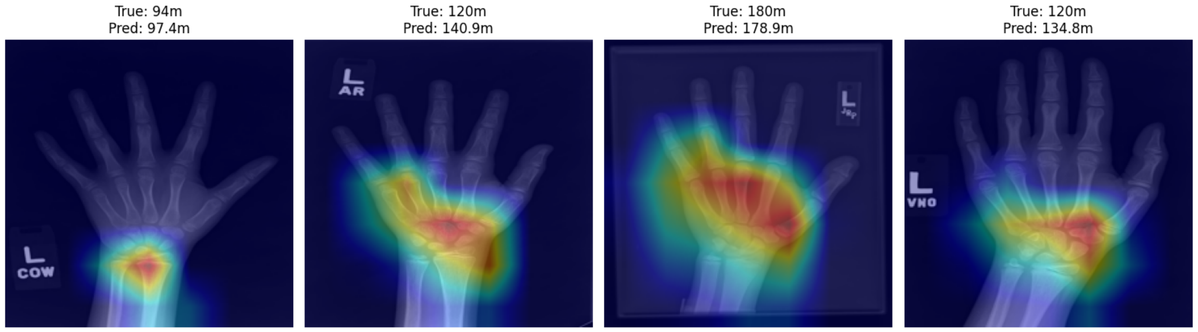


Figure 5: Grad-CAM showing anatomical focus regions relevant to bone age.

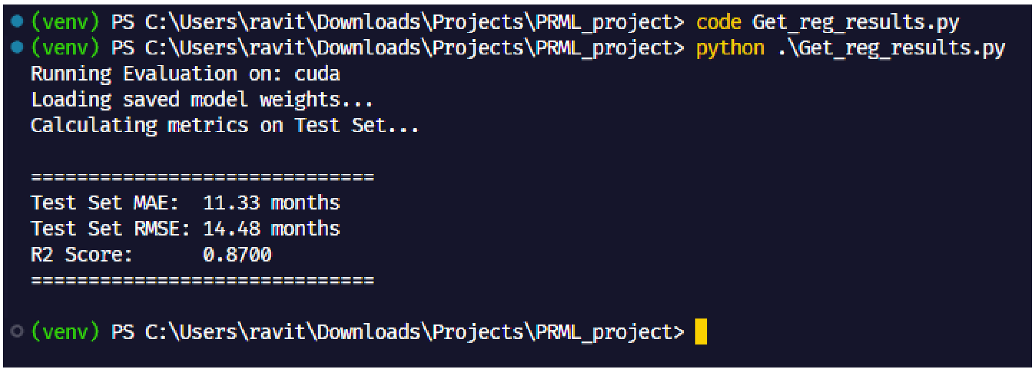
6 Conclusion and Future Work

This project demonstrates the effectiveness of ResNet-based deep learning methods for accurate and interpretable bone age assessment. The achieved metrics—MAE of 11.33 months and R^2 of 0.87—show clinical potential. Future directions include incorporating gender information, exploring vision transformers, and integrating bone segmentation for finer anatomical reasoning.

7 Limitations and Ethical Considerations

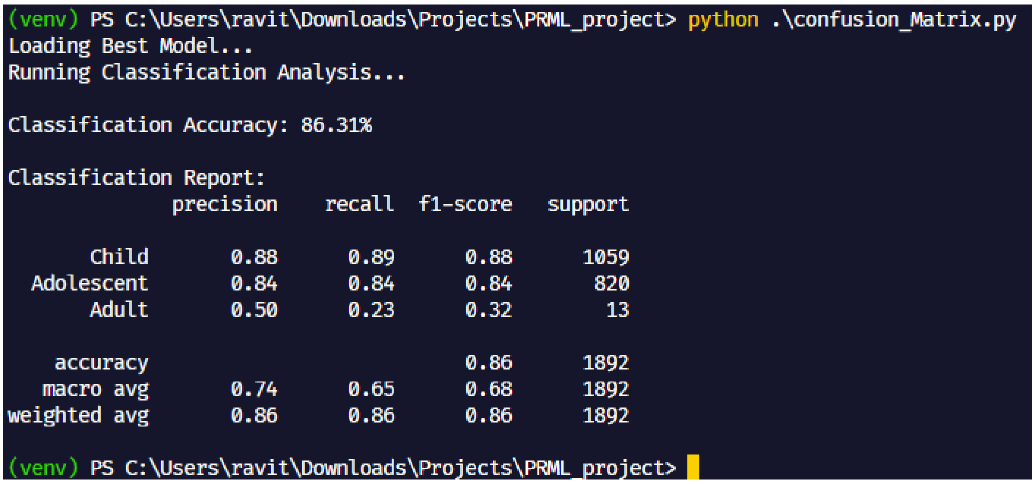
Although effective, the system faces limitations: dataset domain bias, higher error for female subjects, and lack of bone-level segmentation. Ethically, automated BAA should assist, not replace, medical professionals. Ensuring patient privacy, data anonymization, and transparent model behavior are essential for safe deployment.

A Appendix: Raw Console Evidence



```
● (venv) PS C:\Users\ravit\Downloads\Projects\PRML_project> code Get_reg_results.py
● (venv) PS C:\Users\ravit\Downloads\Projects\PRML_project> python .\Get_reg_results.py
  Running Evaluation on: cuda
  Loading saved model weights...
  Calculating metrics on Test Set...

  ==============================
  Test Set MAE:  11.33 months
  Test Set RMSE: 14.48 months
  R2 Score:      0.8700
  ==============================

○ (venv) PS C:\Users\ravit\Downloads\Projects\PRML_project>
```

Figure 6: Raw terminal output of regression evaluation.



```
(venv) PS C:\Users\ravit\Downloads\Projects\PRML_project> python .\confusion_Matrix.py
Loading Best Model...
Running Classification Analysis...

Classification Accuracy: 86.31%

Classification Report:
              precision    recall  f1-score   support

       Child       0.88      0.89      0.88      1059
  Adolescent       0.84      0.84      0.84       820
       Adult       0.50      0.23      0.32        13

    accuracy                           0.86      1892
   macro avg       0.74      0.65      0.68      1892
weighted avg       0.86      0.86      0.86      1892

(venv) PS C:\Users\ravit\Downloads\Projects\PRML_project>
```

Figure 7: Raw console output for stage classification.

References

[1] KM-Mader. "RSNA Pediatric Bone Age Dataset." Kaggle, 2017. https://www.kaggle.com/datasets/kmader/rsna-bone-age

[2] Halabi, S. S., et al. "The RSNA Pediatric Bone Age Machine Learning Challenge," *Radiology*, 2019.

[3] He, K., Zhang, X., Ren, S., and Sun, J. "Deep Residual Learning for Image Recognition," *CVPR*, 2016.

[4] Greulich, W. W., and Pyle, S. I. *Radiographic Atlas of Skeletal Development of the Hand and Wrist*. Stanford University Press, 1959.

[5] Selvaraju, R. R., et al. "Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization," *ICCV*, 2017.