FACIAL LANDMARK PREDICTION OF MASKED FACES

ECE-6397 PROJECT

TEAM

SHEETAL KUMAR - 2103084

SRIKANTH REDDY NIMMALA-2100527

SAIVARDHAN REDDY NOMULA-1799472

SAI SRI HARSHA AMBATI -2090997

SUDHARSAN RAGOTHAMAN-2097435

INTRODUCTION

- Face recognition (FR) systems are conventionally presented with primary facial features such as eyes, nose, and mouth, i.e., non-occluded faces.
- However, a wide range of situations and circumstances impose that people wear masks in which faces are partially hidden or occluded.
- Such common situations include pandemics, laboratories, medical operations, or immoderate pollution.
- Accordingly, all countries in the world require that people wear a protective face mask in public places, which has driven a need to investigate and understand how such face recognition systems perform with masked faces.

OBJECTIVE

- The goal of this research work is to use machine learning algorithms to predict the facial landmarks of a person wearing a mask.

- Our work proposes a novel method of predicting facial landmarks, which are hidden under the mask.

- Using these landmarks a realistic face mesh can be developed allowing for facial reconstruction

PREVIOUS WORKS

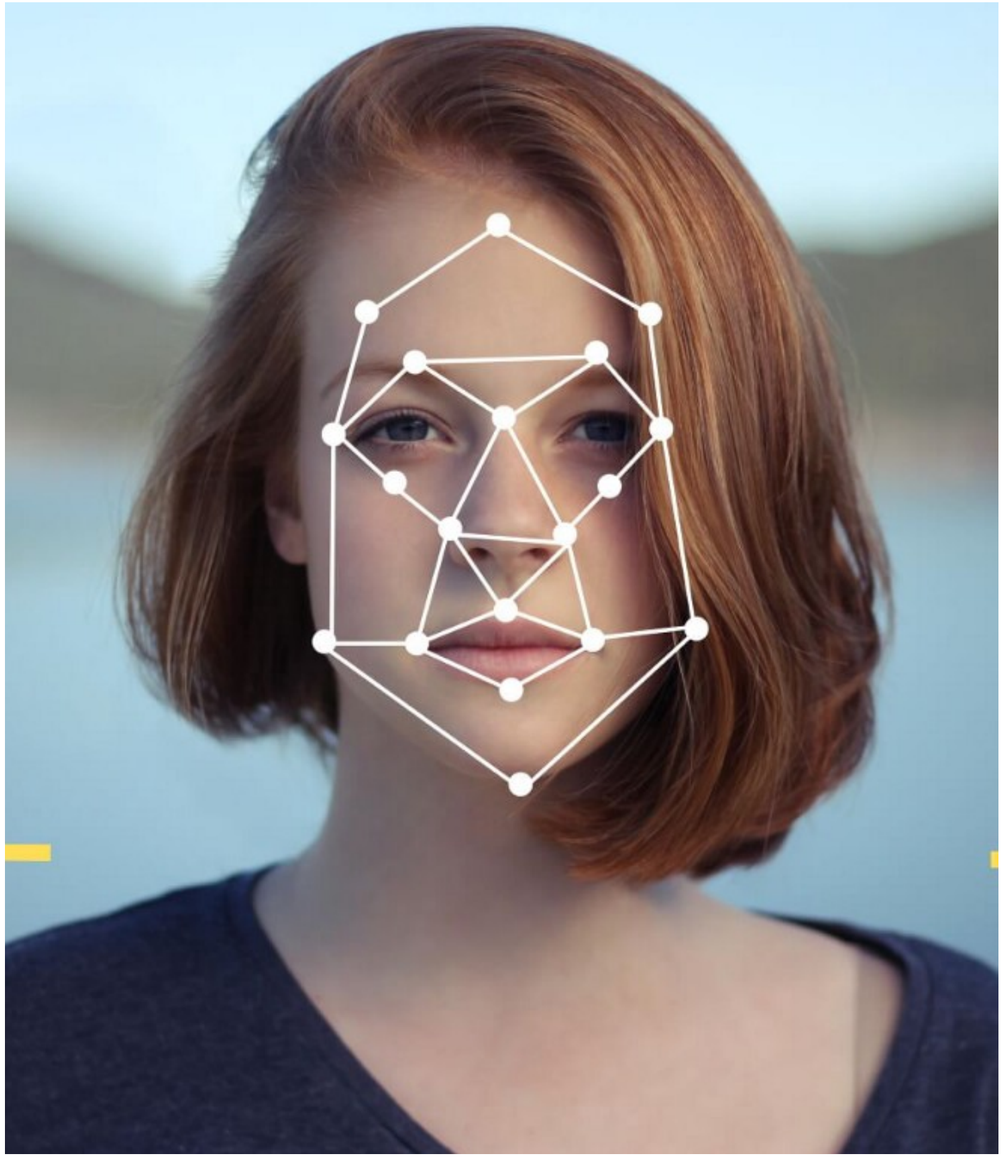
In the past, researchers have shown great interest in facial recognition as it has multiple advantages in the security and medical fields. It took multiple folds in the past 20 years to reach an accuracy of 99%+. Major issues in facial recognition were changing in face shape and structure both inter and intra due to the person's face may look way different in multiple acts, illuminations, expressions, and occlusions. Another major drawback of the model is the obstruction in the face like wearing masks. These are the two problem statements we have addressed in our project.

PREVIOUS WORKS

RESEARCH QUESTION

- Can we predict the hidden facial landmark points as a function of the visible landmark points?

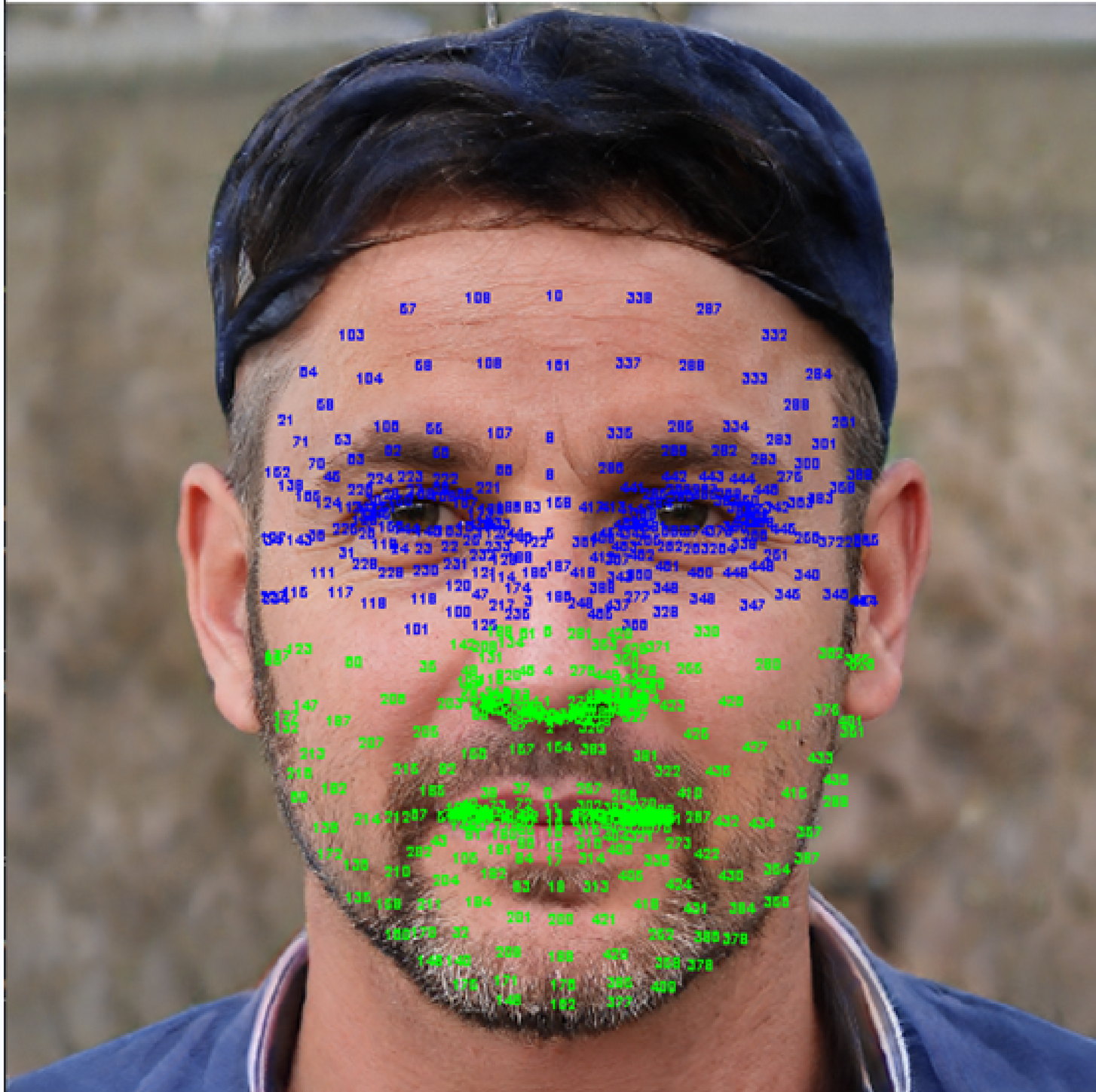
Our work proposes a novel method of predicting facial landmarks, using which a realistic face mesh can be developed allowing for facial reconstruction. The scope of this project is currently limited to detecting only the facial landmarks of a person wearing a mask.



DATA COLLECTION

METHODLOGY

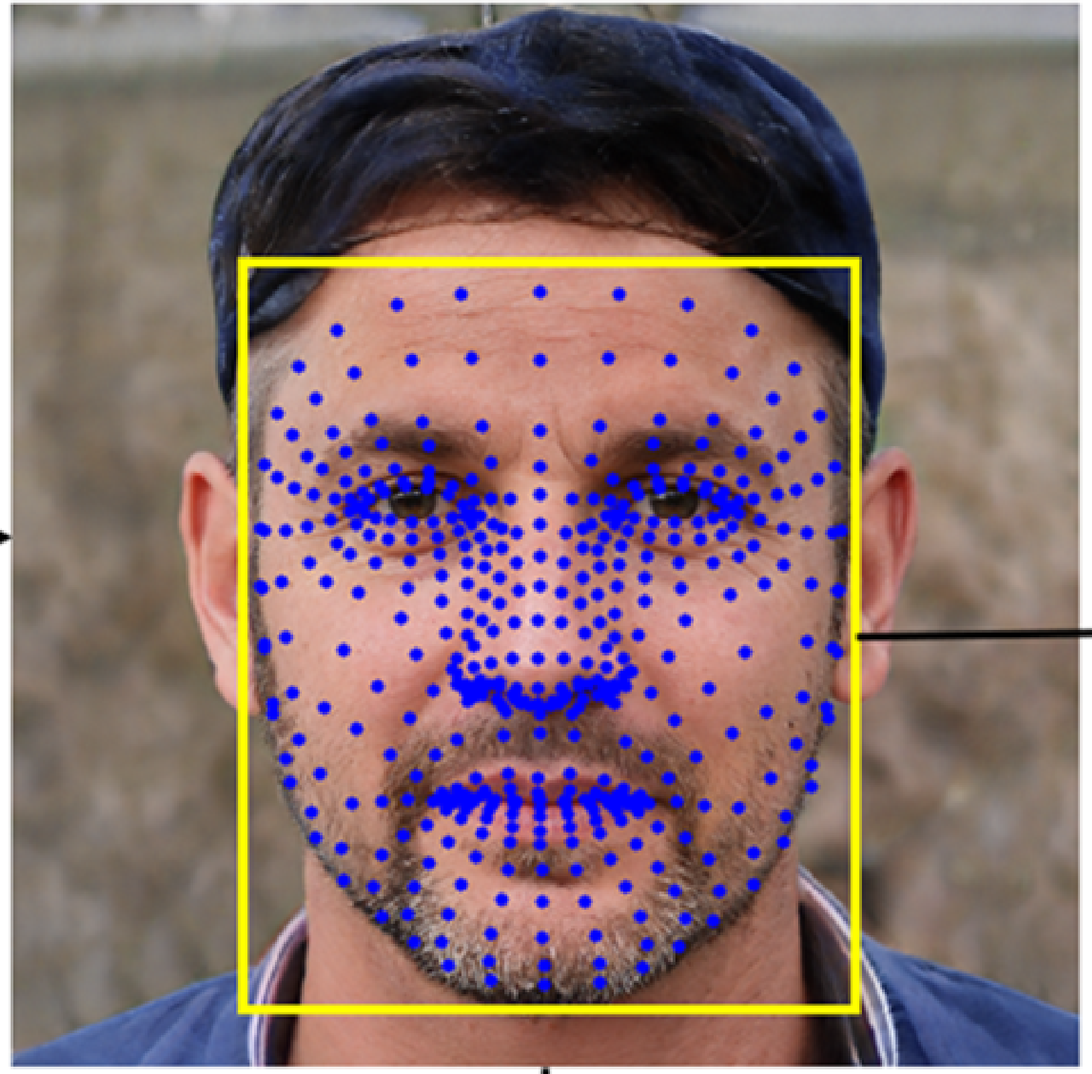
First, we have developed a dataset containing 468 facial landmarks per image, using the MediaPipe framework. In order to account for the different sizes of the faces on an image, we have scaled the datapoints w.r.t a certain scaling factor like the face width and distance of a data point from the central location of the face.



METHODLOGY

Original Image

MediaPipe

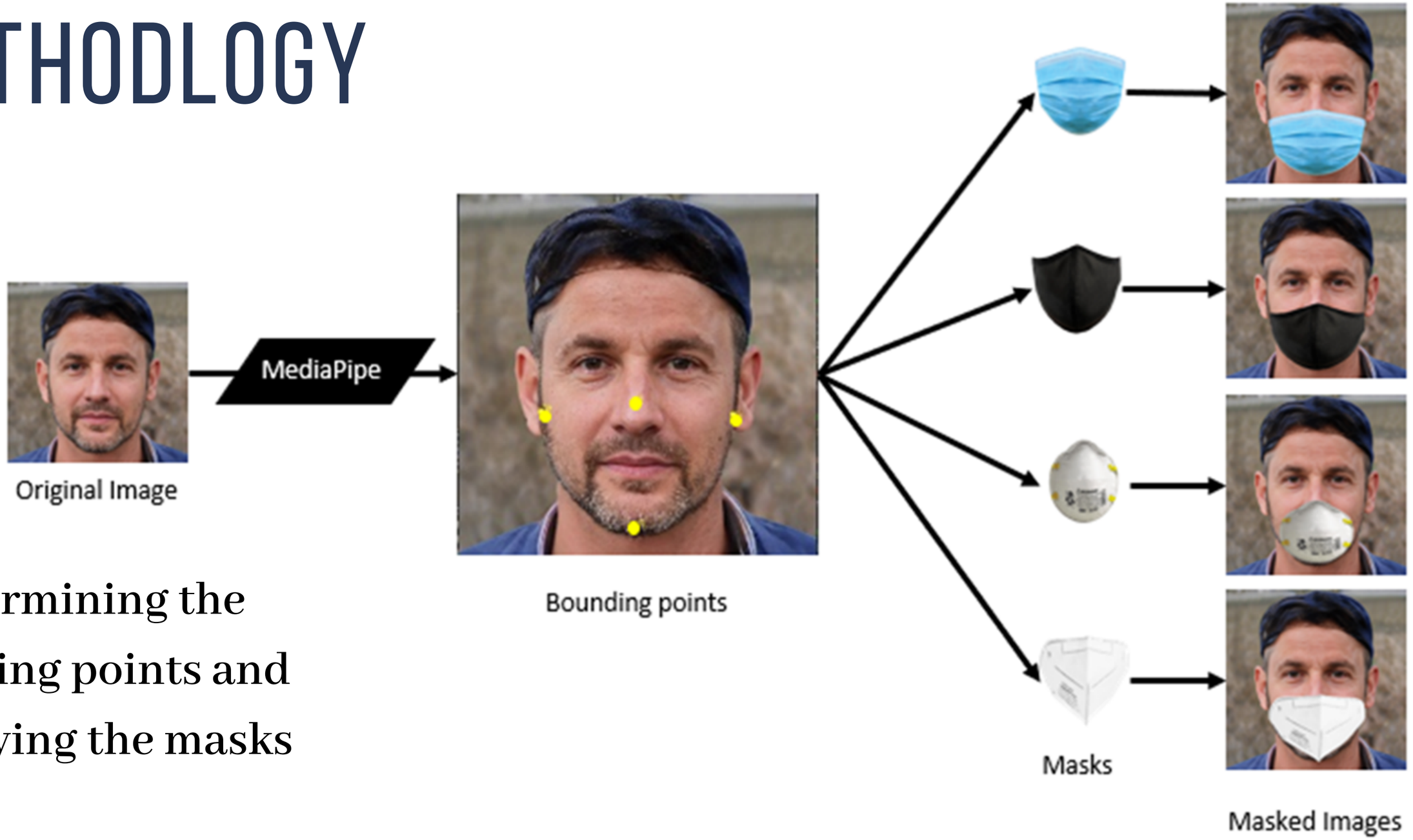


Facial Landmark points

Processing the image to extract all the landmarks

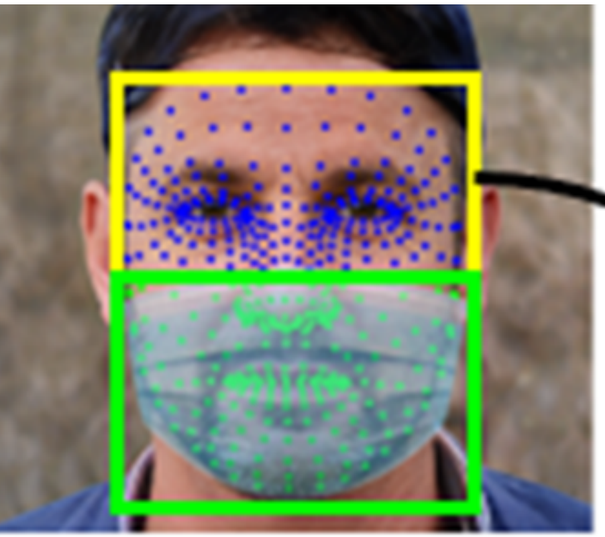
```
original_dataset =
      {'Image 1':
            {'x-points': [x1,x2,...xn]},
            {'y-points': [x1,x2,...xn]},
```

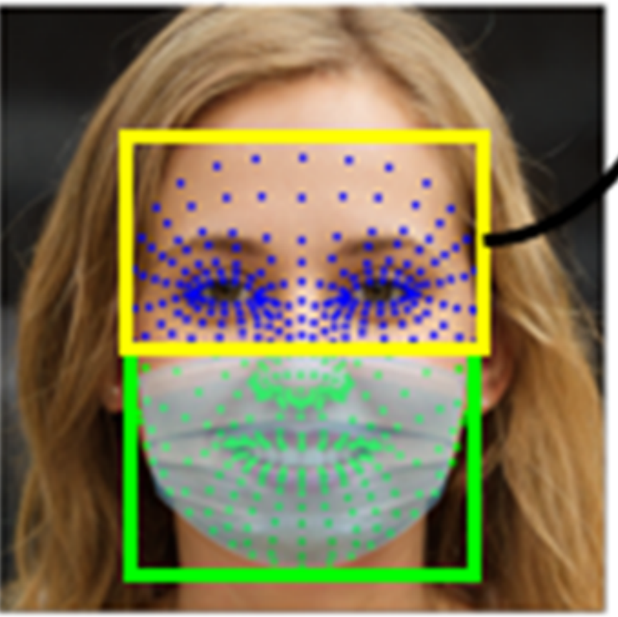
METHODLOGY



Determining the bounding points and overlaying the masks

METHODLOGY





Common Visible Points

common_vis_points 206

Inputs- Visible points

Neural Network

common_hid_points 261

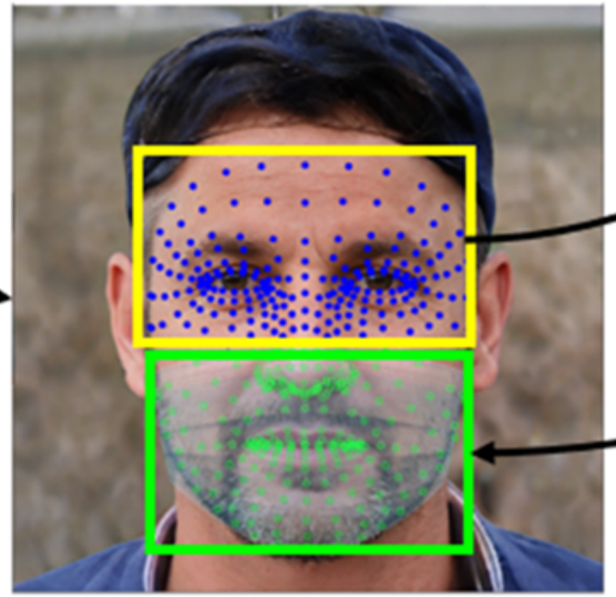
Identifying common visible points across all the images

METHODLOGY

Visible landmark points as input to the NN

Masked Image

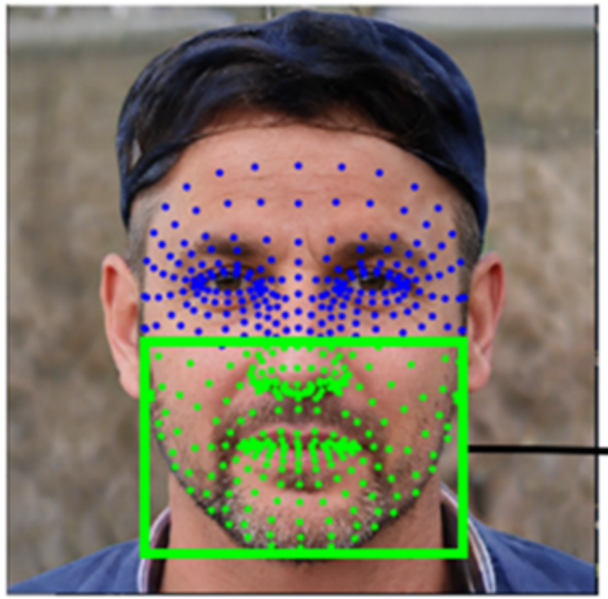
Face Mask Estimation - MediaPipe



Inputs- Visible points

Neural Network

Prediction- Hidden points



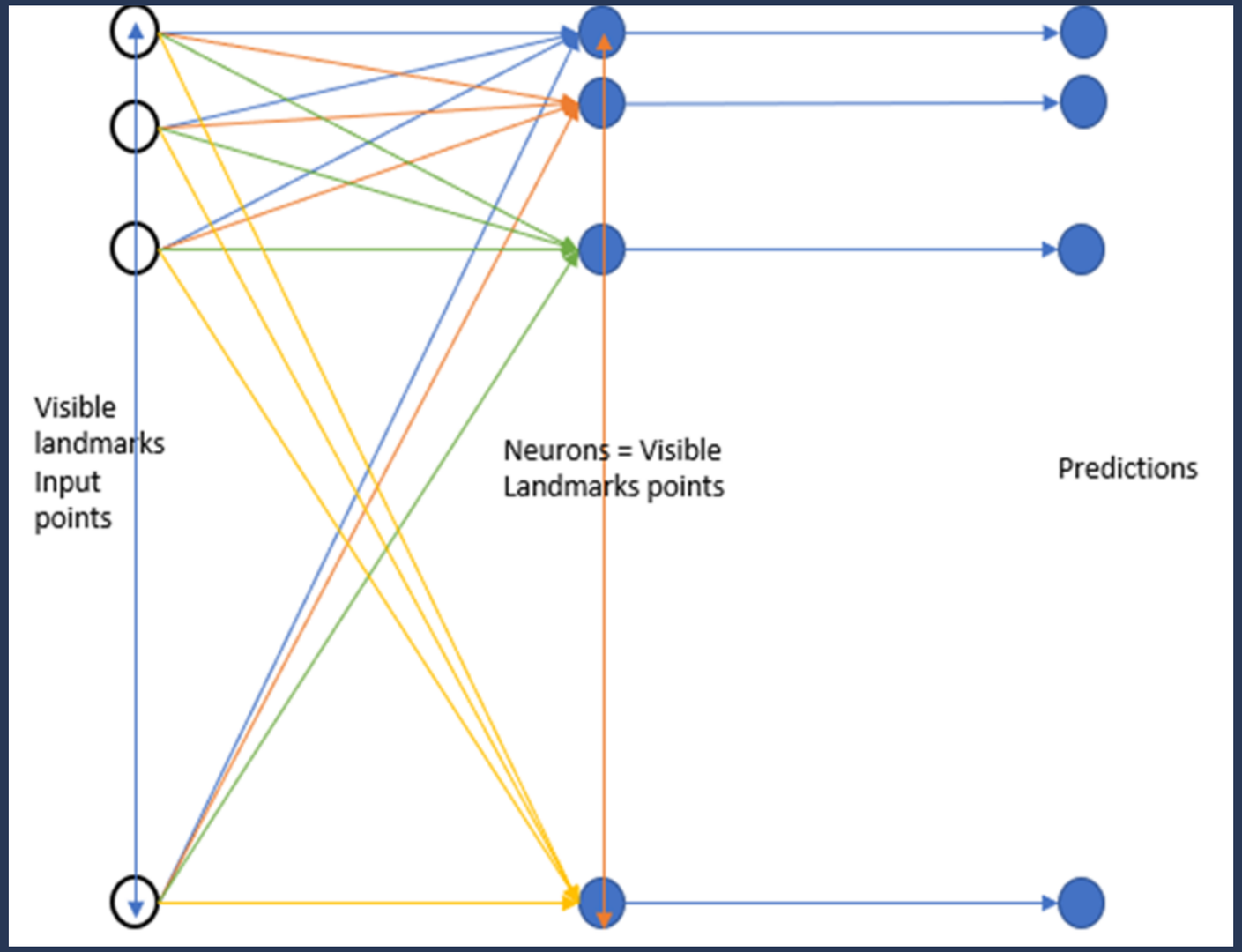
Original Hidden points

Error

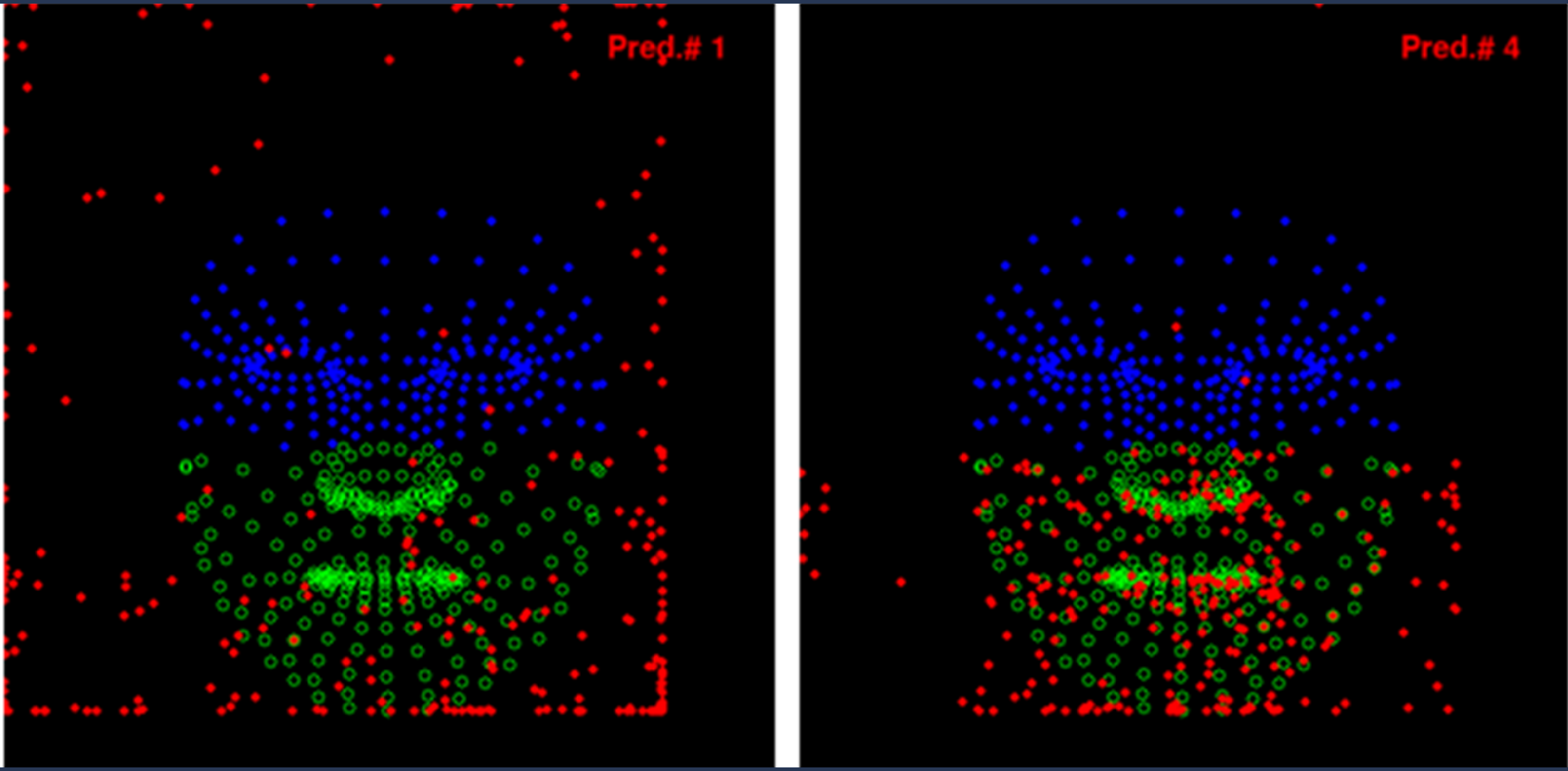
SIMULATIONS & RESULTS

COMPARISON AND INFERENCE

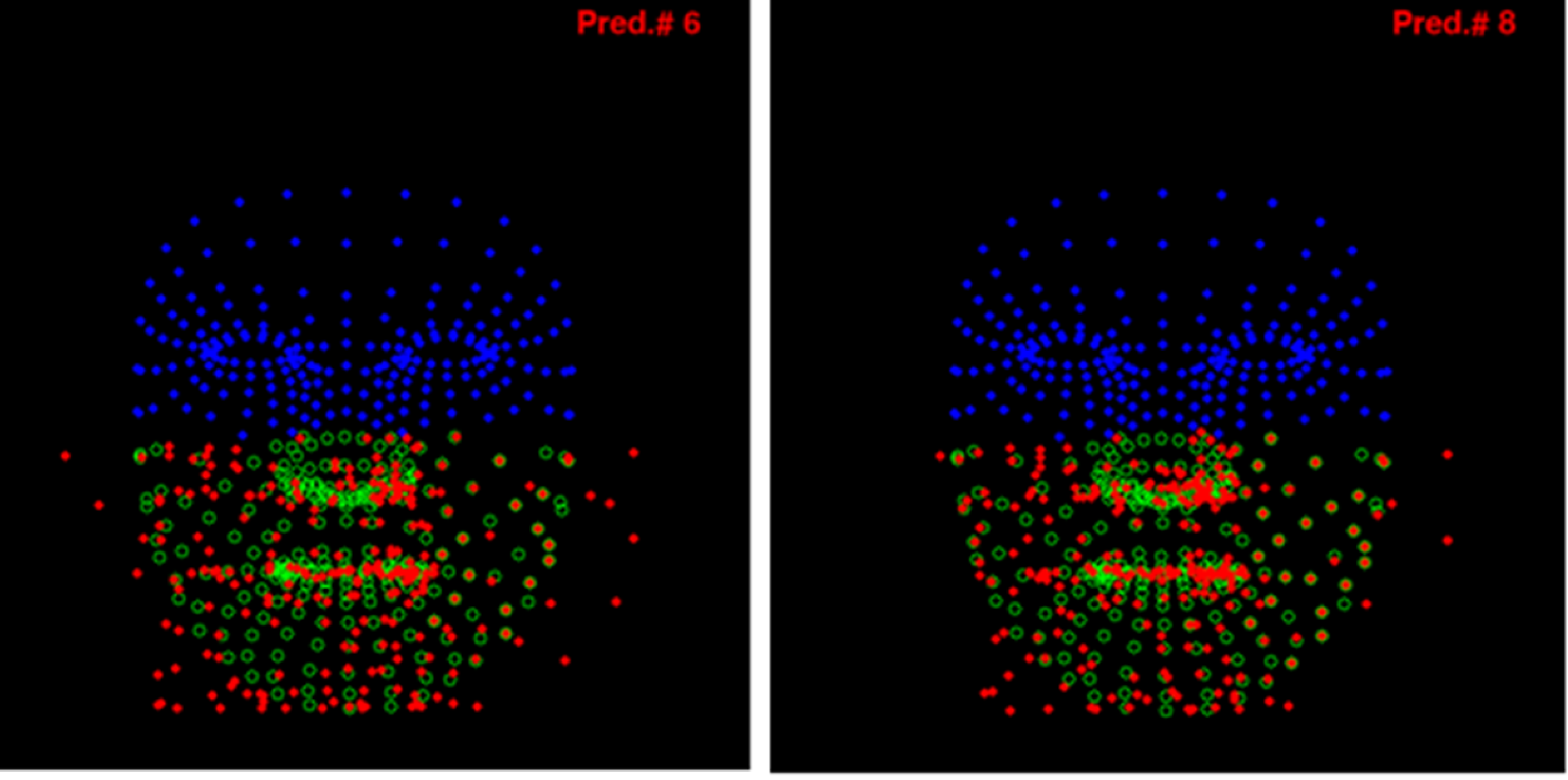
SIMLUATION & RESULTS-1

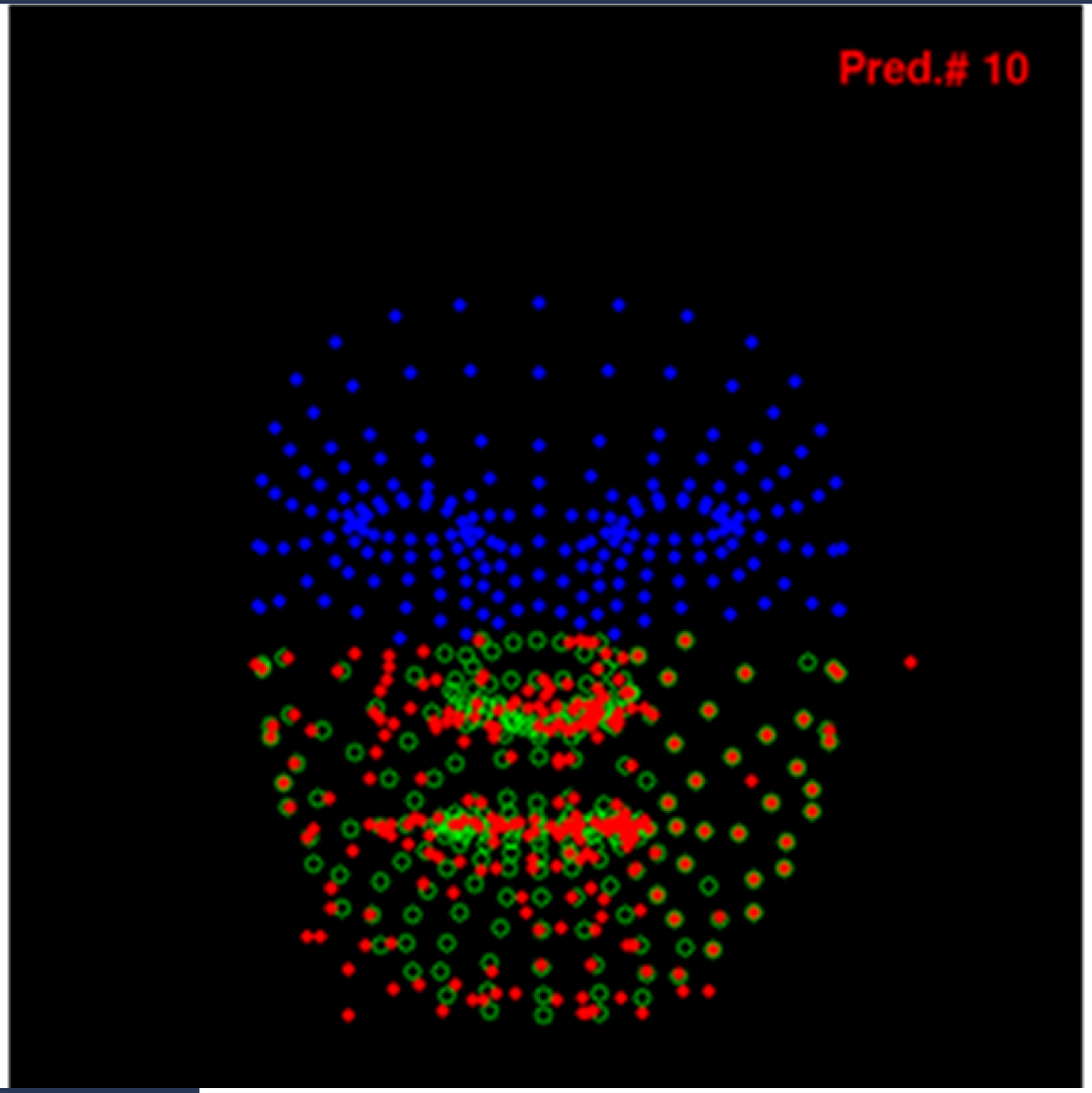


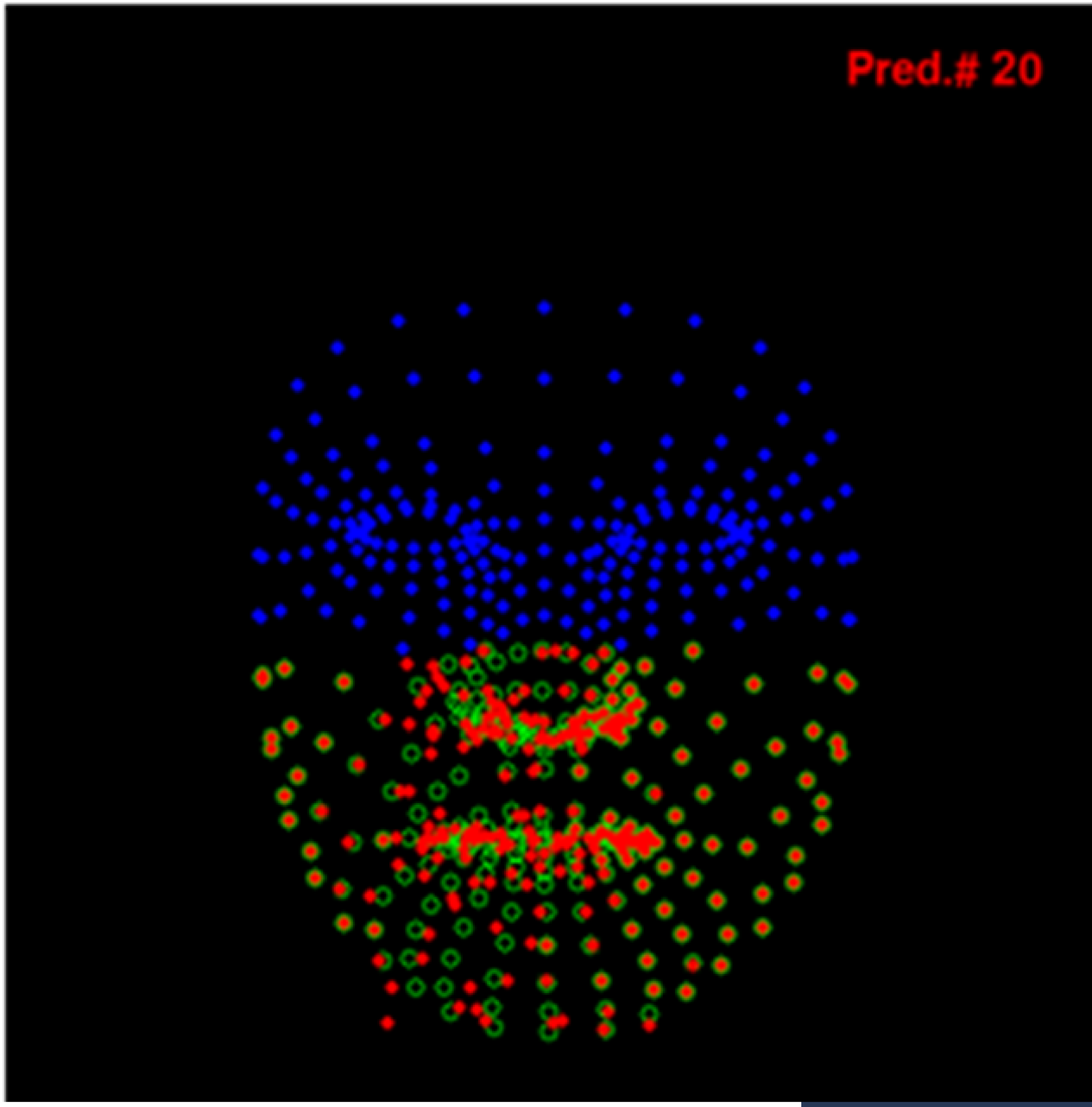
Architecture of 1 Layer FCNN used for initial simulation



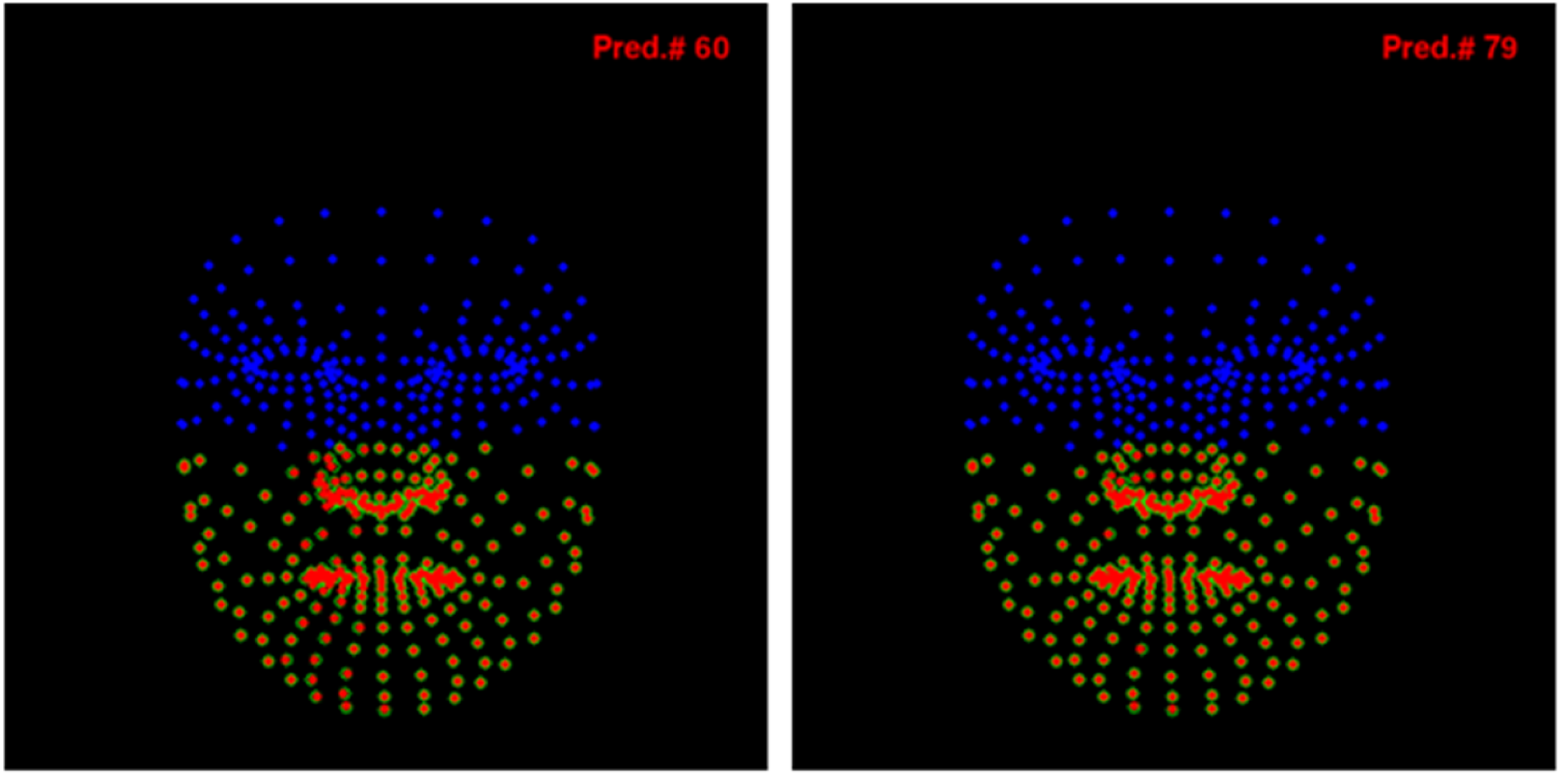
: Inputs (Visible points) : Actual hidden points : Predicted hidden points


Pred.# 6
Pred.# 8
: Inputs (Visible points)
: Actual hidden points
: Predicted hidden points





● : Inputs (Visible points) ● : Actual hidden points ● : Predicted hidden points


Pred.# 60
Pred.# 79
: Inputs (Visible points)
: Actual hidden points
: Predicted hidden points

SIMLUATION & RESULTS-1

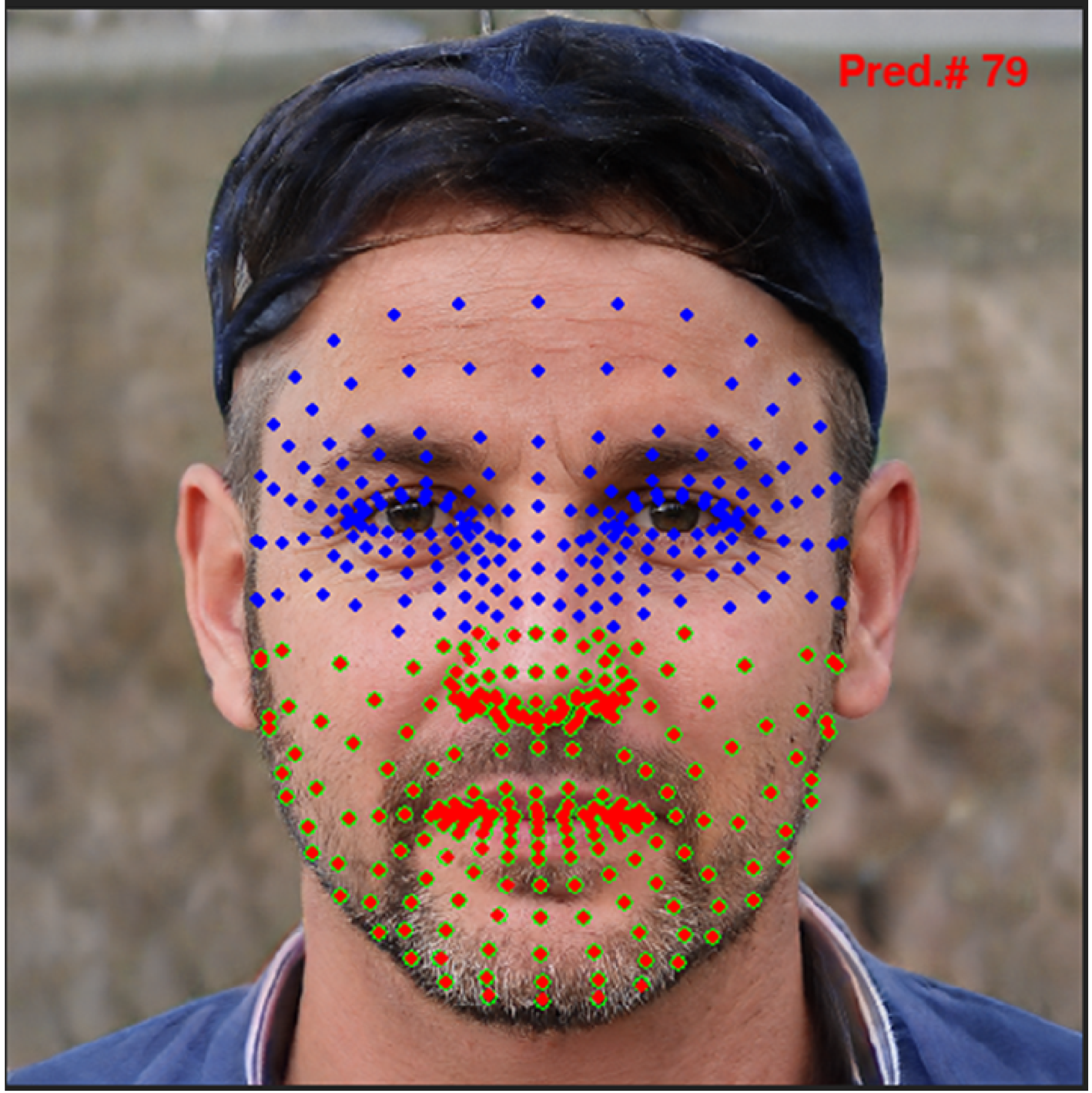
Image Sample : 1.

Epochs : 80

Model : 1 Layer FCNN

Batch size : N

Optimizer: Gradient Descent



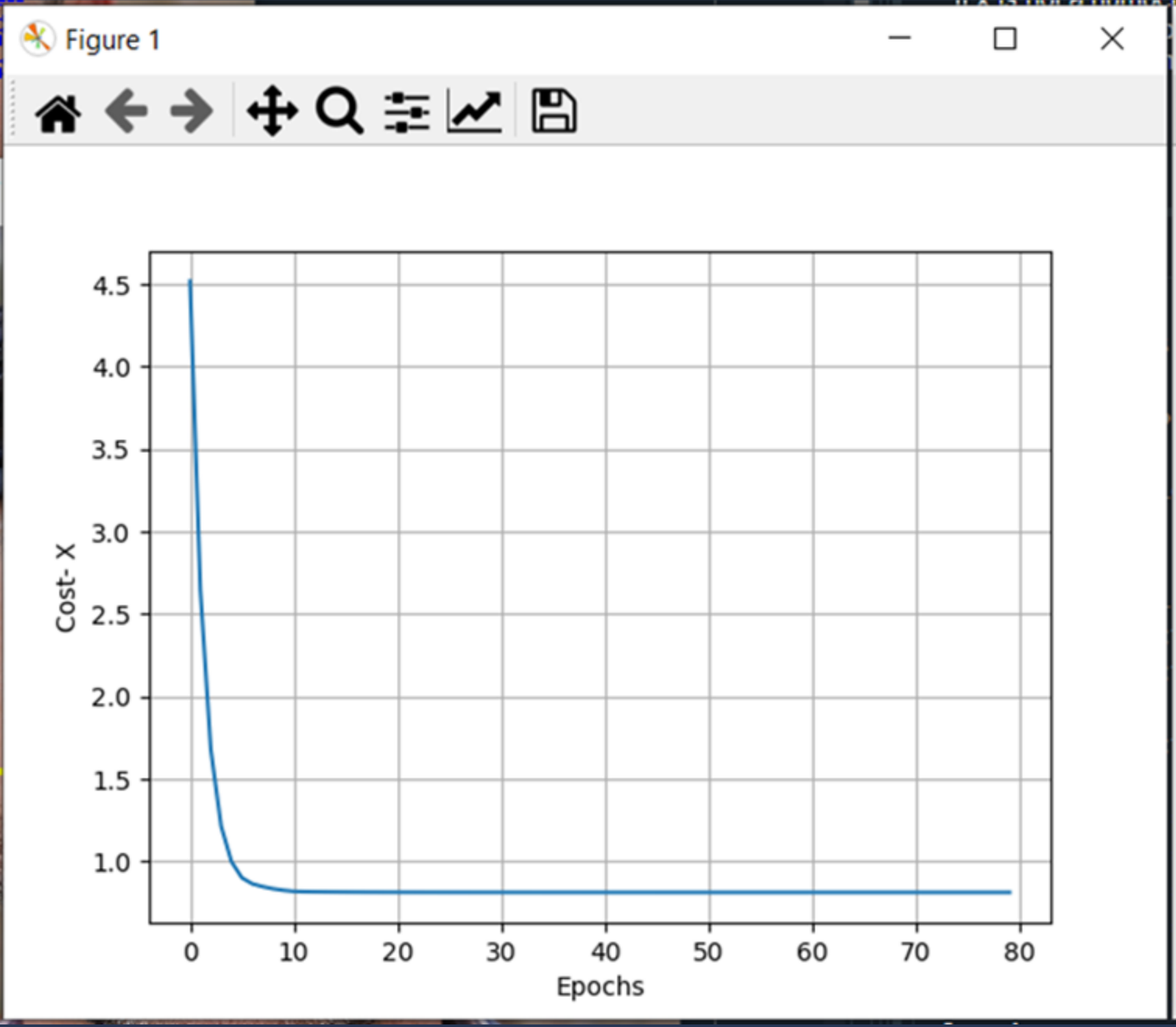


Figure 1
Cost- X
Epochs
4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0
10
20
30
40
50
60
70
80

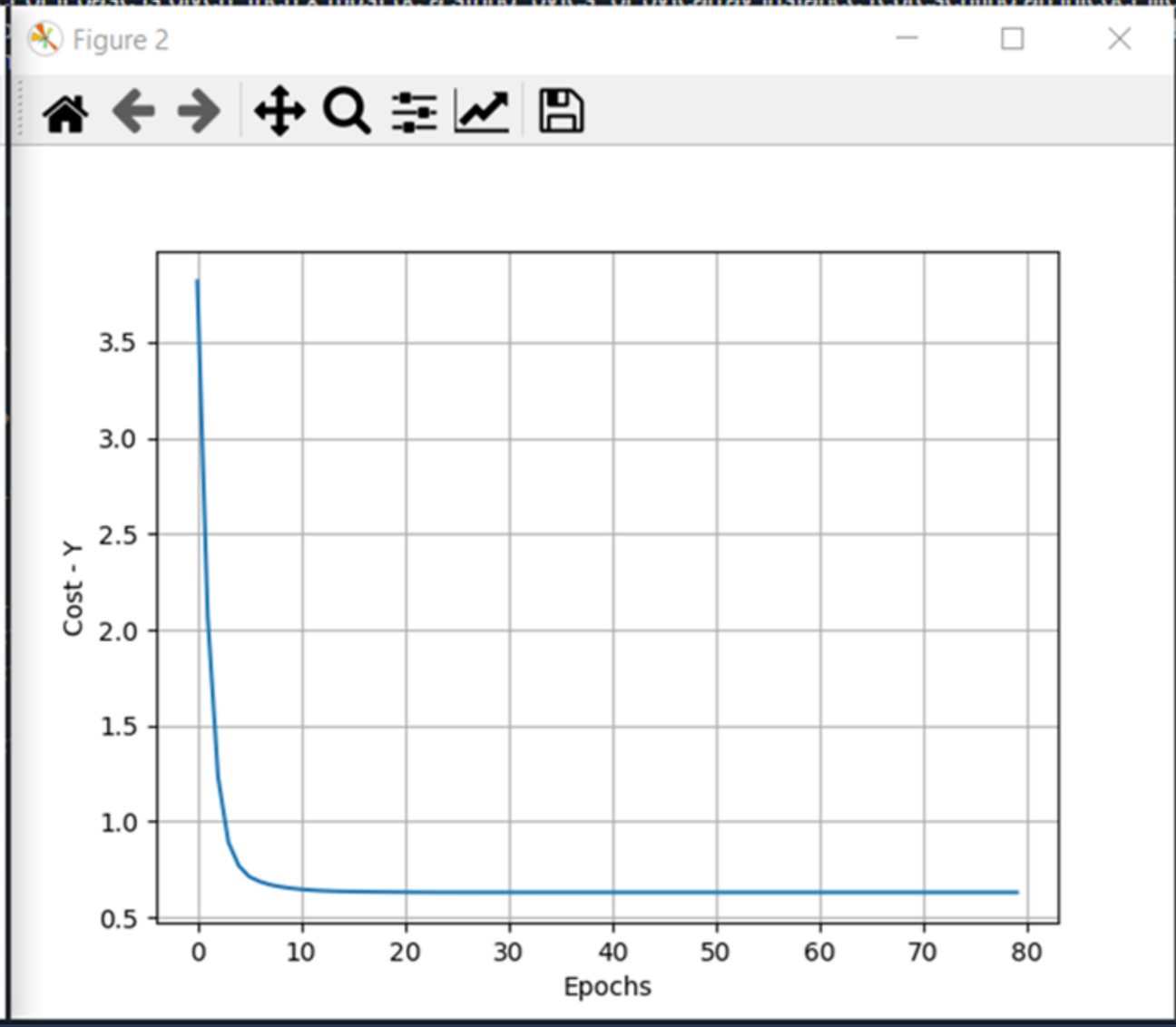


Figure 2
Cost - Y
Epochs
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0
10
20
30
40
50
60
70
80

SIMLUATION & RESULTS-2

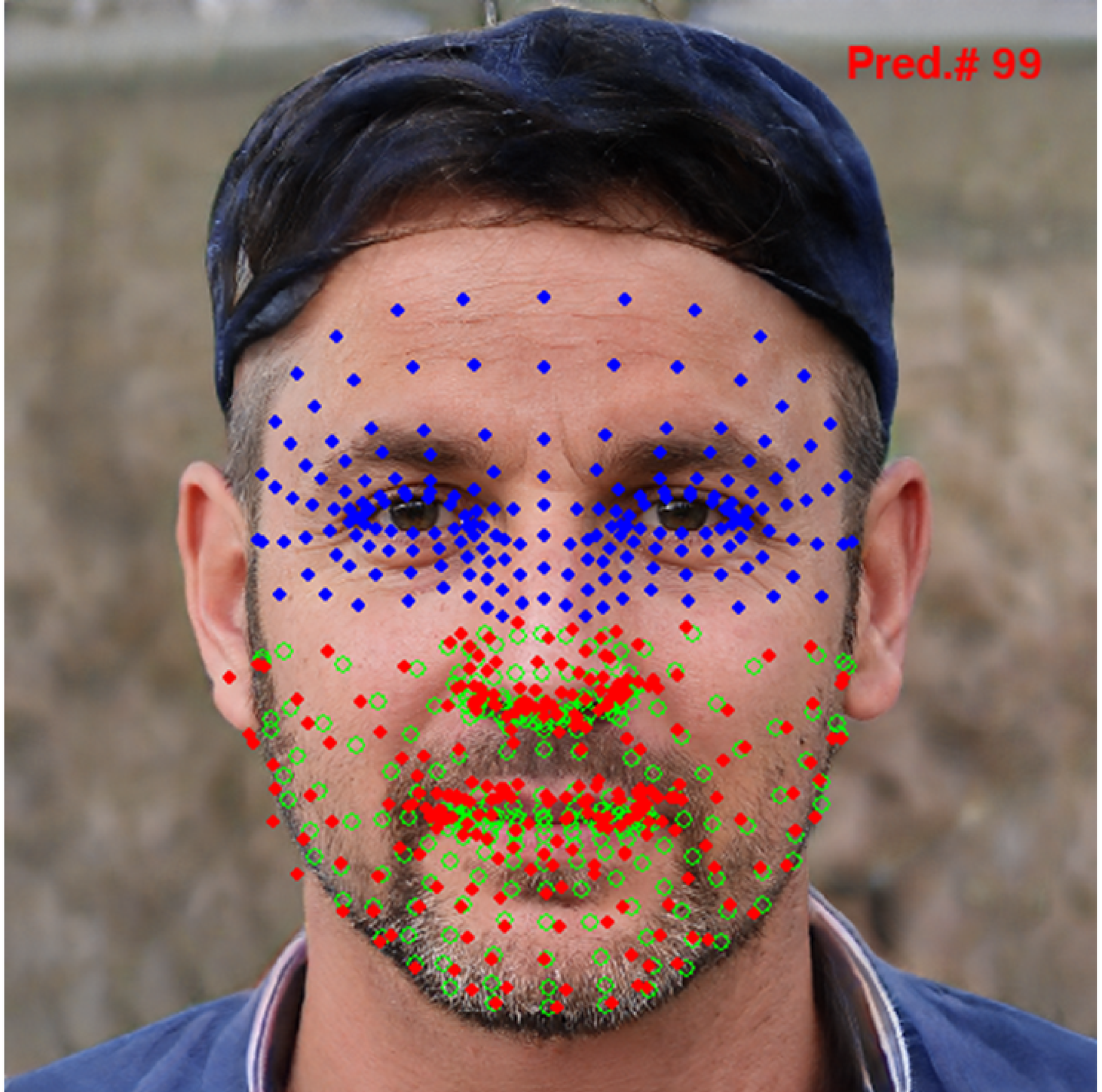
Image Sample : 10

Epochs : 5000

Model : 1 Layer FCNN

Batch size : N

Optimizer: Gradient Descent



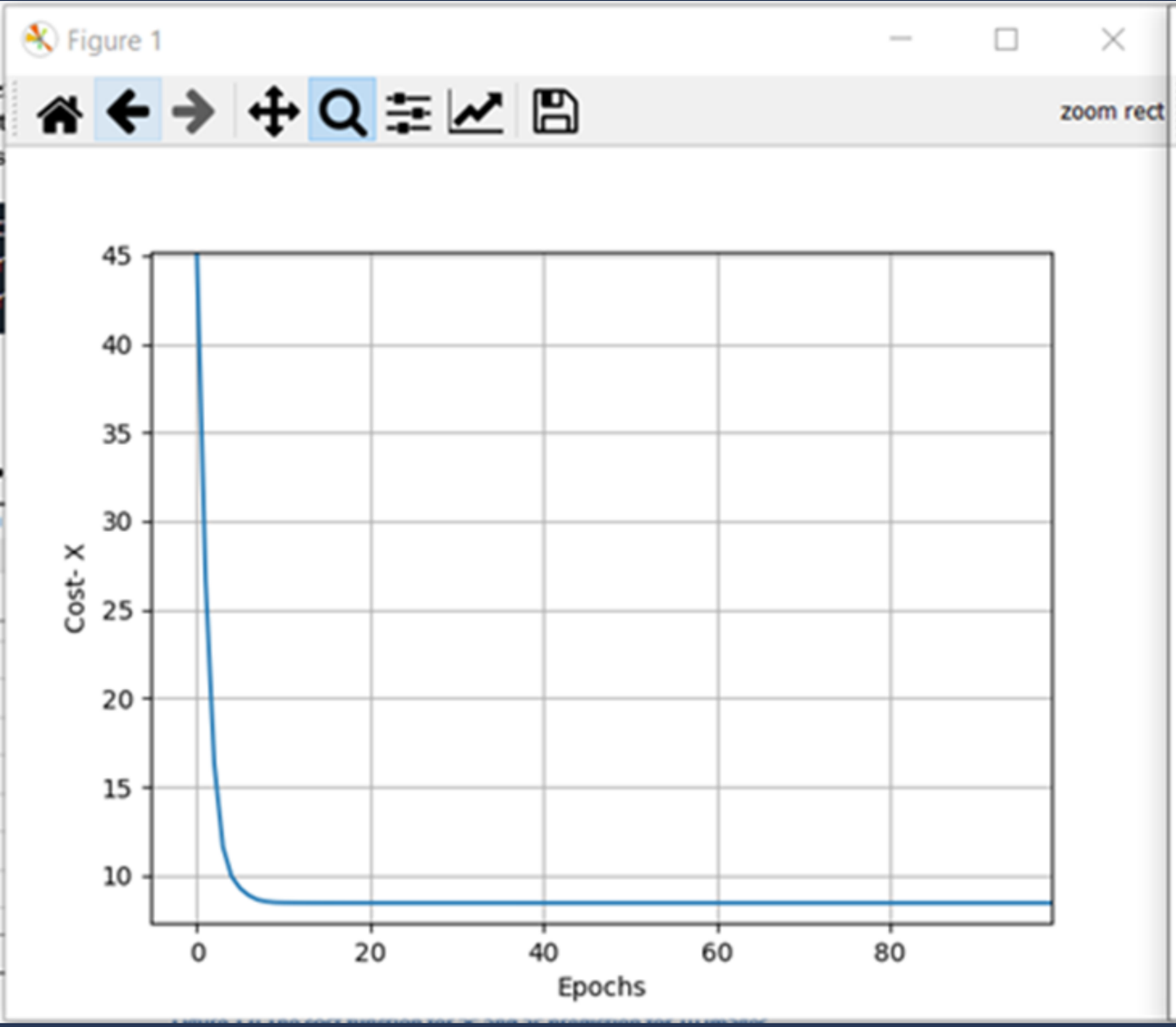


Figure 1
zoom rect
45
40
35
30
25
20
15
10
Cost- X
0
20
40
60
80
Epochs

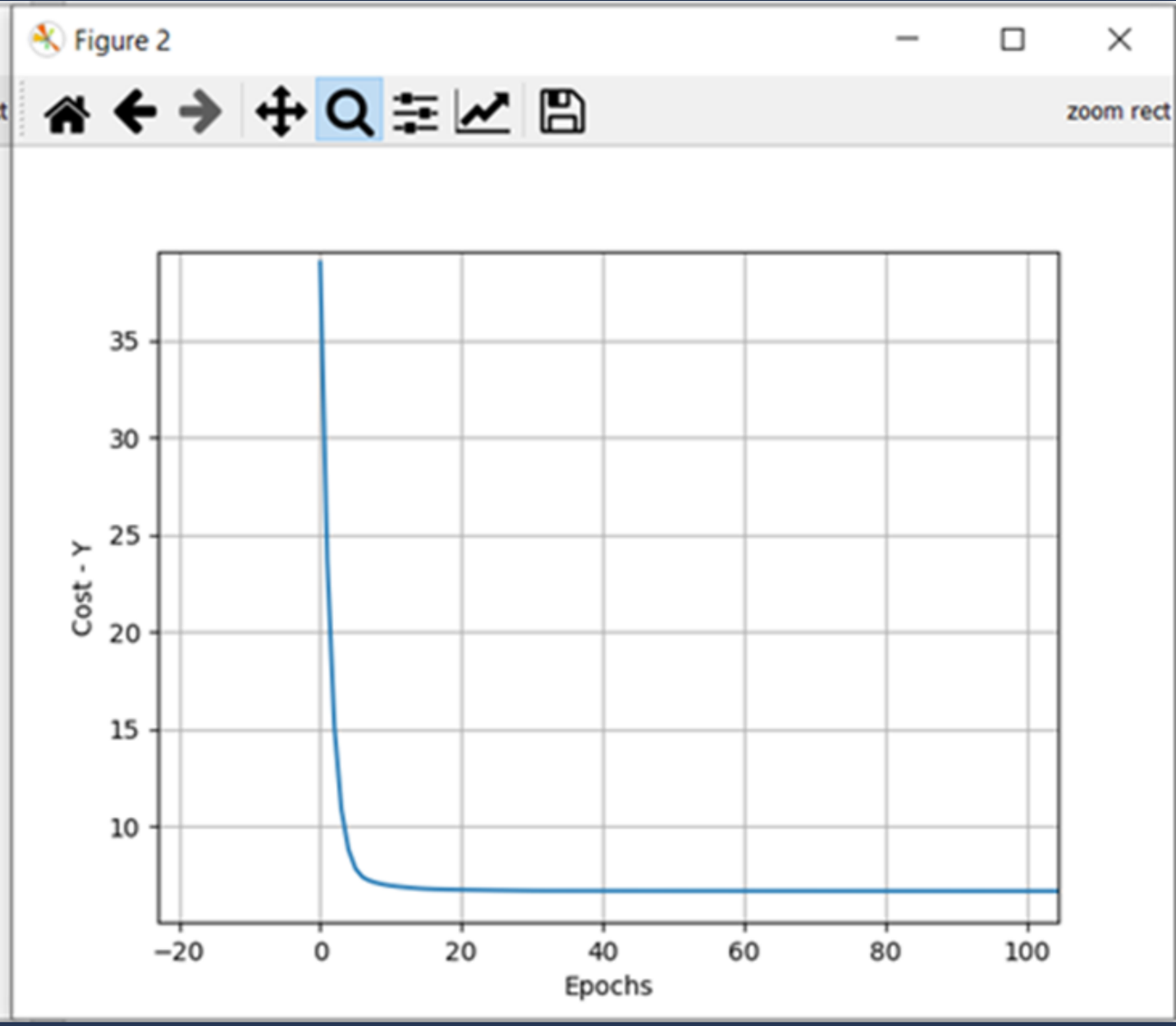


Figure 2
zoom rect
35
30
25
20
15
10
Cost - Y
−20
0
20
40
60
80
100
Epochs

SIMLUATION & RESULTS-3

Image Sample : 100

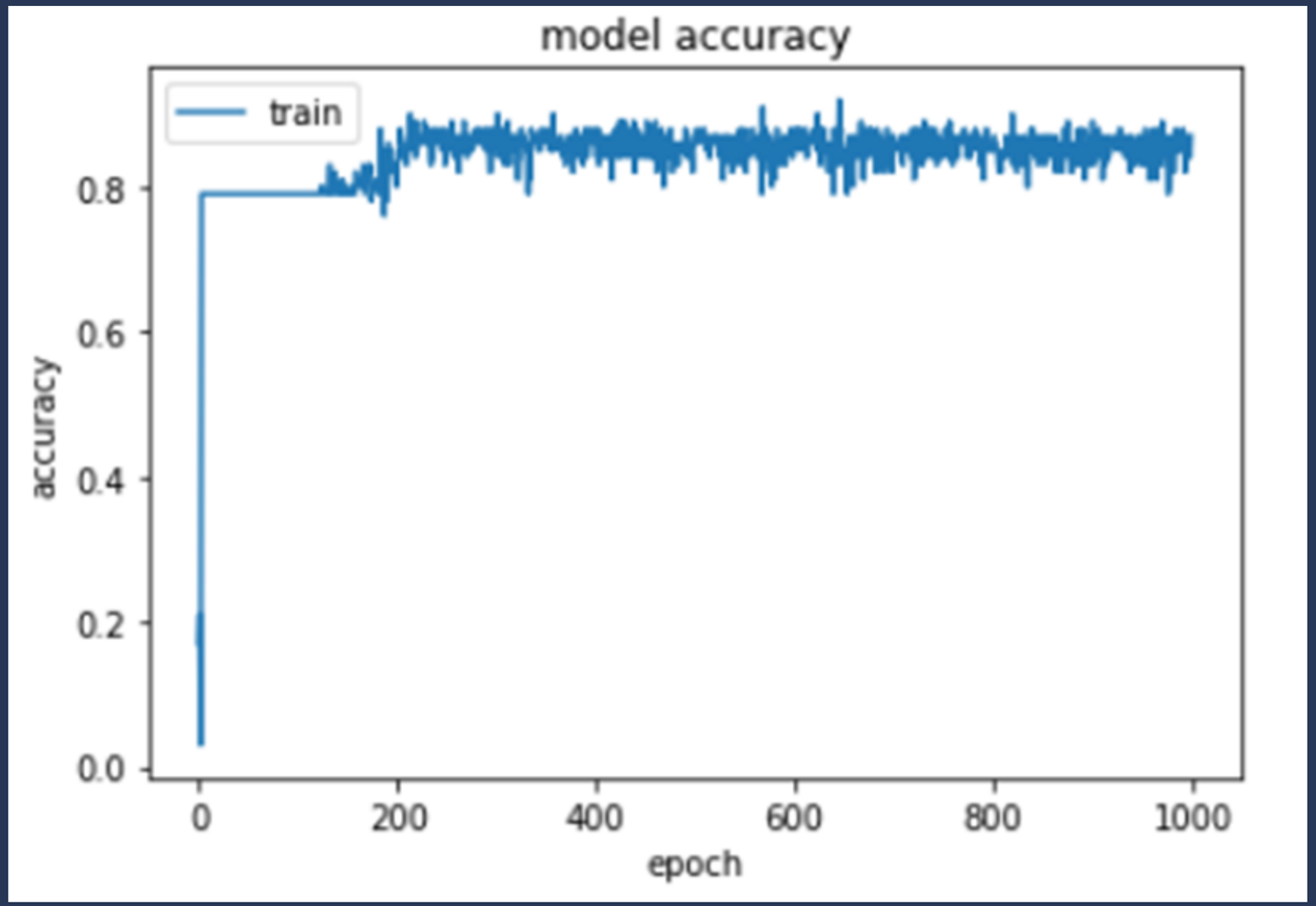
Epochs : 1000

Model : 3 Layer DNN

Batch size : 10

Optimizer: ADAM

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense (Dense)               (None, 100)               19600

 dense_1 (Dense)             (None, 100)               10100

 dense_2 (Dense)             (None, 1000)              101000

 dense_3 (Dense)             (None, 261)               261261

=================================================================
Total params: 391,961
Trainable params: 391,961
Non-trainable params: 0
_________________________________________________________________
```




```
4/4 [==============================] - 0s
Accuracy: 87.00
```

SIMLUATION & RESULTS-3

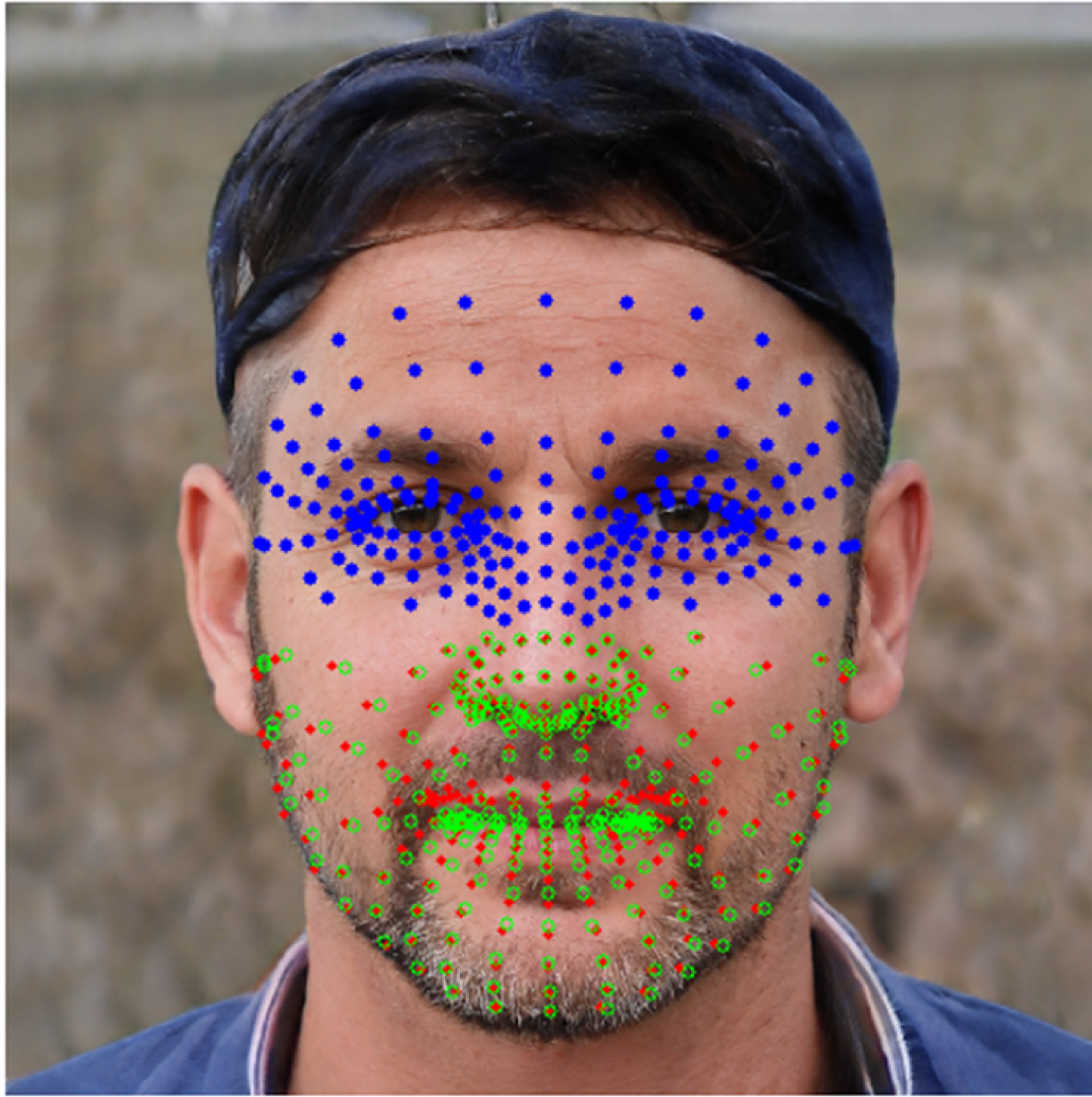
Image Sample : 100

Epochs : 1000

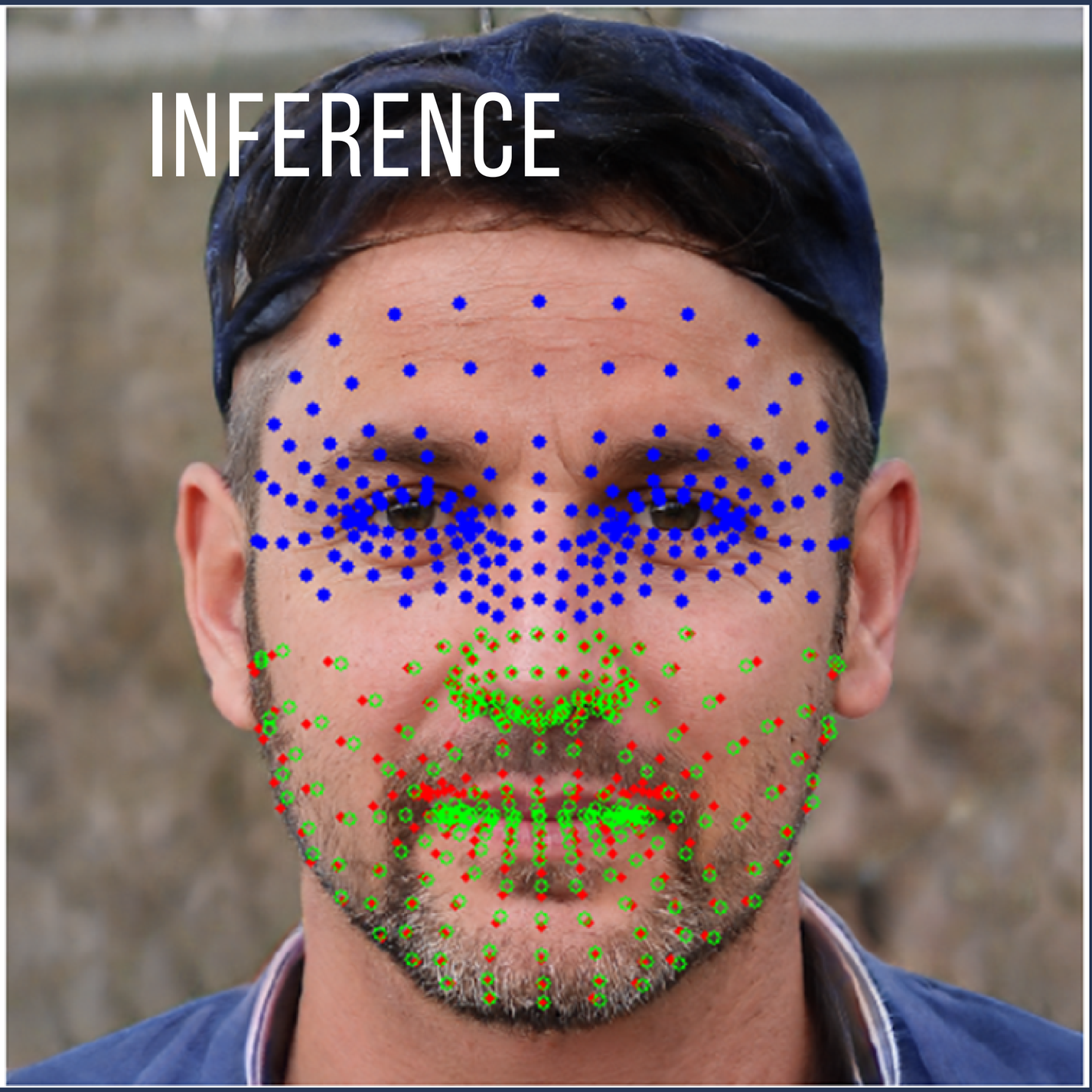
Model : 3 Layer DNN

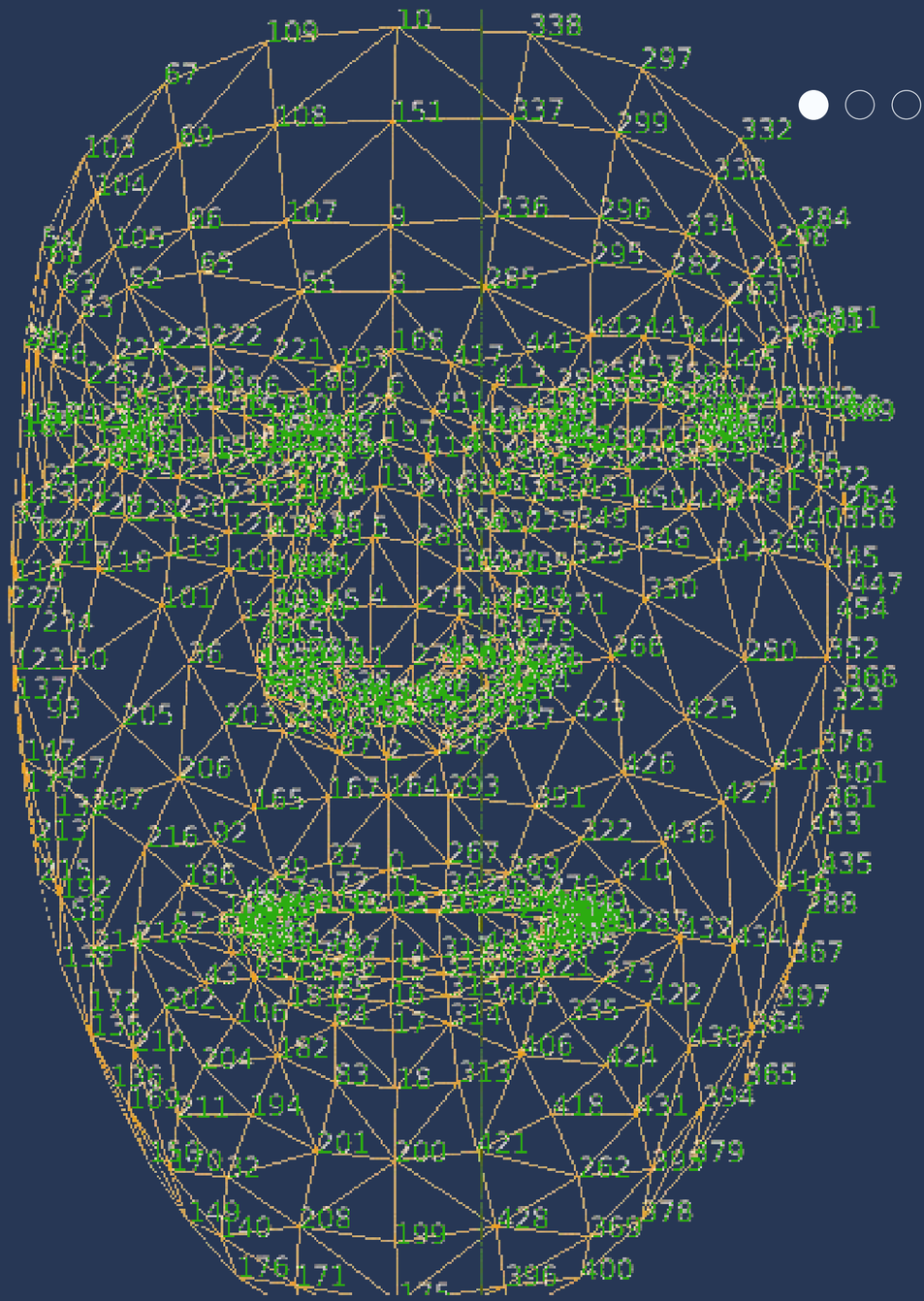
Batch size : 10

Optimizer: ADAM

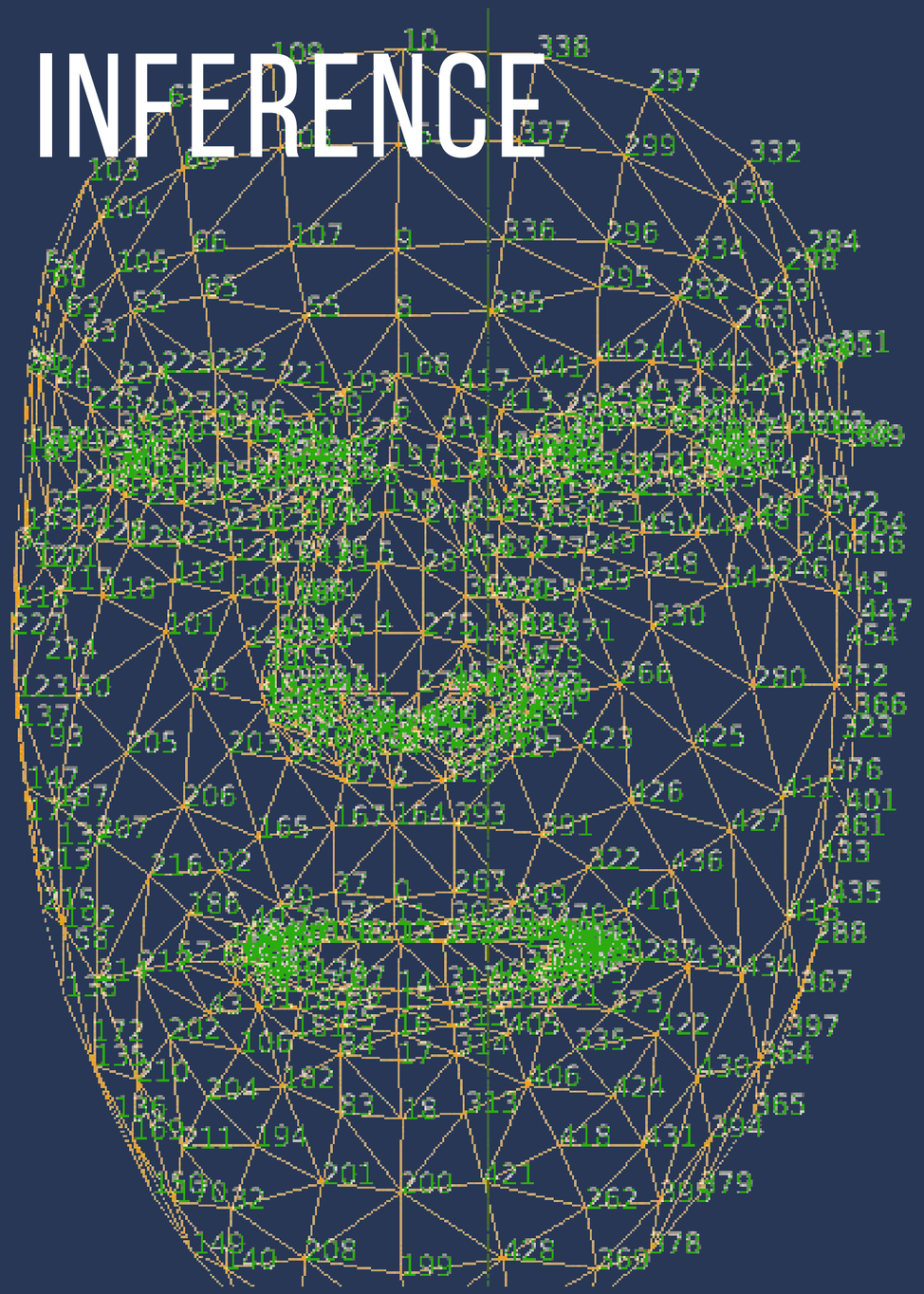


INFERENCE

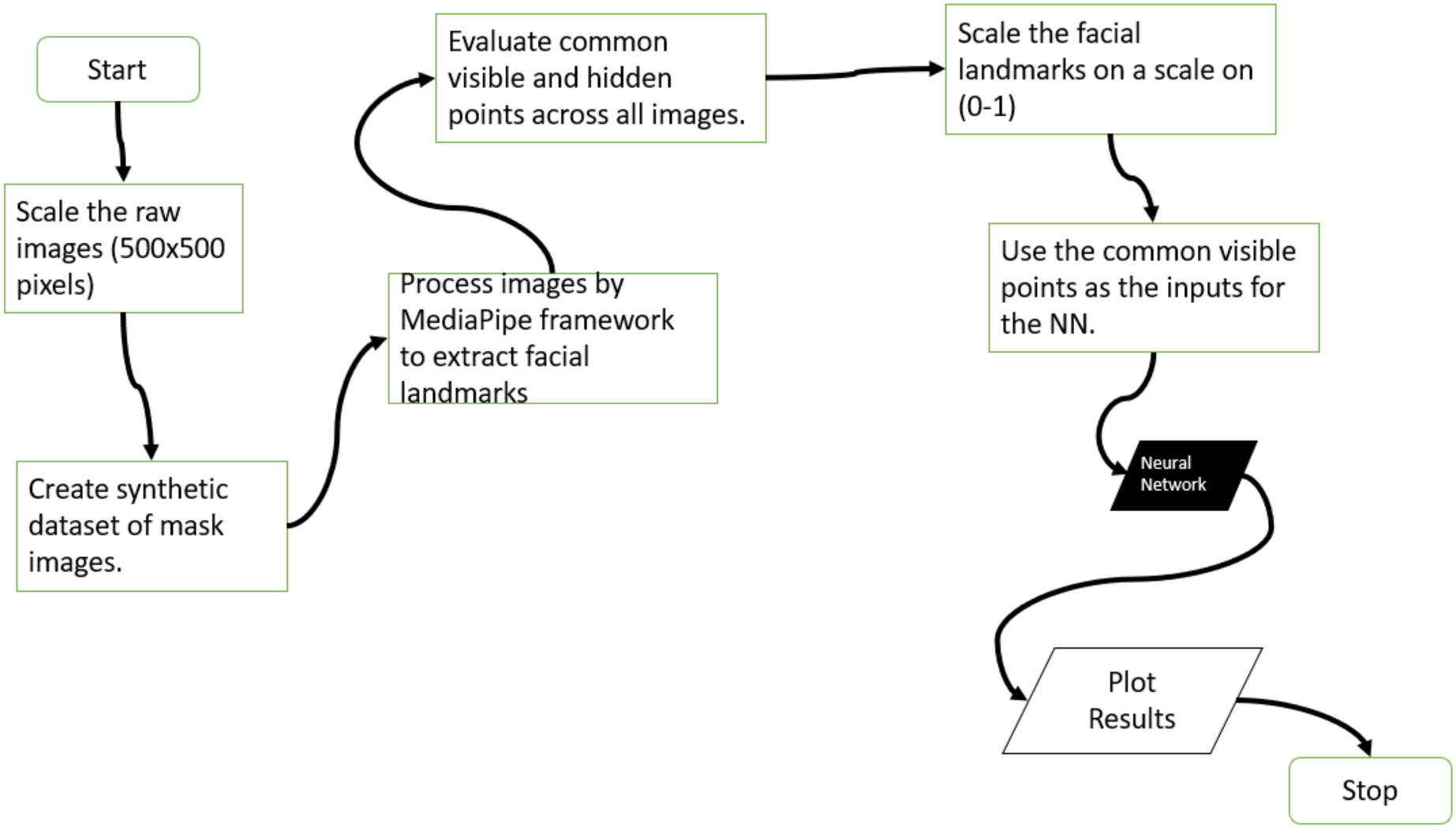



109
10
338
67
297
108
151
337
299
332
103
69
204
107
9
336
296
334
284
105
66
65
295
55
285
168
417
443
442
441
444
222
221
223
189
190
413
414
286
464
465
453
452
451
272
271
273
249
250
329
348
349
330
401
275
279
371
266
280
352
366
205
203
423
425
426
427
401
206
165
164
393
391
422
436
410
216
92
267
186
428
432
434
367
202
106
335
422
397
210
204
182
406
424
430
365
211
194
418
431
201
200
421
262
279
208
199
428
278
171
396
400

INFERENCE



SYNOPSIS OF CODE

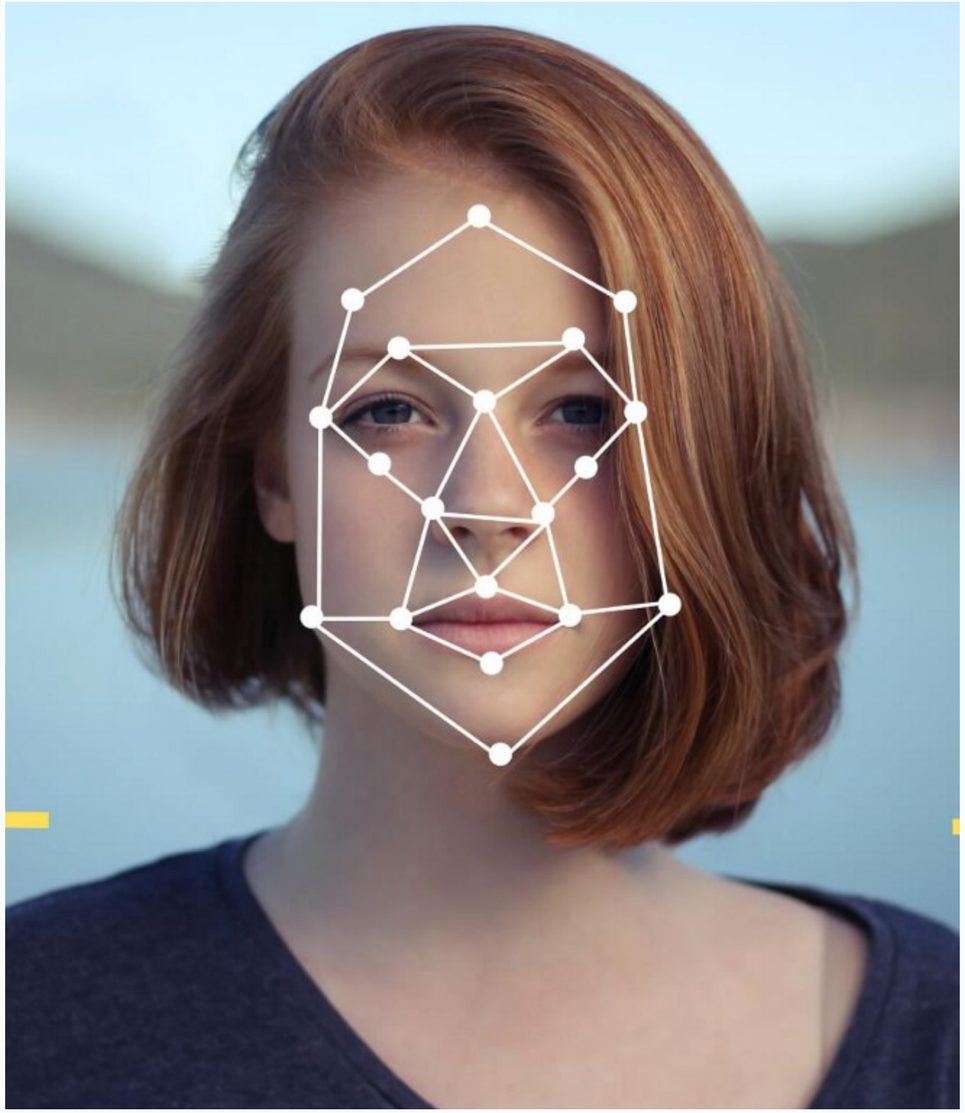


SYNOPSIS OF CODE

```
num_of_imgs_to_process = 10
epochs = 5000
##########################################################################
# Scale and rename images in the raw_images folder
# If already scaled, this portion can be commented out
print("\nRunning - S1_extract_landmarks_original_ima
SCALE.main()

print('-----------------------------------------
##########################################################
# Run the MediaPipe framework to extract all facial
print("Running - S1_extract_landmarks_original_image
EXTRCT.main(img_analyse_count = num_of_imgs_to_proce
          show_res            = False)

##########################################################
# Overlay masks on the images
print("\nRunning - S2_create_masked_dataset")
MASKD.main(img_analyse_count = num_of_imgs_to_process)

##########################################################
# Perform relative scaling of datapoints
print("\nRunning - S3_facial_relative_land")
NORM.main(img_analyse_count = num_of_imgs_to_process)

##########################################################
# Perform prediction of hidden datapoints
print("\nRunning - S4_predict_hid_points_NN")

# saving with image background
PRED.main(img_analyse_count = num_of_imgs_to_process,
          epochs            = epochs,
          show_anim         = True,
          save_file         = True) # for saving predicti

##########################################################
```

FUTURE ASPECTS

- In this research work we were able to determine with high accuracy the facial landmark points which are hidden by the mask.
- In the future extension of this work, we will use these predicted landmark points to generate a face mesh, and over which we will render the face. Thus providing high degree of accuracy in the facial reconstruction,



THANK YOU

Any questions?