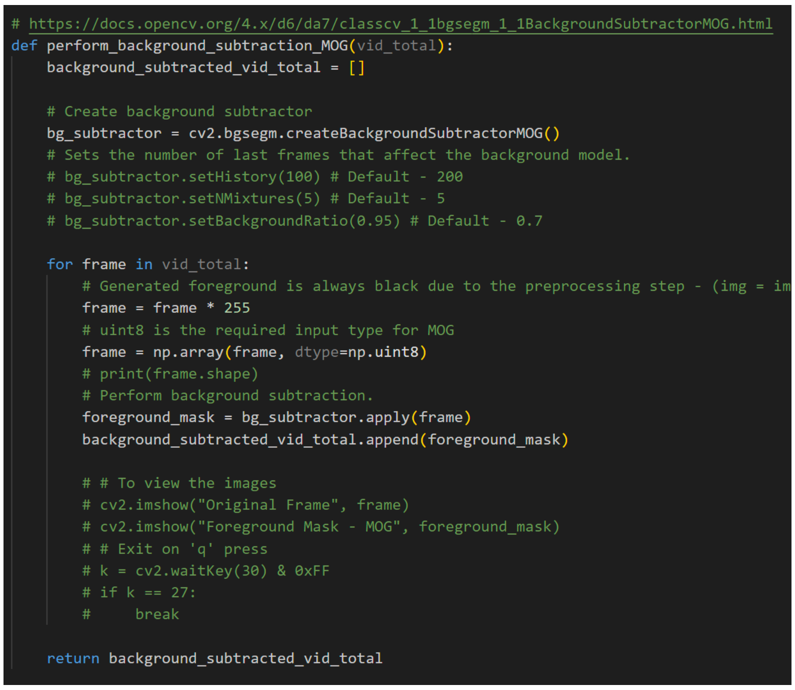
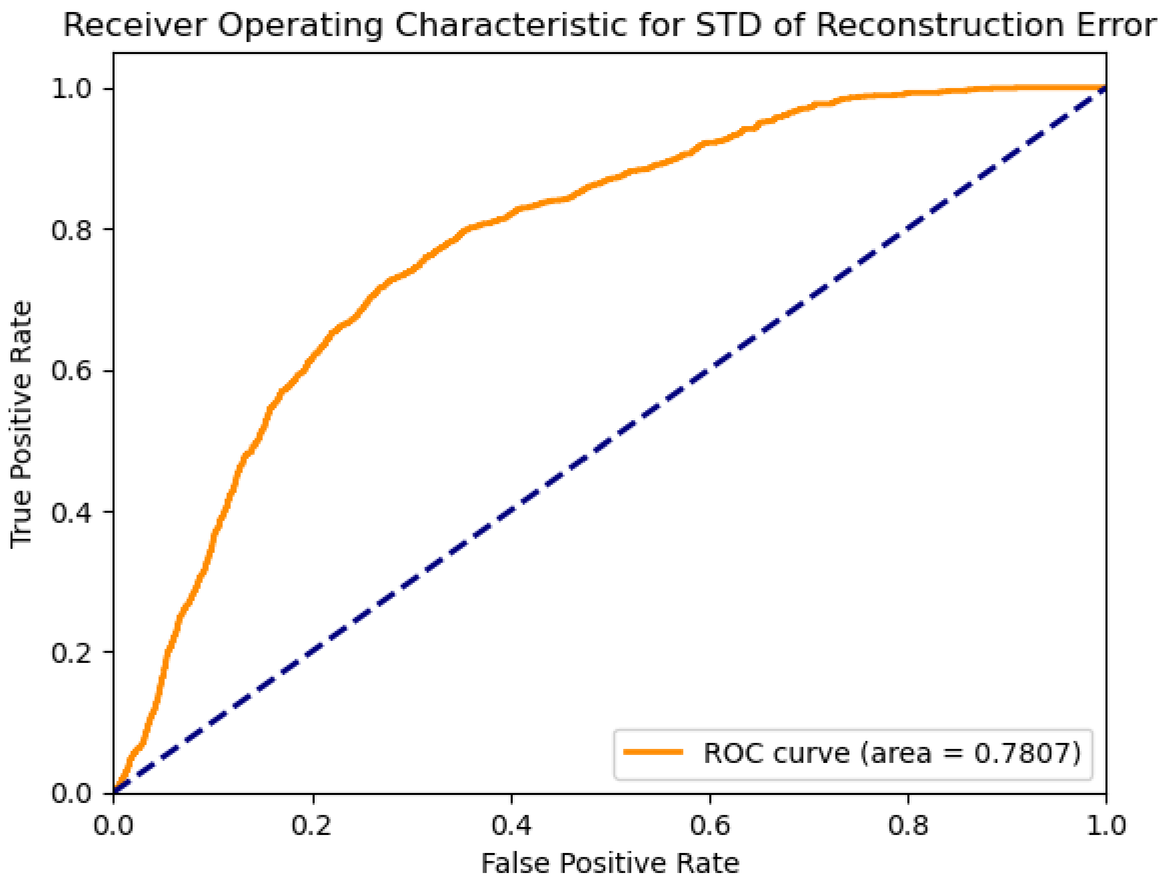


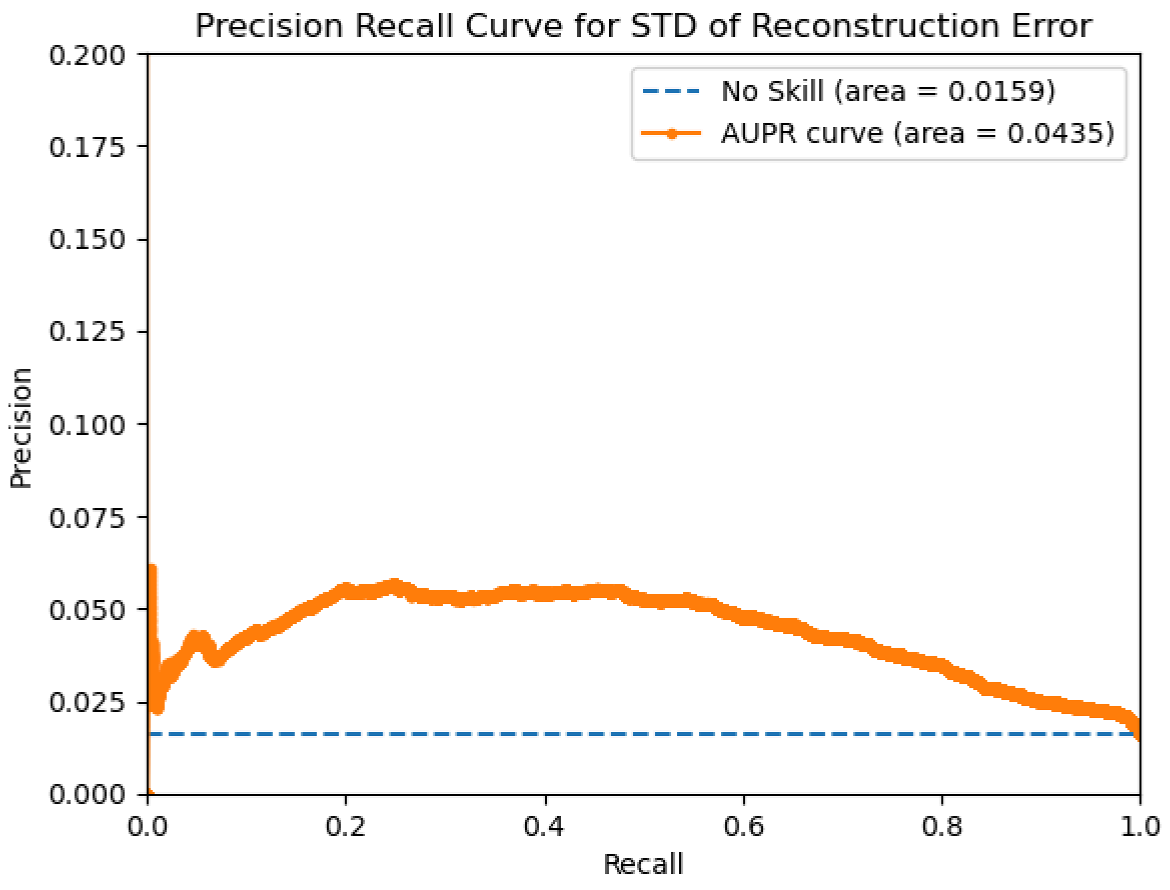
```python
# https://docs.opencv.org/4.x/d6/da7/classcv_1_1bgsegm_1_1BackgroundSubtractorMOG.html
def perform_background_subtraction_MOG(vid_total):
    background_subtracted_vid_total = []

    # Create background subtractor
    bg_subtractor = cv2.bgsegm.createBackgroundSubtractorMOG()
    # Sets the number of last frames that affect the background model.
    # bg_subtractor.setHistory(100) # Default - 200
    # bg_subtractor.setNMixtures(5) # Default - 5
    # bg_subtractor.setBackgroundRatio(0.95) # Default - 0.7

    for frame in vid_total:
        # Generated foreground is always black due to the preprocessing step - (img = im
        frame = frame * 255
        # uint8 is the required input type for MOG
        frame = np.array(frame, dtype=np.uint8)
        # print(frame.shape)
        # Perform background subtraction.
        foreground_mask = bg_subtractor.apply(frame)
        background_subtracted_vid_total.append(foreground_mask)

        # # To view the images
        # cv2.imshow("Original Frame", frame)
        # cv2.imshow("Foreground Mask - MOG", foreground_mask)
        # # Exit on 'q' press
        # k = cv2.waitKey(30) & 0xFF
        # if k == 27:
        #     break

    return background_subtracted_vid_total
```

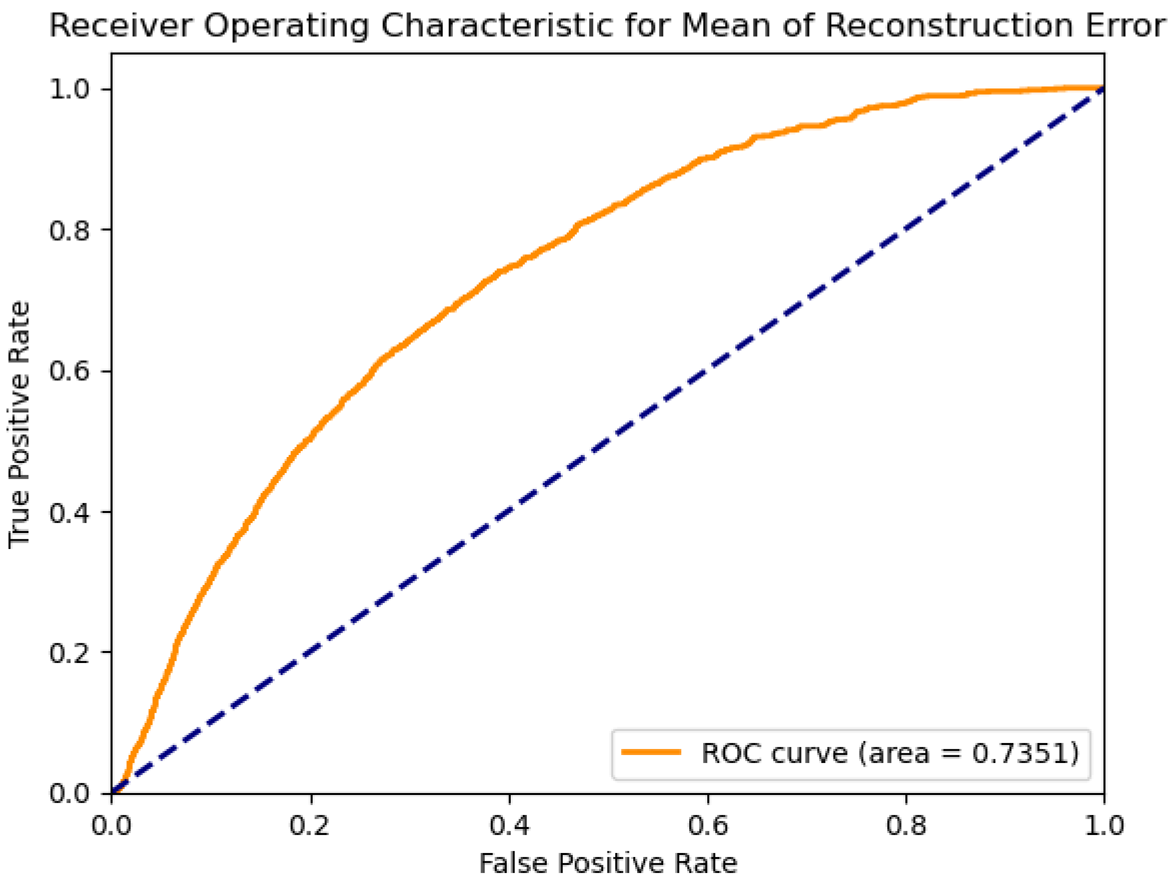
```
Train Dataloader - 48
Test Dataloader - 173

Device Used - cuda

Model Used - Base_3DCAE
Feature Extraction - True
Background Subtraction - True
Background Subtraction Algorithm - MOG
Data Augmentation - False

Window Length = 8
Stride = 1
Fair Comparison = True
Dropout = 0.25
Learning Rate = 0.0002
Num Epochs = 20
Chunk Size = 64
Forward Chunk = 8
Forward Chunk Size = 8
Loss Fn = L1Loss()

Training has Begun
epoch [1/20], loss:9.5353
epoch [2/20], loss:9.5293
epoch [3/20], loss:9.5283
epoch [4/20], loss:9.5280
epoch [5/20], loss:9.5278
epoch [6/20], loss:9.5276
epoch [7/20], loss:9.5276
epoch [8/20], loss:9.5274
epoch [9/20], loss:9.5271
epoch [10/20], loss:9.5268
epoch [11/20], loss:9.5261
epoch [12/20], loss:9.5254
epoch [13/20], loss:9.5245
epoch [14/20], loss:9.5227
epoch [15/20], loss:9.5216
epoch [16/20], loss:9.5207
epoch [17/20], loss:9.5199
epoch [18/20], loss:9.5190
epoch [19/20], loss:9.5183
c:\Users\abdul\anaconda3\envs\fyp_base_paper_2\lib\site-packages\numpy\lib\npyio.py:528: V
isibleDeprecationWarning: Creating an ndarray from ragged nested sequences (which is a lis
t-or-tuple of lists-or-tuples-or ndarrays with different lengths or shapes) is deprecated.
If you meant to do this, you must specify 'dtype=object' when creating the ndarray.
  arr = np.asanyarray(arr)
```

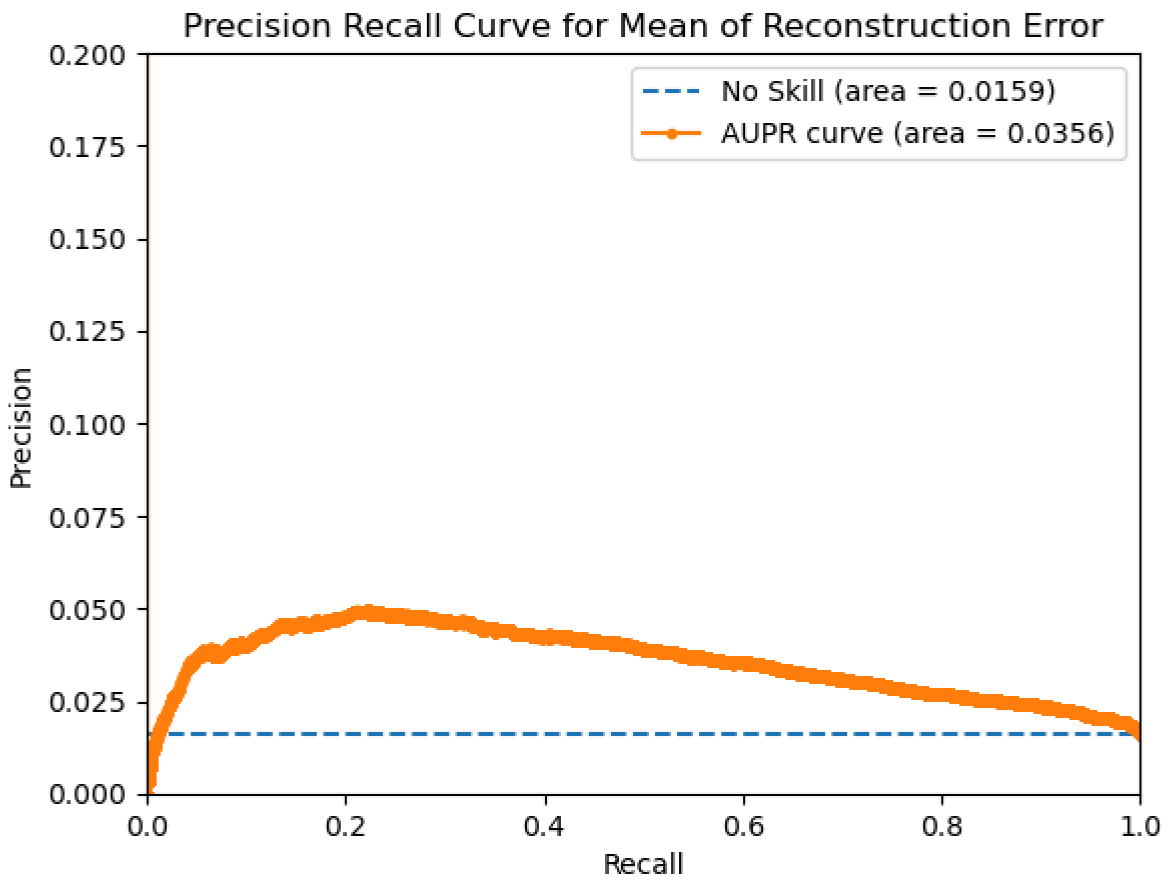
```
epoch [20/20], loss:9.5175
Training has Completed

Forward pass occuring
Forward pass completed

Thermal_T3_2024-03-14-02-44-46

----------------------------------
STD Global Classification Results
TPR 0.725, FPR 0.277, Precision 0.041, Recall 0.725
tn 48004, fp 18385, fn 295, tp 777
std_AUROC  0.781
----------------------------------
----------------------------------
Mean Global Classification Results
TPR 0.725, FPR 0.376, Precision 0.030, Recall 0.725
tn 41431, fp 24958, fn 295, tp 777
mean_AUROC 0.735
----------------------------------
```

```
d:\Abdul Rasheed NITT\Academics\Eigth Semester\FYP\Implementation\FallDetection\Code\funct
ions.py:250: RuntimeWarning: Mean of empty slice
  final_performance_mean = np.nanmean(video_metrics, axis=0)  # get the mean performance a
cross all videos
c:\Users\abdul\anaconda3\envs\fyp_base_paper_2\lib\site-packages\numpy\lib\nanfunctions.p
y:1670: RuntimeWarning: Degrees of freedom <= 0 for slice.
  var = nanvar(a, axis=axis, dtype=dtype, out=out, ddof=ddof,
```




Precision Recall Curve for STD of Reconstruction Error
No Skill (area = 0.0159)
AUPR curve (area = 0.0435)
Precision
Recall
0.000
0.025
0.050
0.075
0.100
0.125
0.150
0.175
0.200
0.0
0.2
0.4
0.6
0.8
1.0


Receiver Operating Characteristic for Mean of Reconstruction Error
ROC curve (area = 0.7351)
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0


Precision Recall Curve for Mean of Reconstruction Error
No Skill (area = 0.0159)
AUPR curve (area = 0.0356)
Precision
Recall
0.200
0.175
0.150
0.125
0.100
0.075
0.050
0.025
0.000
0.0
0.2
0.4
0.6
0.8
1.0