FACIAL KEYPOINTS DETECTION

Hannah Gross | Anand Patel

Frances Leung | Rumi Nakagawa

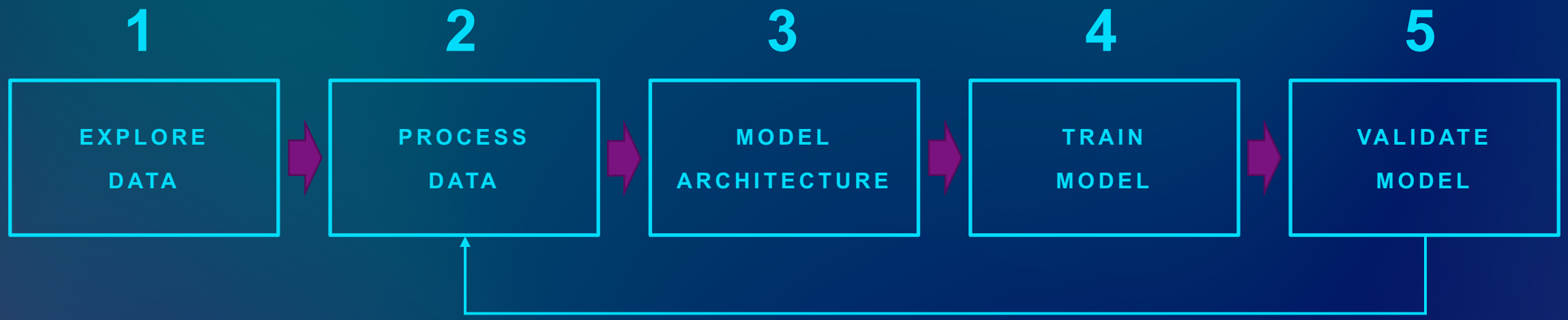
OUR MISSION

Build a machine learning model that predicts

15 keypoints given a facial image

HOW WE ROLL

Our project workflow



Iteratively redefine approach and improve model architecture

THE TRAINING DATASET

left_eye_center_x
left_eye_center_y
right_eye_center_x
right_eye_center_y

left_eye_inner_corner_x
left_eye_inner_corner_y
left_eye_outer_corner_x
left_eye_outer_corner_y
right_eye_inner_corner_x
right_eye_inner_corner_y
right_eye_outer_corner_x
right_eye_outer_corner_y

left_eyebrow_inner_corner_x
left_eyebrow_inner_corner_y
left_eyebrow_outer_corner_x
left_eyebrow_outer_corner_y
right_eyebrow_inner_corner_x
right_eyebrow_inner_corner_y
right_eyebrow_outer_corner_x
right_eyebrow_outer_corner_y

nose_tip_x
nose_tip_y

mouth_left_corner_x
mouth_left_corner_y
mouth_left_corner_x
mouth_left_corner_y
mouth_center_top_lip_x
mouth_center_top_lip_x
mouth_center_top_lip_x
mouth_center_top_lip_y

7,049

Kaggle samples

Split: ~85% train | ~15% validation | 100 samples for evaluation

96X96 | 0-255

Inputs:

Images with 96x96 pixels and 0-255 gradient values

15 (X,Y)

Outputs:

15 keypoints or 30 (x, y) coordinates

EDA (1): IMAGE OVERVIEW

GENDERS

ETHNICITY

SAME PERSON

ACCESSORIES

MUSTACHES

GLASSES

EDA (2): OUTLIERS

PERSPECTIVES

UNIQUE EXPRESSIONS

PAINTINGS/COMICS

TOO BRIGHT/DARK

BABIES/KIDS

PARTIAL FACES

OBSTRUCTIONS

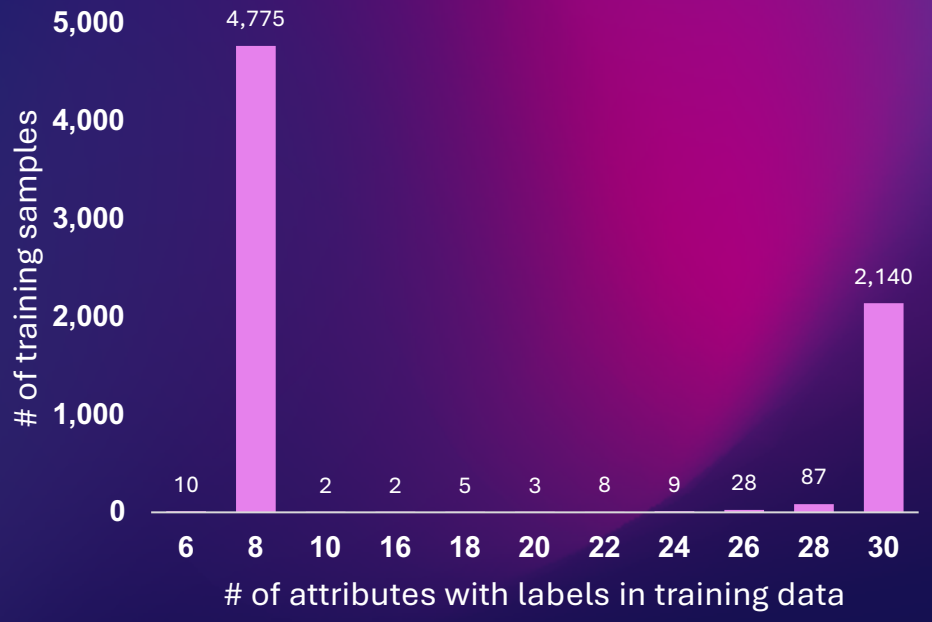
THEATRICAL

EDA (3): OVER HALF OF THE IMAGES HAVE ONLY 8 OUT OF 30 ATTRIBUTES LABELED

COUNTS OF SAMPLES WITH NAN

left_eye_center_x	10
left_eye_center_y	10
right_eye_center_x	13
right_eye_center_y	13
left_eye_inner_corner_x	4778
left_eye_inner_corner_y	4778
left_eye_outer_corner_x	4782
left_eye_outer_corner_y	4782
right_eye_inner_corner_x	4781
right_eye_inner_corner_y	4781
right_eye_outer_corner_x	4781
right_eye_outer_corner_y	4781
left_eyebrow_inner_corner_x	4779
left_eyebrow_inner_corner_y	4779
left_eyebrow_outer_corner_x	4824
left_eyebrow_outer_corner_y	4824
right_eyebrow_inner_corner_x	4779
right_eyebrow_inner_corner_y	4779
right_eyebrow_outer_corner_x	4813
right_eyebrow_outer_corner_y	4813
nose_tip_x	0
nose_tip_y	0
mouth_left_corner_x	4780
mouth_left_corner_y	4780
mouth_left_corner_x	4779
mouth_left_corner_y	4779
mouth_center_top_lip_x	4774
mouth_center_top_lip_x	4774
mouth_center_top_lip_x	33
mouth_center_top_lip_y	33

NUMBER OF ATTRIBUTES LABELED



MODELLING OVERVIEW

TYPE

- Supervised learning
- Output: Regression

ALGORITHM

- Convolutional neural network (CNN)

METRICS

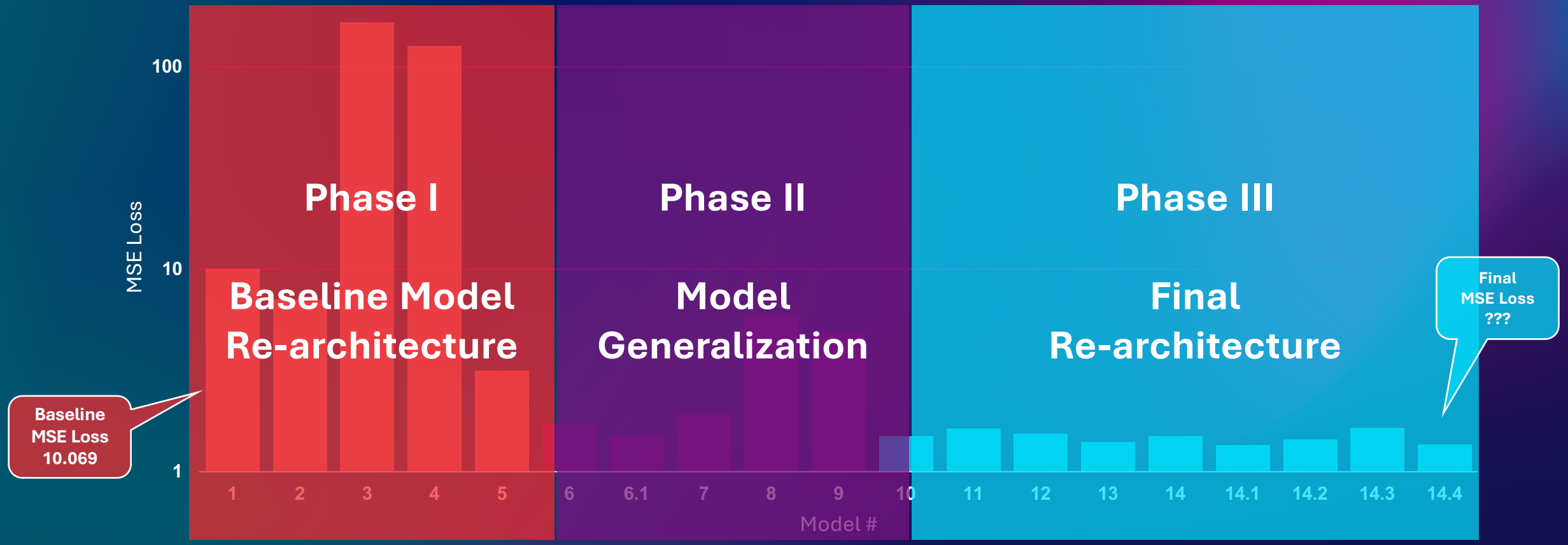
- MSE Loss
- RMSE

CHALLENGE

- Handle missing target values
- Design architecture and strategy to minimize prediction error

OUR MODELLING JOURNEY

Minimizing Mean Squared Error Loss



BASELINE MODEL

BASELINE

- Used textbook CNN architecture (Fashion dataset)
- Filled in NaN values with pandas.DataFrame.ffill to replace NULL values with values from a previous sample in the training dataset

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 94, 94, 32)	320
max_pooling2d (MaxPooling2D)	(None, 47, 47, 32)	0
conv2d_1 (Conv2D)	(None, 45, 45, 64)	18496
max_pooling2d_1 (MaxPooling2	(None, 22, 22, 64)	0
conv2d_2 (Conv2D)	(None, 20, 20, 128)	73856
flatten (Flatten)	(None, 51200)	0
dense (Dense)	(None, 64)	3276864
dropout (Dropout)	(None, 64)	0
dense_1 (Dense)	(None, 30)	1950

Total params: 3,371,486
Trainable params: 3,371,486
Non-trainable params: 0

PHASE I

Baseline Model Re-architecture

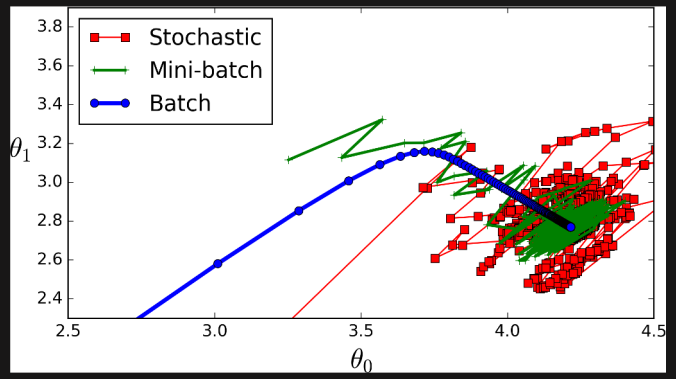
PHASE I – BASELINE MODEL RE-ARCHITECTURE

PIVOTAL CHANGES

- Data shuffling
- LeakyRelu, more CNN layers, batch normalization
- Changing number of epochs and mini-batch sizes
- Running a 2nd training session with a larger mini-batch
- Saving out best models via callouts

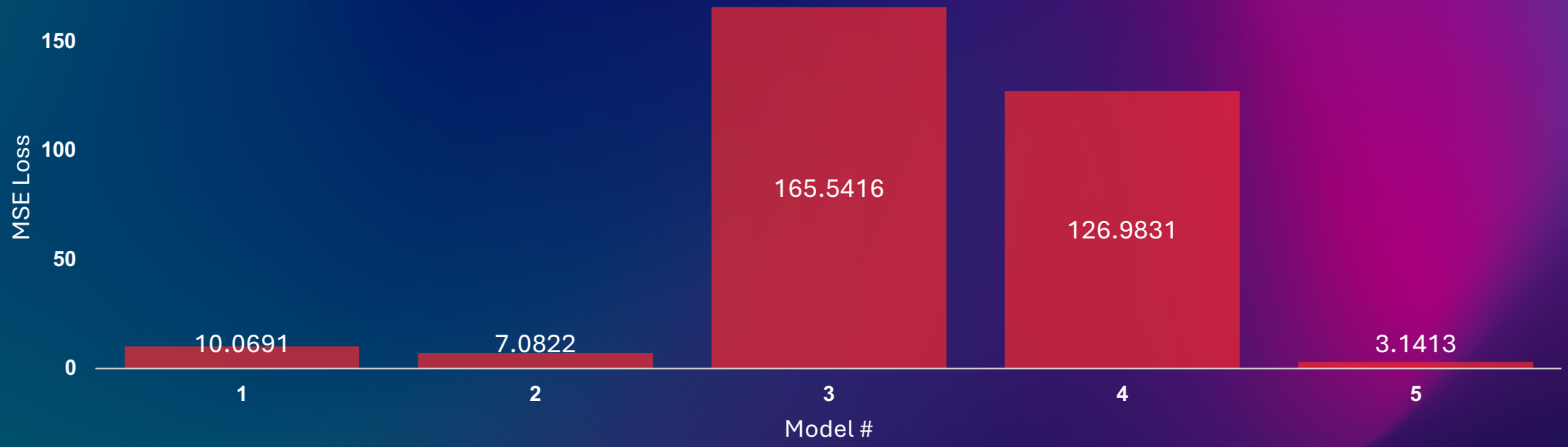
LESSONS LEARNED

- Mini-batch > SGD
- Architecture improvements helped
- 2nd training session helped
- Saving best model by MAE, RMSE



CNN MODEL ITERATIONS

Phase I - Baseline Model Re-architecture



1	2	3	4	5
Fashion dataset (textbook) CNN architecture, Relu	Kaggle CNN Architecture, LeakyRelu			Data shuffling, 2nd training session, other changes
50 Epochs				80, 10 Epochs
128 Batch size	256 Batch size	1 Batch size	1 Batch size, shuffle	64, 512 Batch sizes

PHASE II

Model Generalization

PHASE II – SEMI-SUPERVISED LEARNING (1)

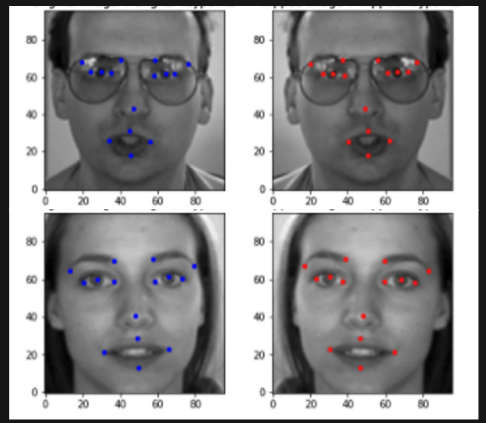
SEMI-SUPERVISED LEARNING

- 14 models ran, each for a set of (x, y) facial keypoint (Note: Nose tip has complete labeled data and thus does not need filling)
- Applied our best model to each keypoint
- 20-hour total runtime to generate filled in training data and generate a new .csv

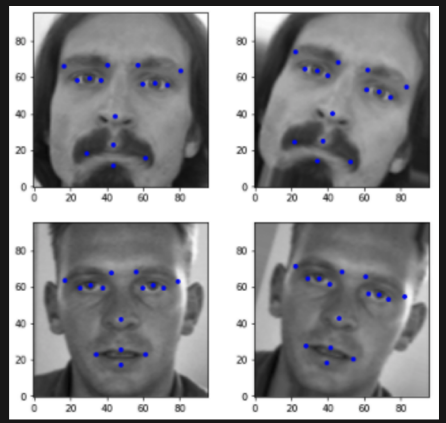
LEFT EYE OUTER CORNER PREDICTION

PHASE II – DATA AUGMENTATION

- Increase training sample size by flipping and rotating training data
- This is done based on the assumption that increasing sample size would generalize the model



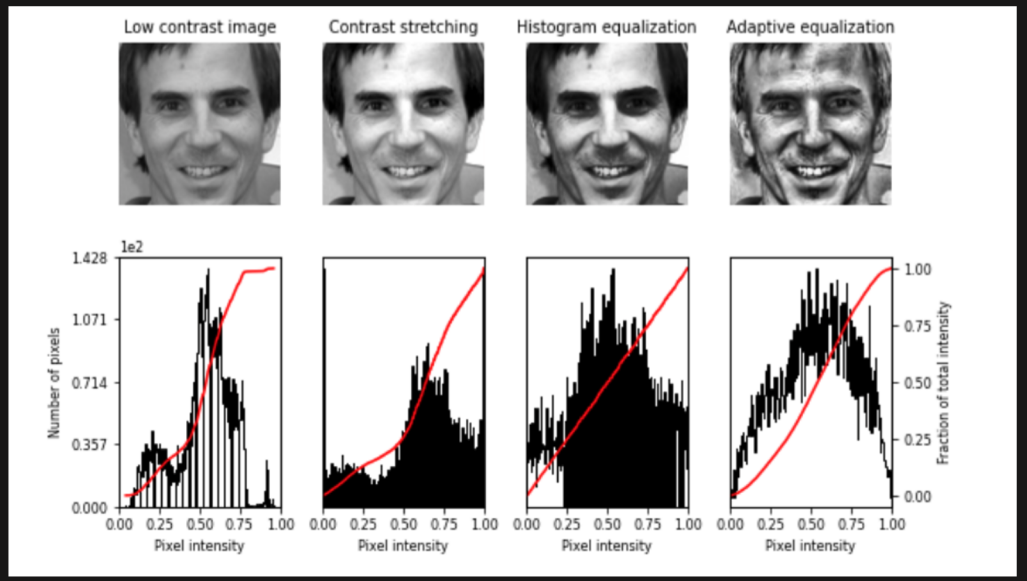
Horizontal Flip



Rotation (20 degrees)

PHASE II – ADAPTIVE EQUALIZATION (1)

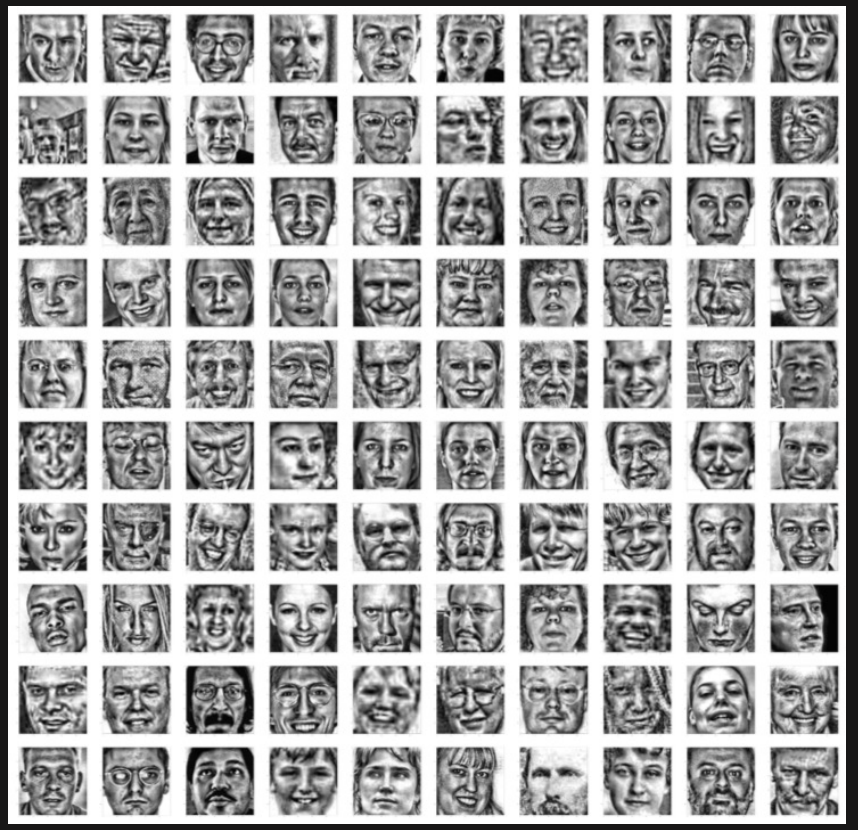
- Problem: Images having varying brightness and contrast
- Attempts made: We experimented with 3 different types of image adjustments using skimage.exposure:
  - Contrast stretching
  - Histogram equalization
  - Adaptive equalization
- Adaptive equalization appeared to be able to best normalize brightness/contrast and accentuate the facial features. It was applied to all raw training images in some of our models



PHASE II – ADAPTIVE EQUALIZATION (2)

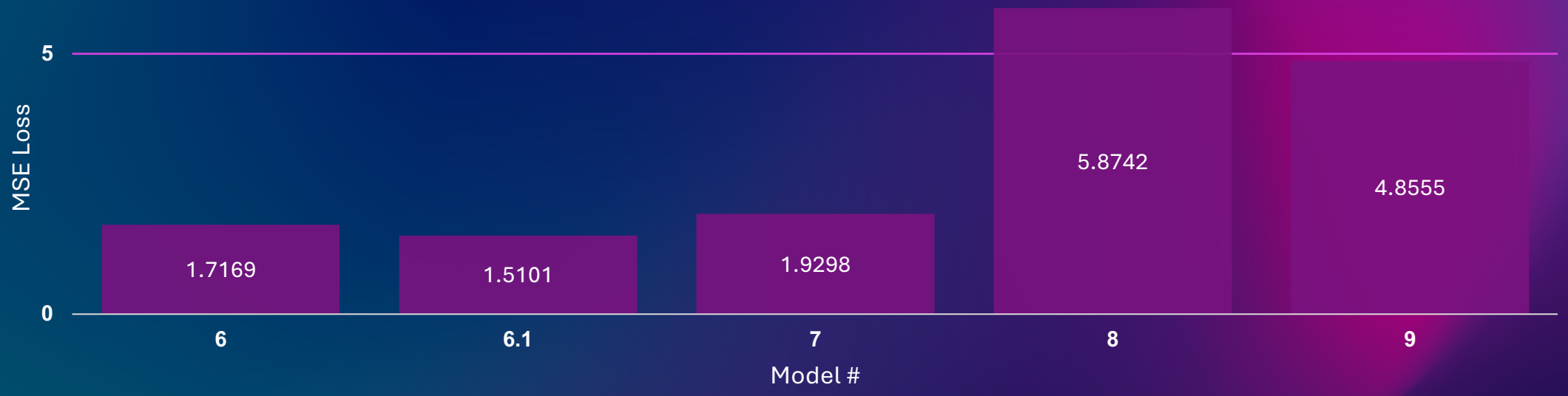
BEFORE

AFTER



CNN MODEL ITERATIONS

Phase II - Baseline Generalization

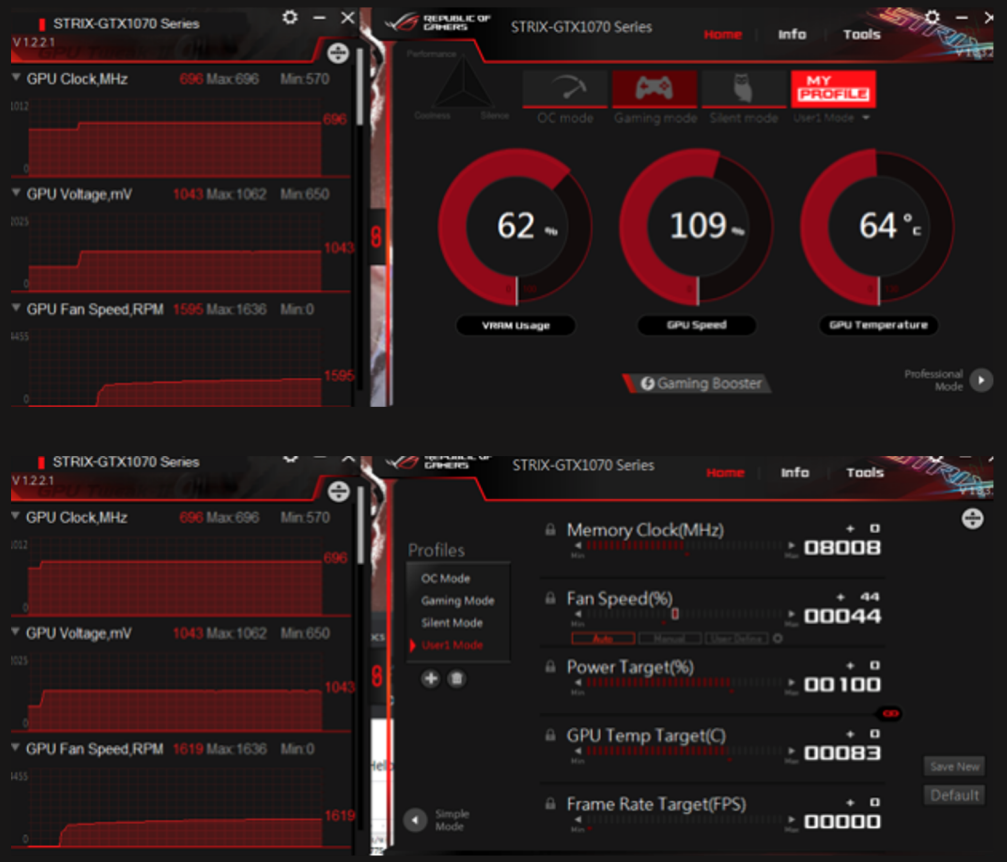


PHASE III

Final Re-architecture

PHASE III – IMPLEMENTATION OF GPU

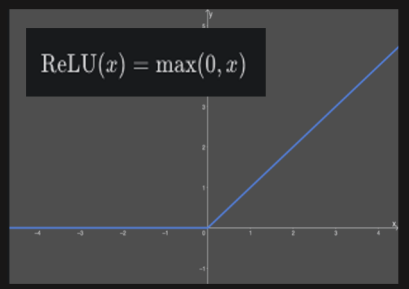
- This project gave some of us a reason to work our gaming GPUs hard
- **NVIDIA GeForce GTX 1070** graphics card on a local desktop: E.g. Runtime for a typical epoch reduced from 33 sec to 9 sec or less
- Alternatively, using GPU offered by **Google Colab ($10/month)** also provided decent performance
- Overall, when generating semi-supervised training models/data, we reduce runtime from 20+ hours to 2+ hours



PHASE III – ACTIVATION FUNCTION

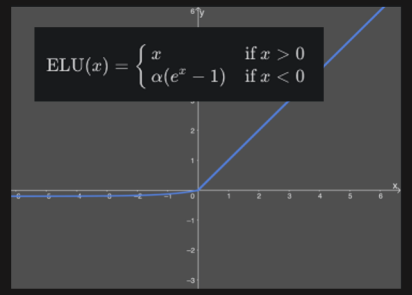
- Performance of models were improved by updating activation function by step
- Also experimented different hyperparameters:
  - Batch normalization contributed
  - We could not see clear contribution of regularization (L1, L2) in our architecture
  - Updating epoch size/batch size contributed

ReLU

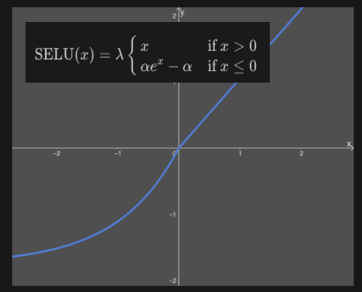


Leaky ReLU

ELU



SELU

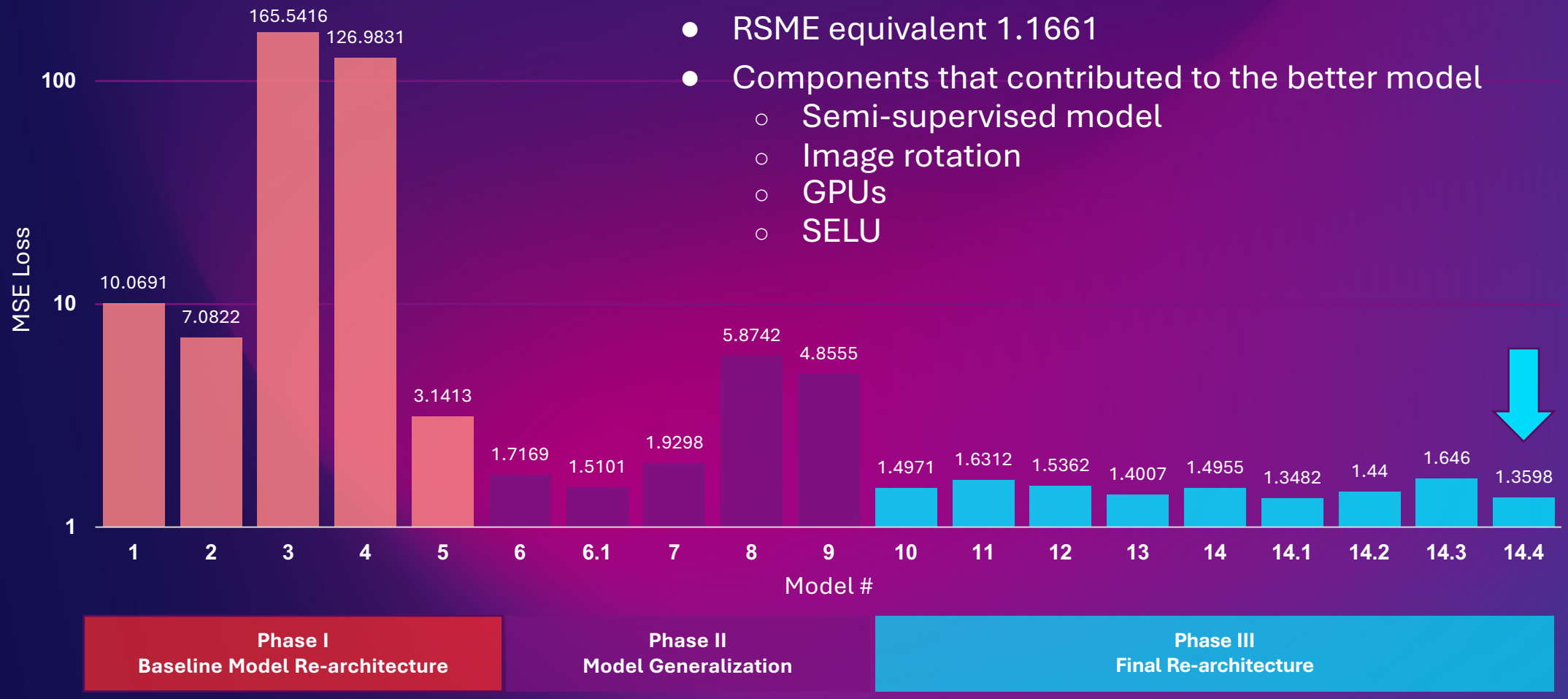


CNN MODEL ITERATIONS

Phase III – Baseline Re-Architecture

FINAL RESULTS

- MSE loss improved from 10.069 to 1.3598
- RSME equivalent 1.1661
- Components that contributed to the better model
  - Semi-supervised model
  - Image rotation
  - GPUs
  - SELU

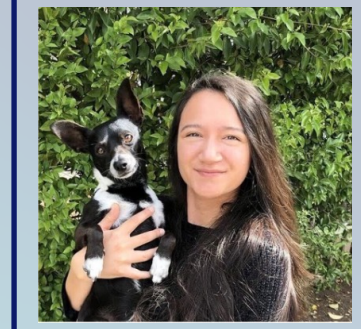


FUTURE EXPERIMENTATION

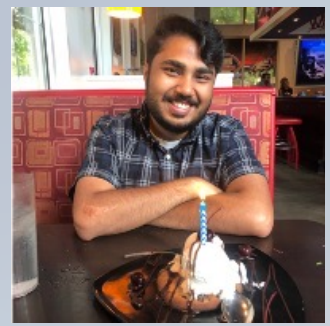
- Explore other CNN architecture : Sequential vs. functional models (E.g. Train models with 8-attribute vs. 30-attribute labeled data separately or in parallel)

- Explore further on feature engineering techniques

THANK YOU



HANNAH



ANAND

FRANCES

RUMI

APPENDIX

SEMI-SUPERVISED LEARNING ARCHITECTURE