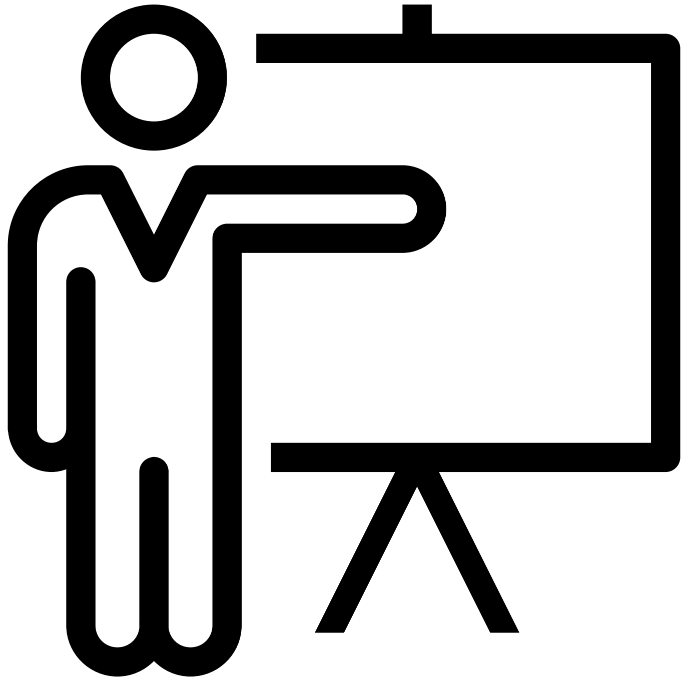
Bone Age Prediction

from hand radiographs

A case study of different
Deep Learning approaches

Author: Umut Berk CAKMAKCI



The bone age study can help evaluate how fast or slowly a child's skeleton is maturing, which can help doctors diagnose conditions that slow down or speed up physical growth and development.

Bone age help doctors monitor kids with:

- diseases such as growth hormone deficiency, hypothyroidism, precocious puberty, and adrenal gland disorders
- genetic growth disorder (Turner syndrome)
- orthopedic or orthodontic problems

Bone age can be used to predict:

- how much time a child will be growing
- when a child will start puberty
- what the child’s final height will be

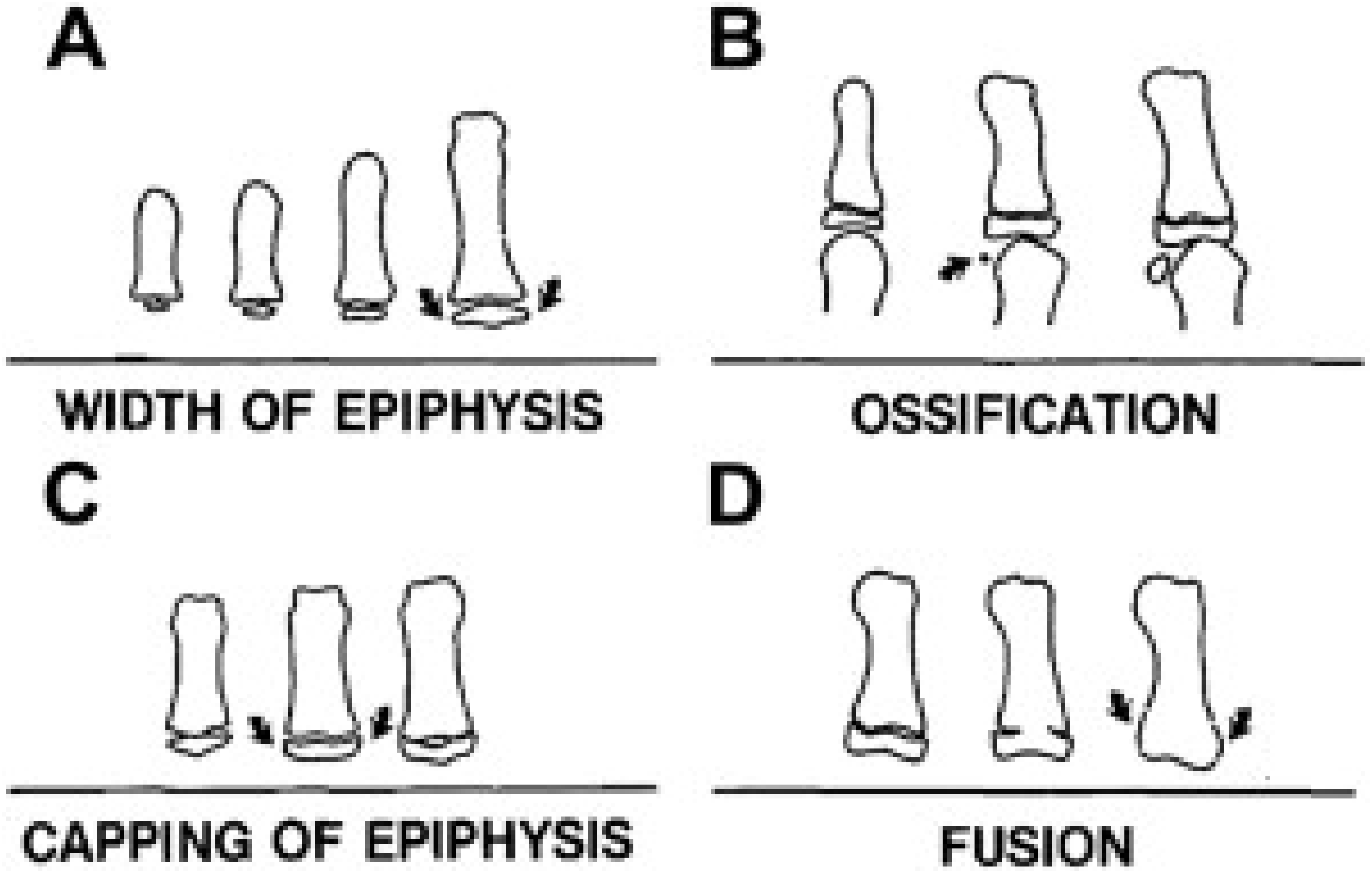
Manual assessment: Greulich-Pyle atlas and Turner-Whitehouse methods

Early automated system: BoneXpert (2009)

2017 RSNA Pediatric Bone Age ML Challange

Four parameters are examined:

- the width of the epiphysis,
- the appearance of ossification centers,
- the capping of the epiphysis,
- and the fusion of the epiphysis



RELATED WORKS

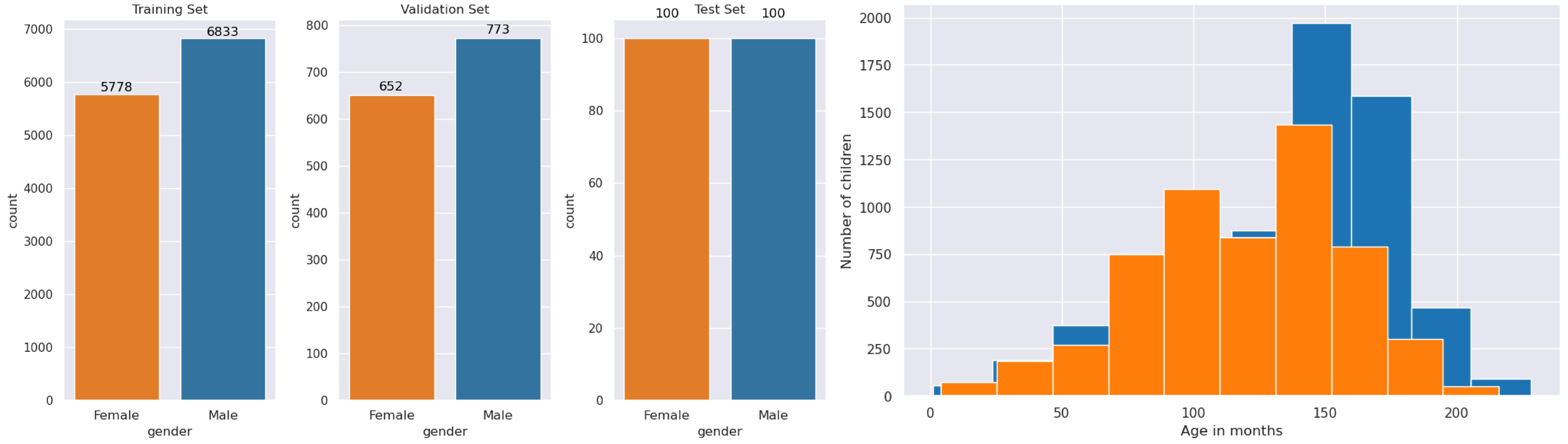
Halabi et. al. (2018)	Larson et. al. (2017)	Gonzalez et. al. (2020)
RSNA Challange winner model InceptionV3	regression-based deep CNN model ResNet50	attention-based deep CNN model SIMBA
average MAE of 4.2 months	average MAE of 6.24 months	average MAE of 6.34 months

DATASET DESCRIPTION

The Radiological Society of North America (RSNA) approved the curation and use of pediatric hand radiographs for the purposes of this Pediatric Bone Age Machine Learning Challenge

14236 X-ray images

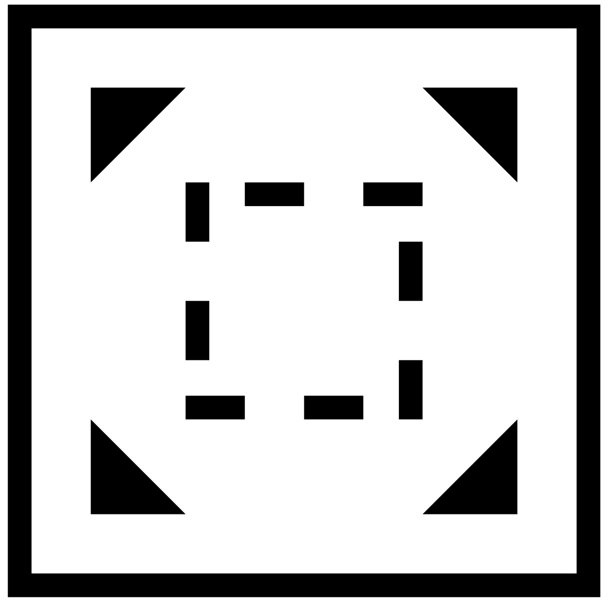
- Training Set: 12611
- Validation Set: 1425
- Test Set: 200



PROCESSING PIPELINE

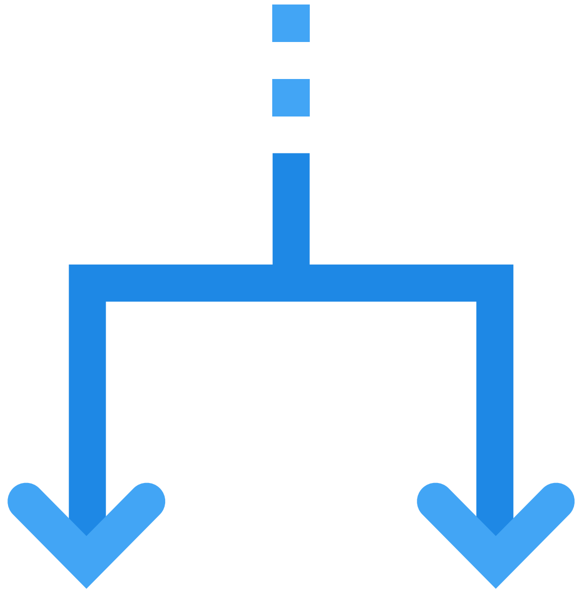
Dataset Cleaning

- Rename columns
- Convert true/false into 0/1 values
- Add z-scores

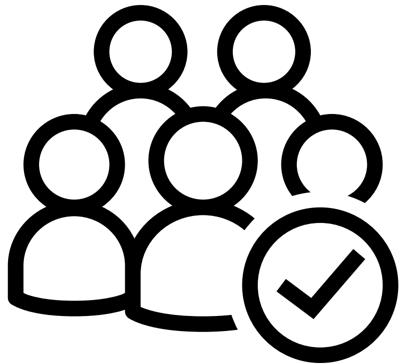


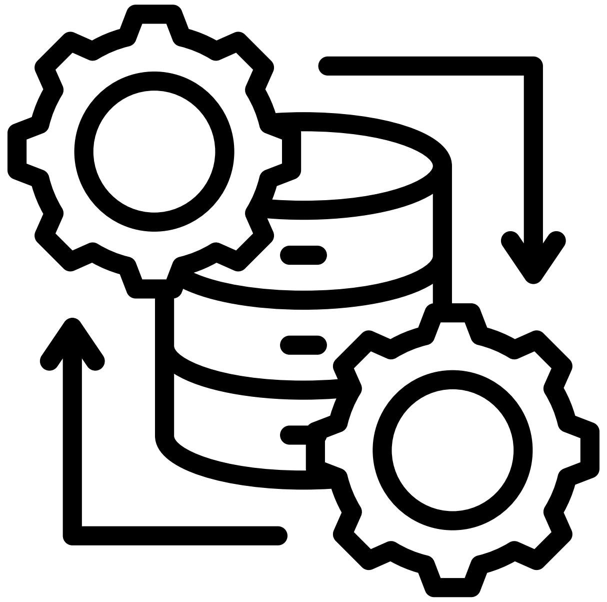
Resize

- 256x256
- 500x500



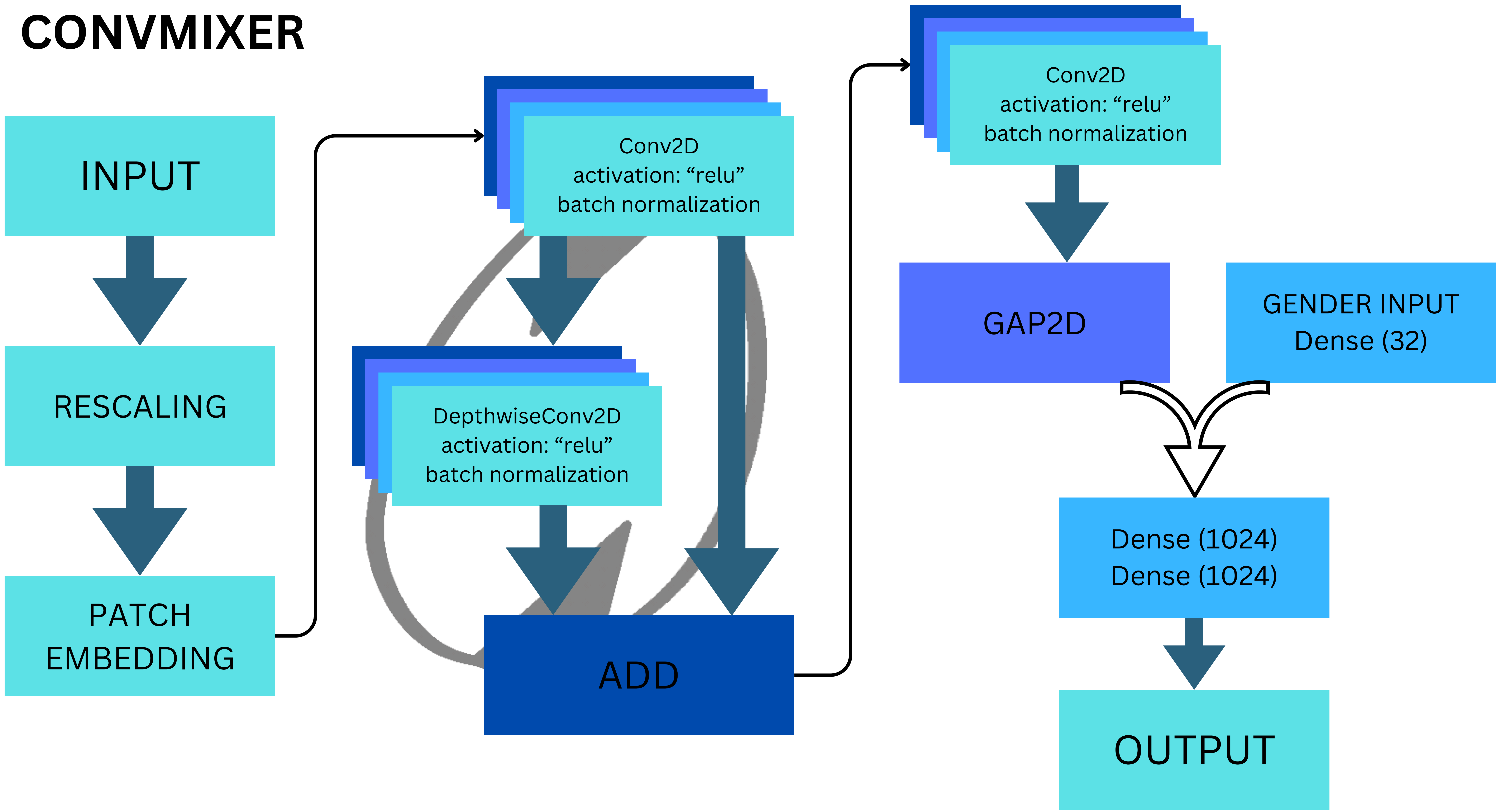
- Divide Batches
- Rescaling
- Flipping
- Rotating
- Shifting



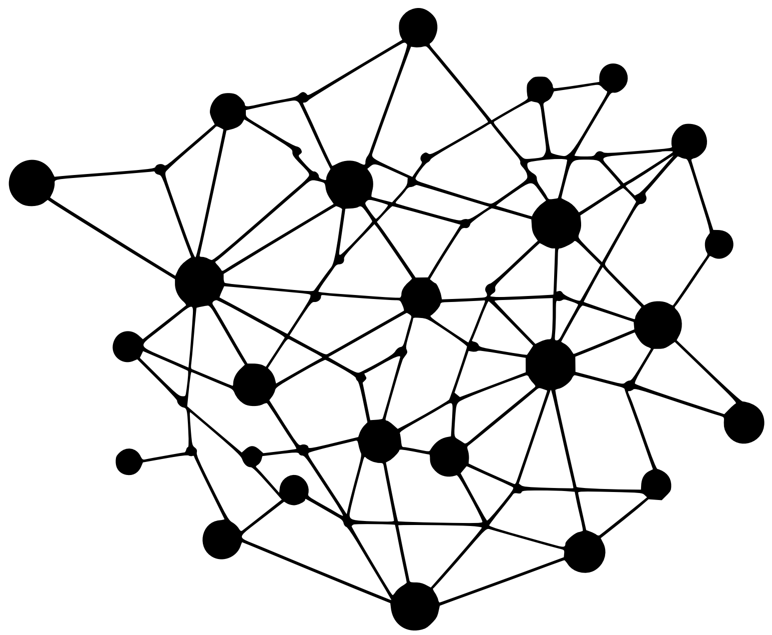


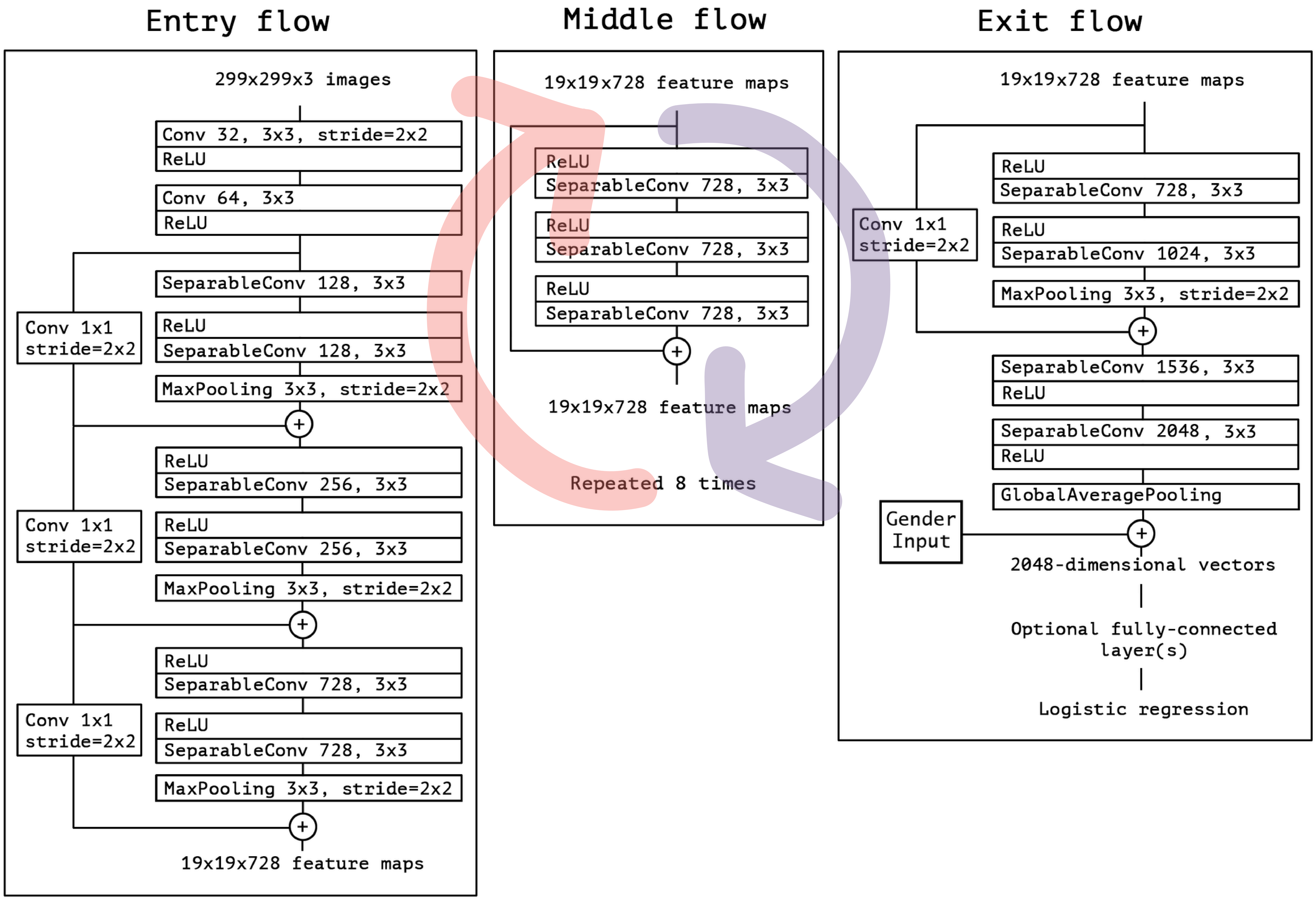
Dataset Pre-processing

- CLAHE
- preprocess_input


CONVMIXER
INPUT
RESCALING
PATCH EMBEDDING
Conv2D
activation: “relu”
batch normalization
DepthwiseConv2D
activation: “relu”
batch normalization
ADD
Conv2D
activation: “relu”
batch normalization
GAP2D
GENDER INPUT
Dense (32)
Dense (1024)
Dense (1024)
OUTPUT

XCEPTION




Entry flow
299x299x3 images
Conv 32, 3x3, stride=2x2
ReLU
Conv 64, 3x3
ReLU
SeparableConv 128, 3x3
Conv 1x1
stride=2x2
ReLU
SeparableConv 128, 3x3
MaxPooling 3x3, stride=2x2
ReLU
SeparableConv 256, 3x3
Conv 1x1
stride=2x2
ReLU
SeparableConv 256, 3x3
MaxPooling 3x3, stride=2x2
ReLU
SeparableConv 728, 3x3
Conv 1x1
stride=2x2
ReLU
SeparableConv 728, 3x3
MaxPooling 3x3, stride=2x2
19x19x728 feature maps
Middle flow
19x19x728 feature maps
ReLU
SeparableConv 728, 3x3
ReLU
SeparableConv 728, 3x3
ReLU
SeparableConv 728, 3x3
19x19x728 feature maps
Repeated 8 times
Exit flow
19x19x728 feature maps
ReLU
SeparableConv 728, 3x3
Conv 1x1
stride=2x2
ReLU
SeparableConv 1024, 3x3
MaxPooling 3x3, stride=2x2
SeparableConv 1536, 3x3
ReLU
SeparableConv 2048, 3x3
ReLU
GlobalAveragePooling
Gender
Input
2048-dimensional vectors
Optional fully-connected
layer(s)
Logistic regression

AdamW

OPTIMIZERS

- adapts the learning rate for each parameter during training
- incorporates momentum to move quickly in the direction of minima
- apply bias correction to adjusts the estimates of first and second moments
- Decouples weight decay from the gradient-based parameter updates
- tends to converge better on tasks with large models or datasets
- learning rate: 0.001 / 0.0001
  weight decay: 0.00001

‘Linear’ Activation Function

MSE and MAE losses

RLROP and Early Stop

HYPERPARAMETERS

CONVMIXER

Optimizer	AdamW
Learning Rate	0.001
Batch size	8
Dataset Splitting	Train+Val+Test
Image Size	256x256
Model Framework	ConvMixer + winner model
preprocessor	CLAHE

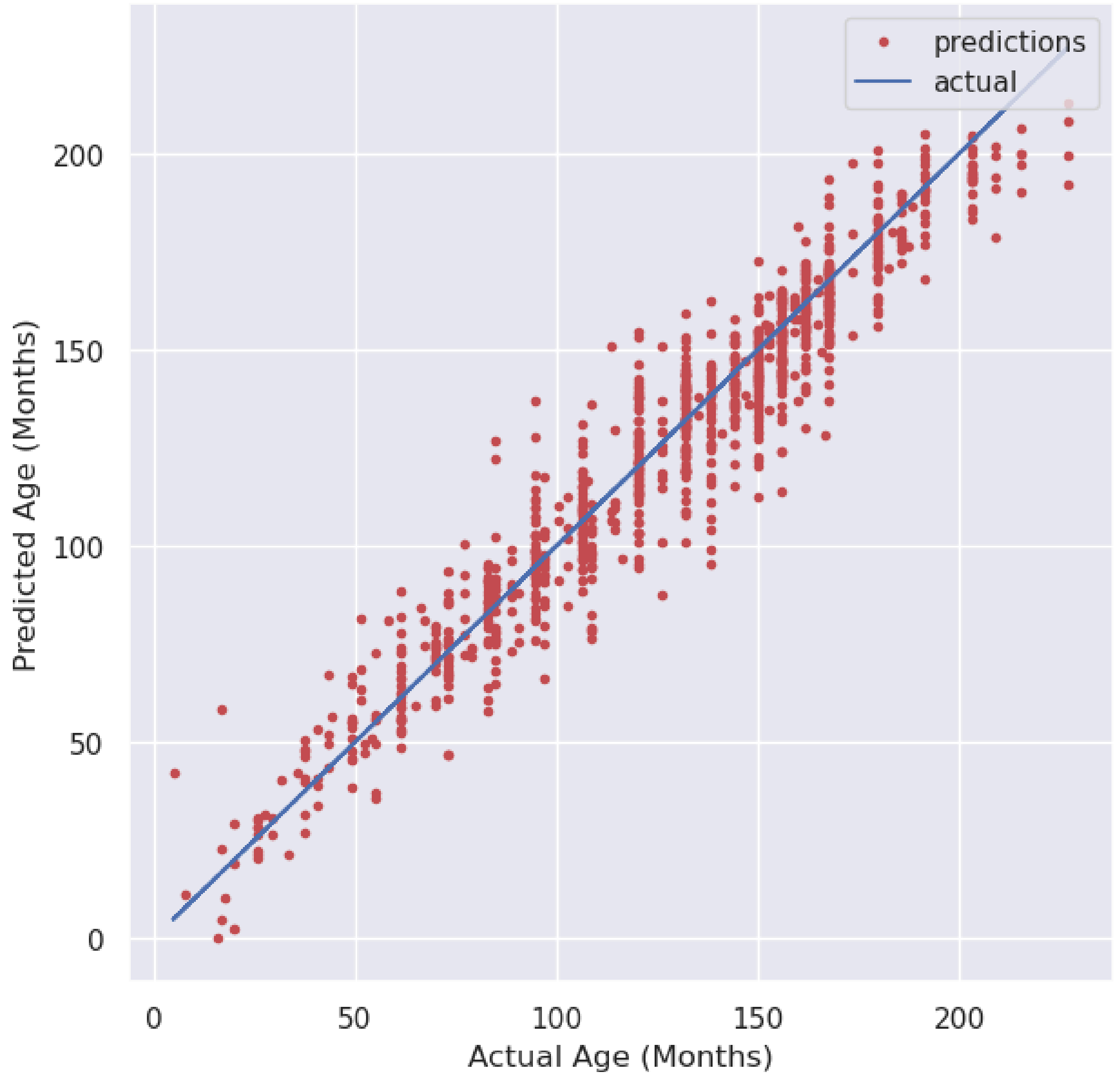
XCEPTION

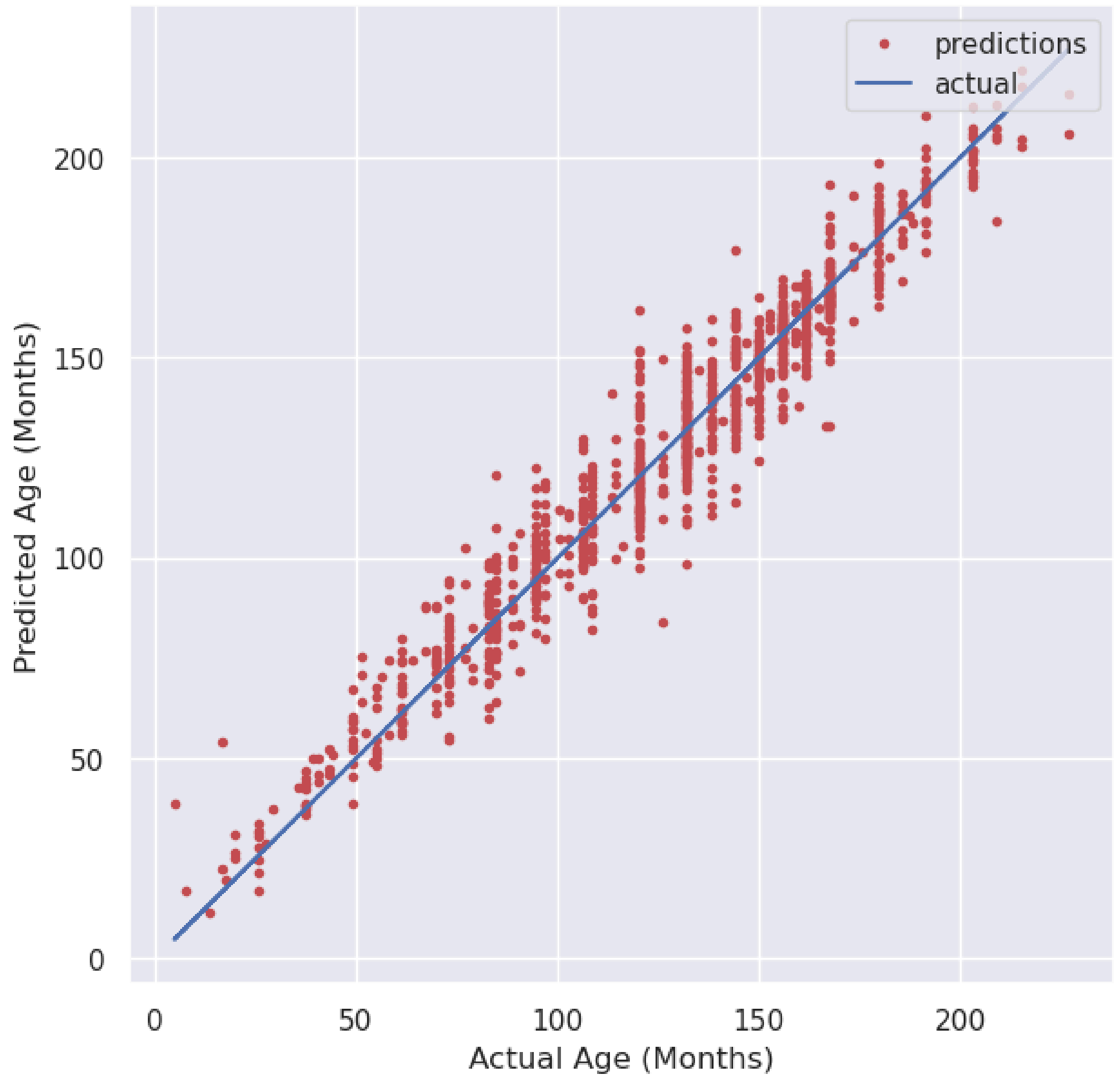
Optimizer	AdamW
Learning Rate	0.0001
Batch size	16
Dataset Splitting	Train+Val+Test
Image Size	500x500
Model Framework	Xception + winner model
preprocessor	preprocess_input

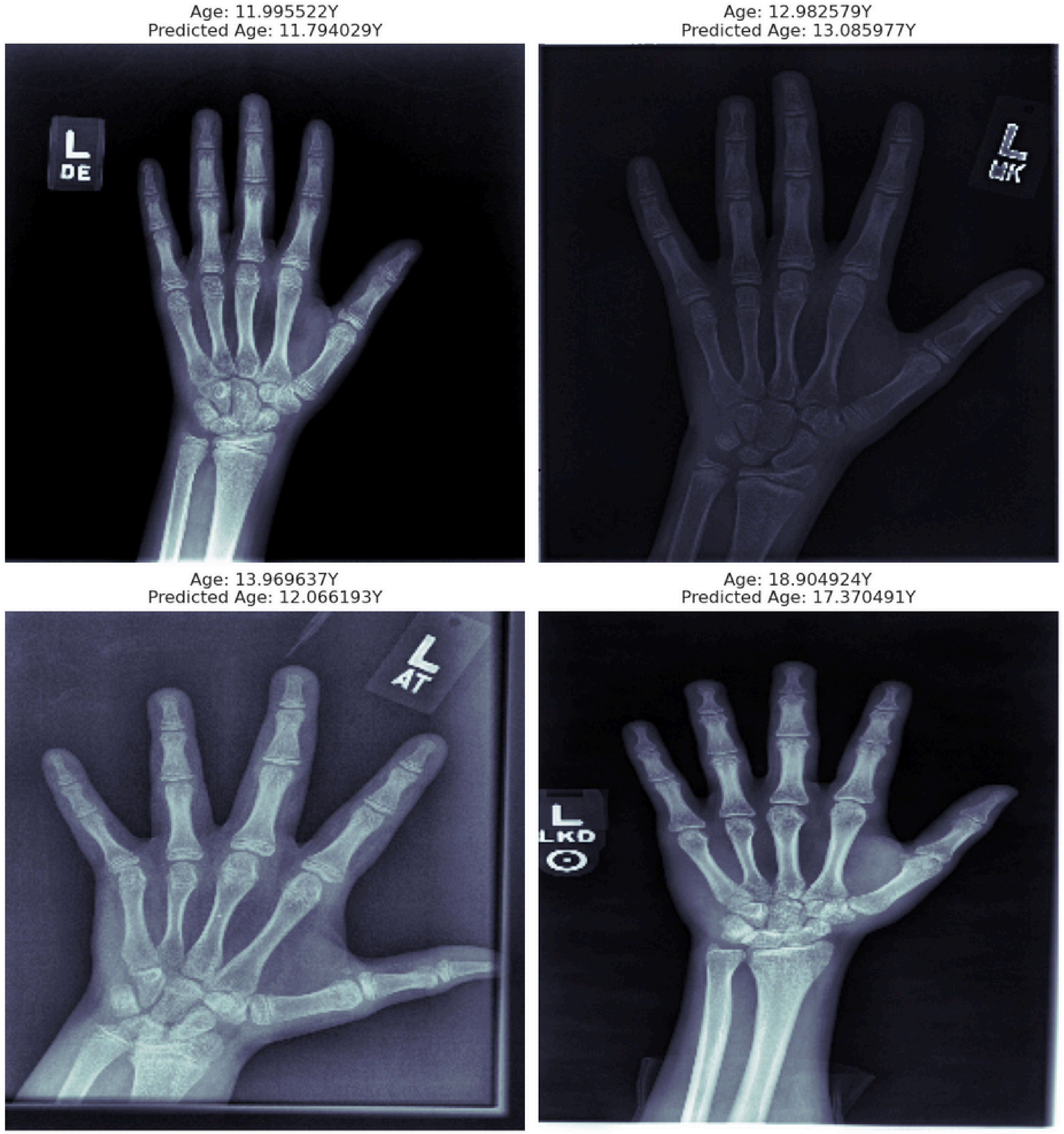
RESULTS

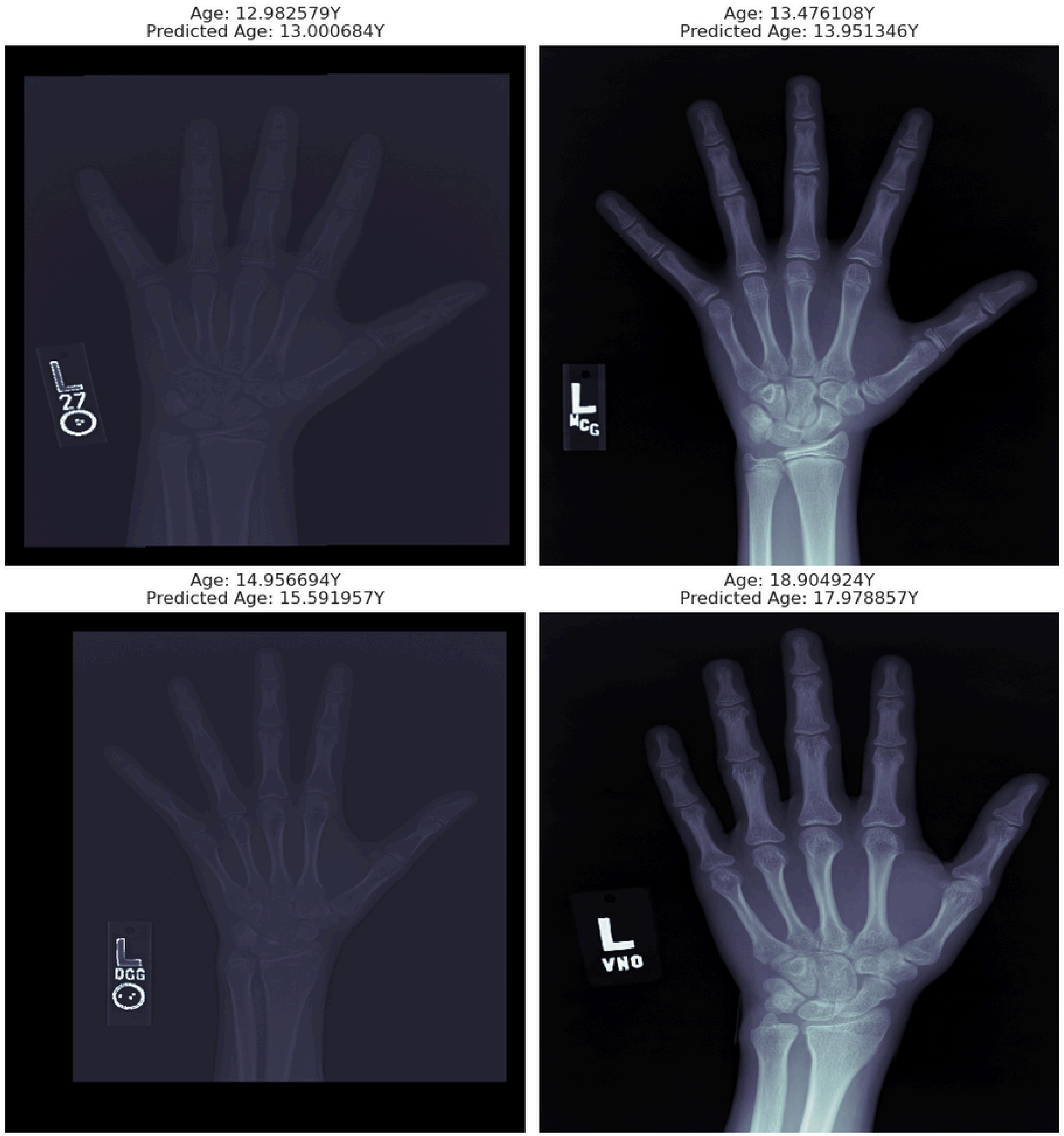
Model	MSE	MAE	R^2	Loss
ConvMixer	142	9.01	0.9216	0.0842
Xception	100	7.63	0.9411	0.0606

Model	Training Time (s)	Inference Time (s)	Memory Usage (MB)
ConvMixer	25724	21.2	5783
Xception	30905	18.6	9762






Age: 11.995522Y
Predicted Age: 11.794029Y
Age: 12.982579Y
Predicted Age: 13.085977Y
Age: 13.969637Y
Predicted Age: 12.066193Y
Age: 18.904924Y
Predicted Age: 17.370491Y


Age: 12.982579Y
Predicted Age: 13.000684Y
Age: 13.476108Y
Predicted Age: 13.951346Y
Age: 14.956694Y
Predicted Age: 15.591957Y
Age: 18.904924Y
Predicted Age: 17.978857Y

CONCLUSION

What we achieved?

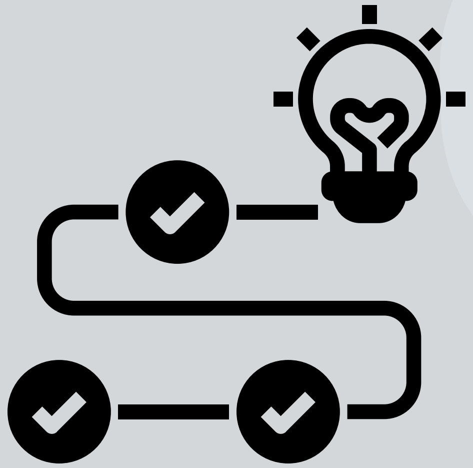
- Developed and evaluated Xception and ConvMixer models for predicting bone age from hand x-rays using the RSNA dataset.
- Incorporated gender input with images through a custom data generator, achieving accurate predictions (MAE < 10 months)

What we learned?

- Handling mixed data types
- Image regression task
- Potential of ML in medical diagnostics

Future Improvements

- ROI-based model
- Newer GPU models (RTX 4090)
- Additional features (ethnicity, chronological age)
- Advanced architectures



Xception → high-performance applications requiring high predictive accuracy and rapid inference

ConvMixer → scenarios where computational efficiency (memory, time) is vital


THANK YOU