01

# SE(3)-UNET

**University of Milan**

**Mattia Ferraretto [00072A]**

A.Y. 2023/2024

**Visione Artificiale**

02

# Table of Contents


Problem statement

SE(3)-Transformer

FaceScape dataset

3D - Pooling

3D - Upsampling

Model architecture

Experiment 1 (BCE)

Experiment 2 (FL)

03

# Problem statement

- The problem consists in defining a novel architectures resilient to 3D roto-translation
- In others words, we want a network  that is equivariant  for every transformation in a given abstract group
- Objective: point cloud segmentation, predicting whether a point belong to a class
- Idea: replacing classic convolution block in a U-net architecture by SE(3)-Transformer block

04

# SE(3)-Transformers

$$\mathbf{f}_{\text{out},i}^{\ell} = \underbrace{\mathbf{W}_V^{\ell\ell}\mathbf{f}_{\text{in},i}^{\ell}}_{\text{③ self-interaction}} + \sum_{k\geq 0}\sum_{j\in\mathcal{N}_i\setminus i} \underbrace{\alpha_{ij}}_{\text{① attention}} \underbrace{\mathbf{W}_V^{\ell k}\left(\mathbf{x}_j - \mathbf{x}_i\right)\mathbf{f}_{\text{in},j}^{k}}_{\text{② value message}} \tag{3}$$

$$\alpha_{ij} = \frac{\exp(\mathbf{q}_i^{\top}\mathbf{k}_{ij})}{\sum_{j'\in\mathcal{N}_i\setminus i}\exp(\mathbf{q}_i^{\top}\mathbf{k}_{ij'})}, \quad \mathbf{q}_i = \bigoplus_{\ell\geq 0}\sum_{k\geq 0}\mathbf{W}_Q^{\ell k}\mathbf{f}_{\text{in},i}^{k}, \quad \mathbf{k}_{ij} = \bigoplus_{\ell\geq 0}\sum_{k\geq 0}\mathbf{W}_K^{\ell k}(\mathbf{x}_j - \mathbf{x}_i)\mathbf{f}_{\text{in},j}^{k} \tag{4}$$

04

# FaceScape Dataset

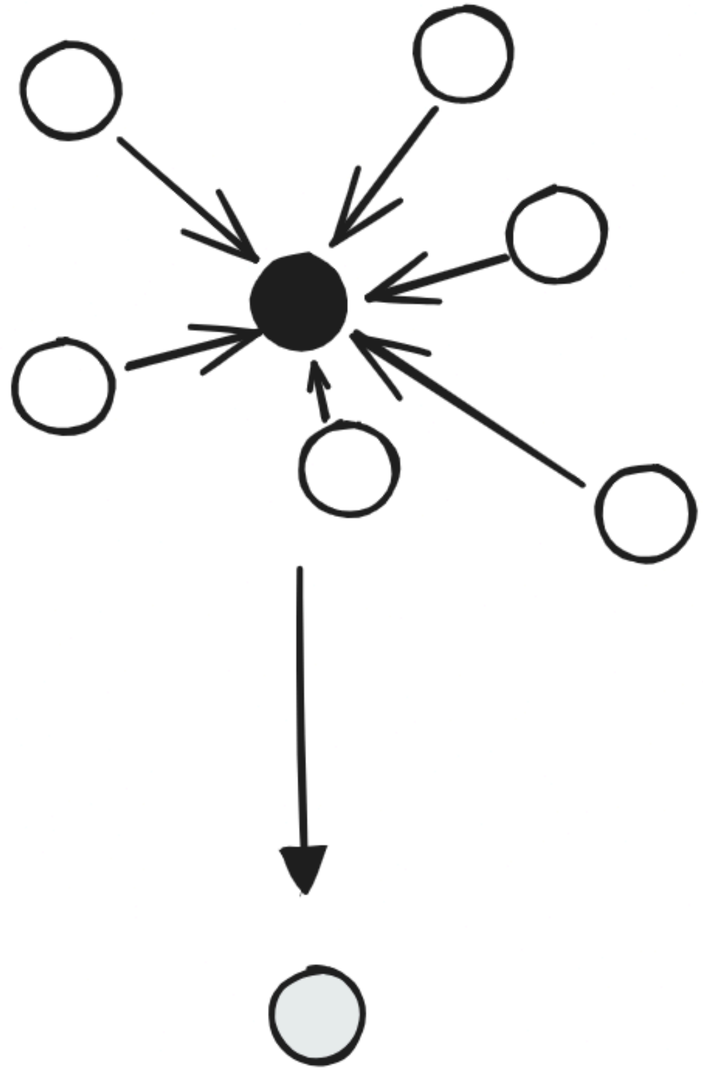
1. **Acquisition**:
   - 18,760 high resolution 3D faces with 20 different expressions
   - Acquired by using a multi-view 3D reconstruction system composed by 68 DSLR cameras
   - Resulting in point clouds having 8192 points and 68 corresponding landmark
2. **Pre-processing**:
   - Introducing random 3D roto-translation then realigned by using ICP
   - Simply introducing random t3D roto-translation
3. **Point cloud dimensionality reduction**:
   - Reduced from 8192 to 2048 points by using Farthest Point Sampling (FPS)
   - Heatmap re-computed via a Gaussian function

05

# 3D - Pooling

3D poling aims to apply classic pooling operation in the 3D filed:

1. A certain number of points are selected, according to a pooling ratio
2. On the basis of each point’s neighborhood features are aggregated (by mean or max)
3. Points not selected are discarded
4. The result is a point cloud reduced by the pooling ratio defined



05

# 3D - Upsampling

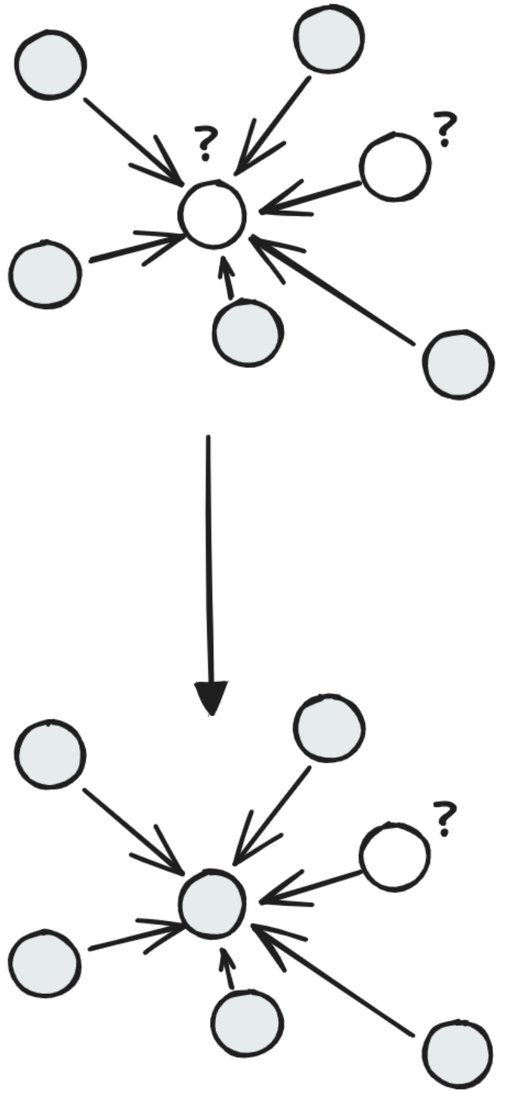
3D upsampling consists in reconstructing the point cloud resolution at the previous step by using IDW:

1. For each features to estimate weights are computed as:

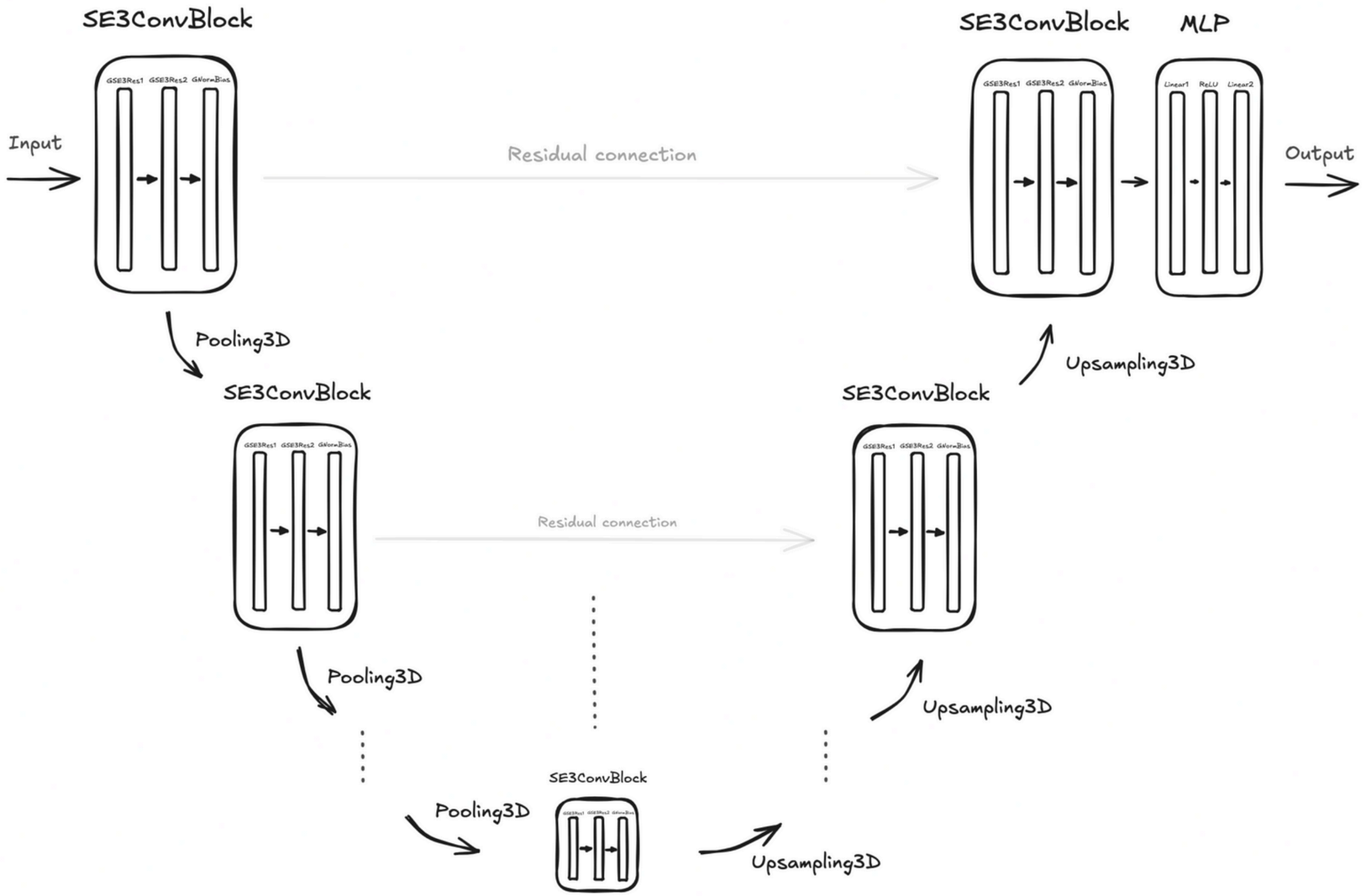
$$w_j = \frac{1}{d(x_i,\, x_j)^p}$$

2. Then features are estimated as the weighted average:

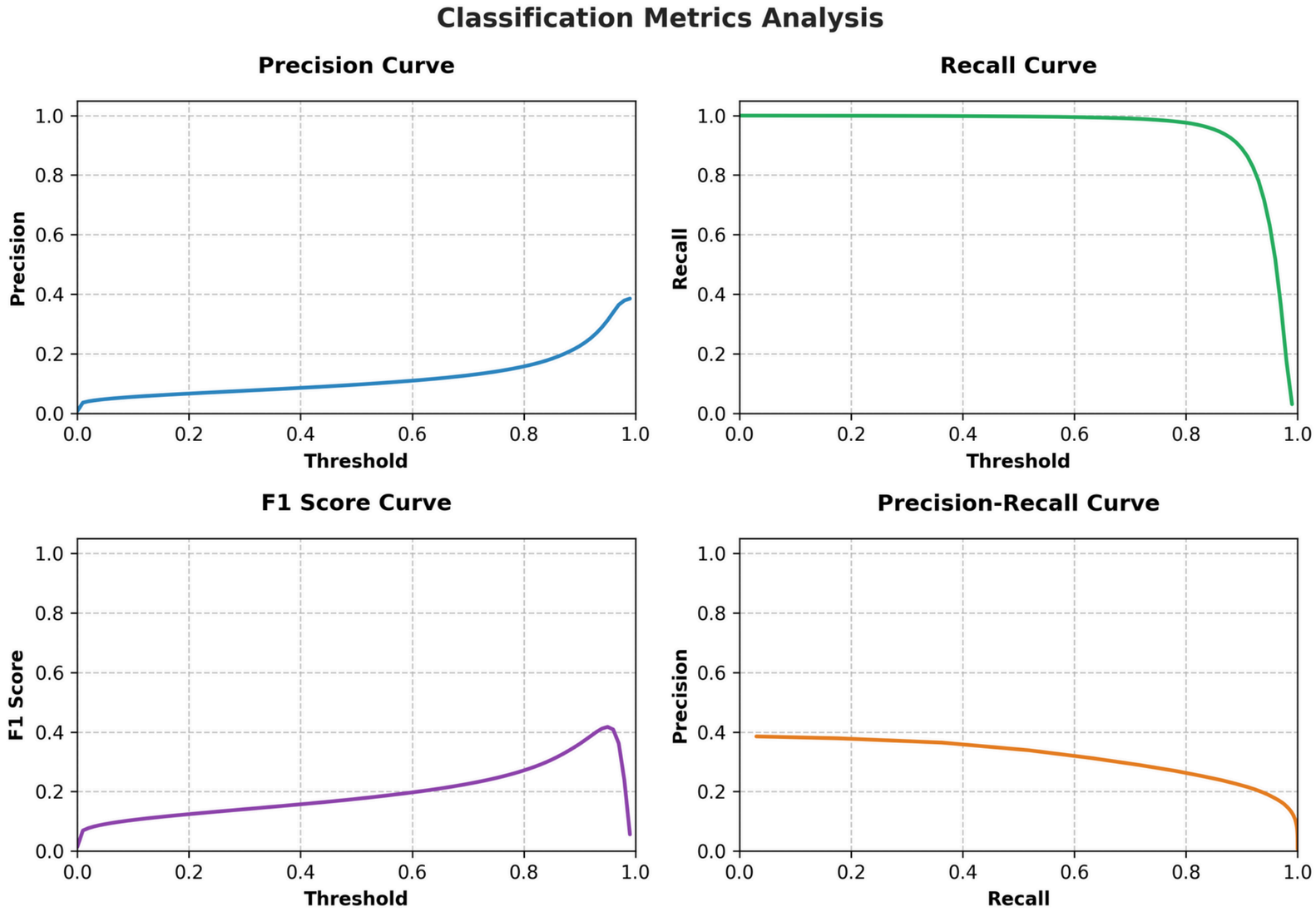
$$f_i = \frac{\sum_{j \in \mathcal{N}_i} w_j f_j}{\sum_{j \in \mathcal{N}_i} w_j}$$



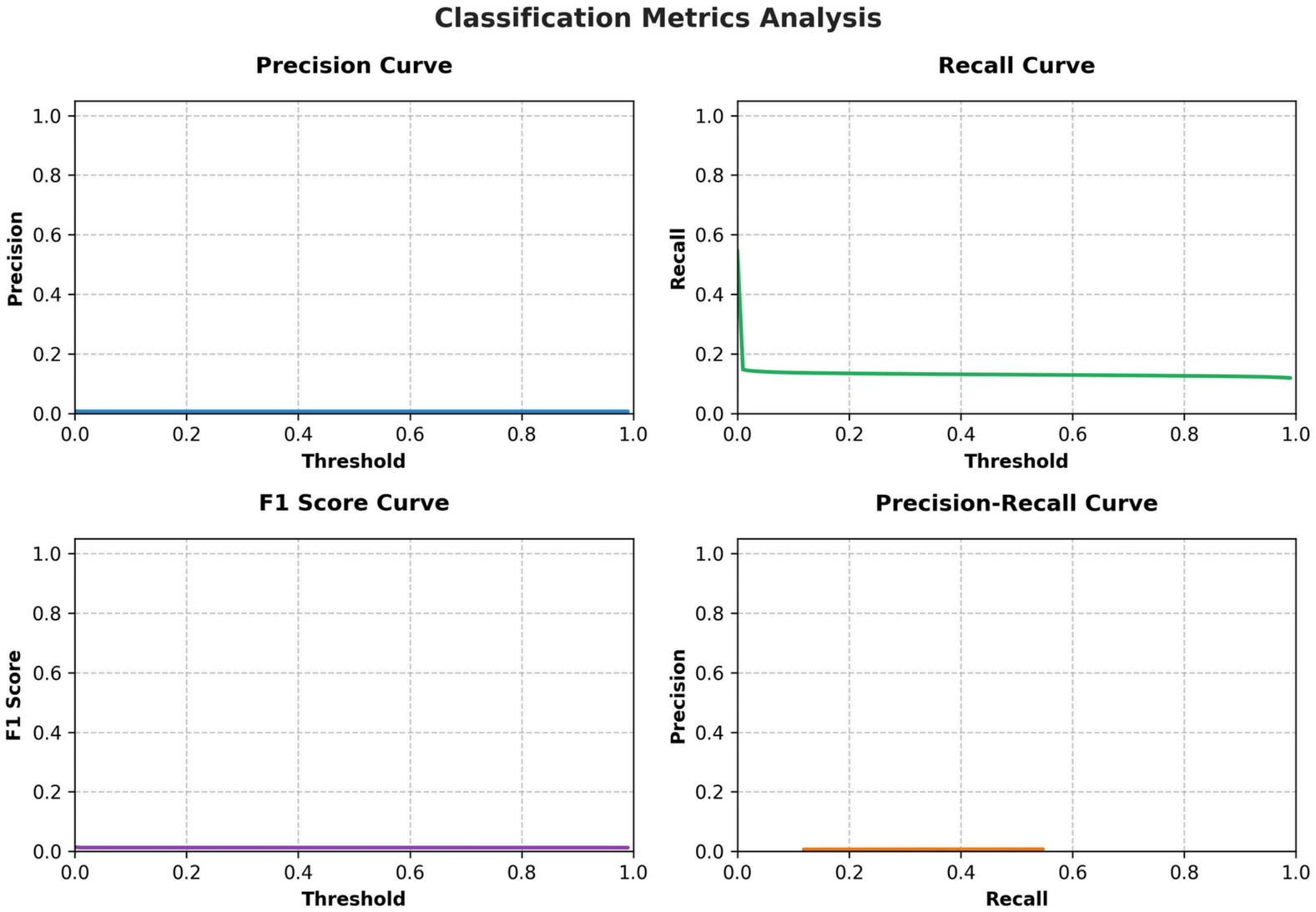
# Model architecture



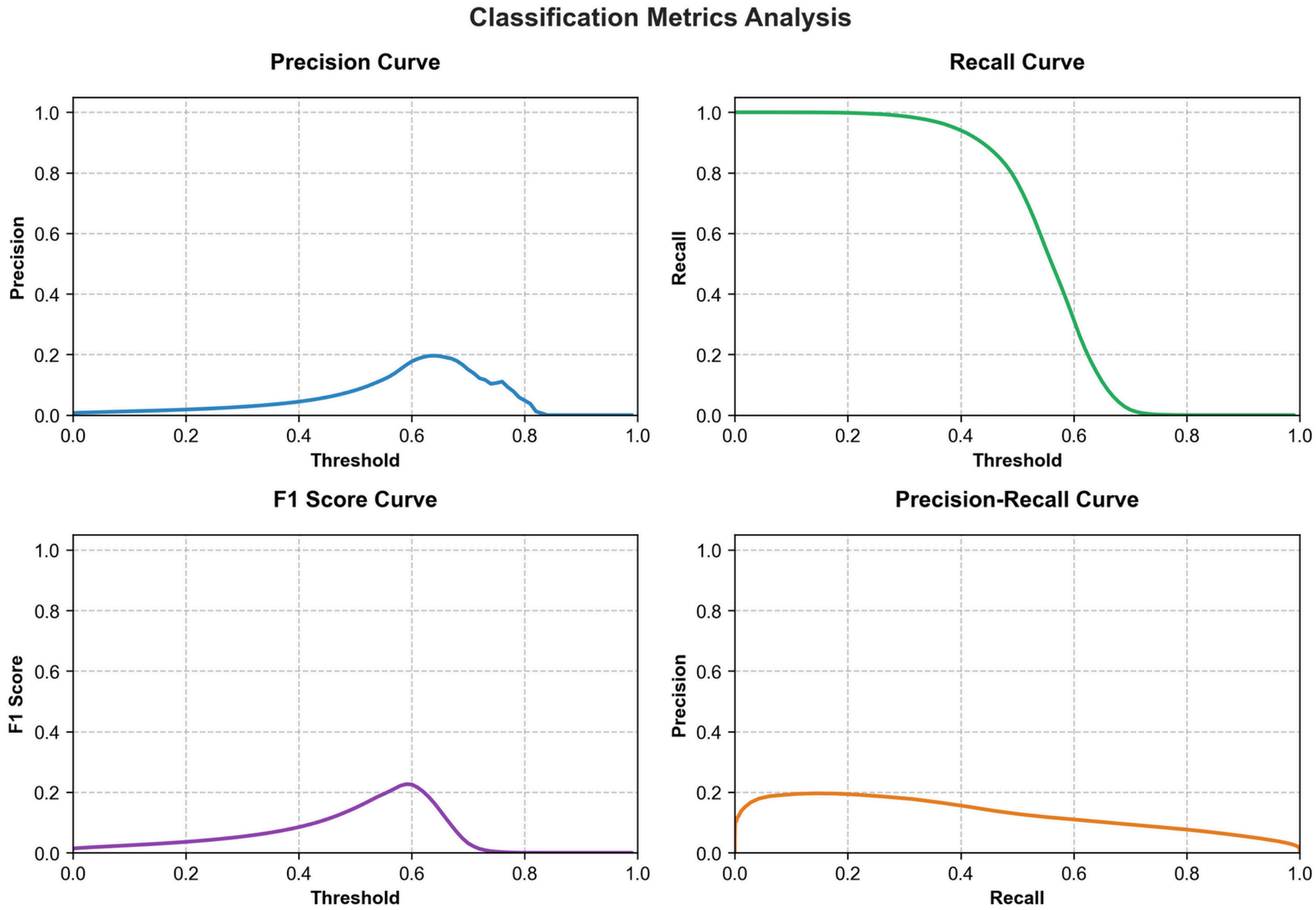
# Experiment 1 (BCE - ICP)



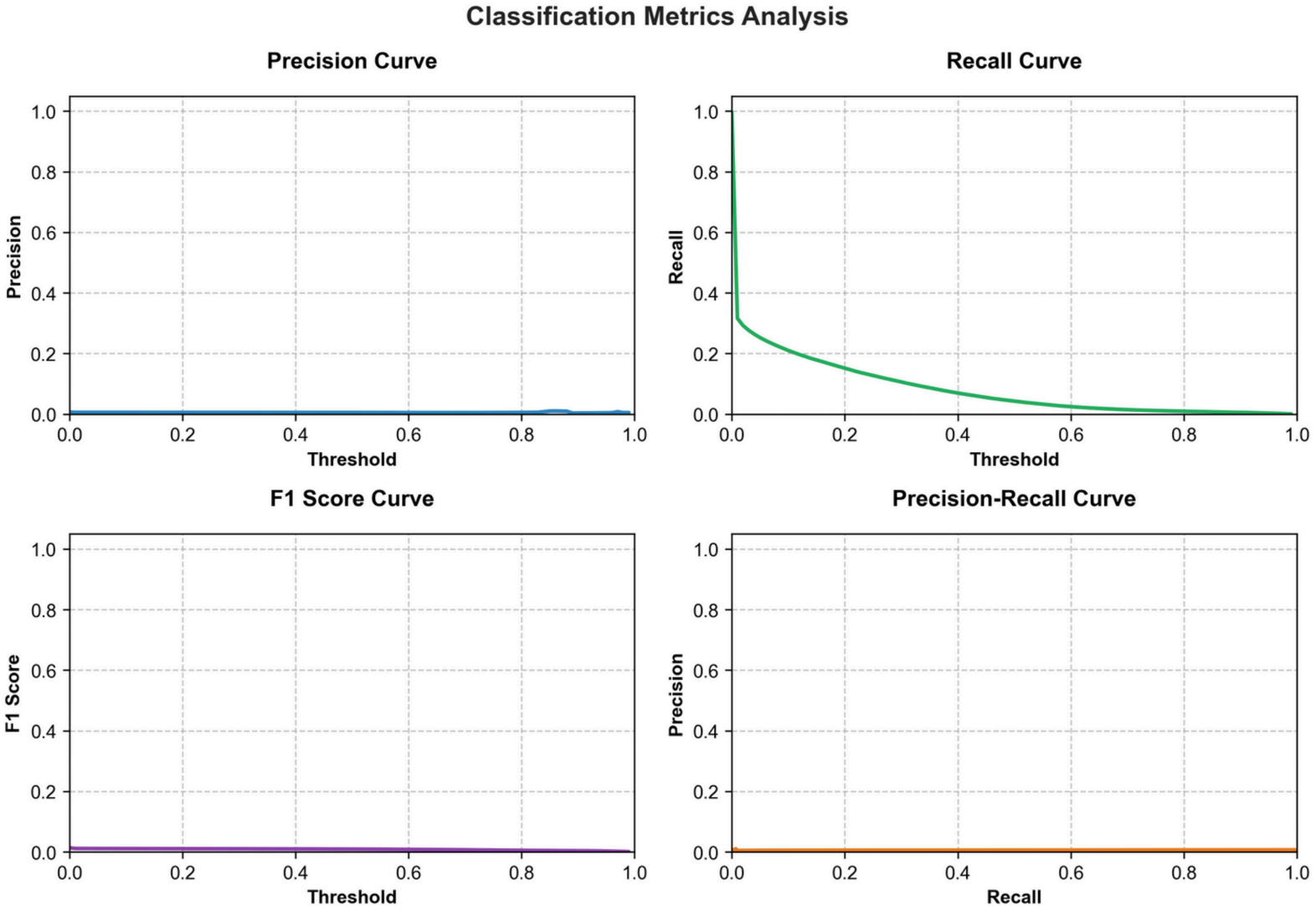
# Experiment 1 (BCE - ST)



# Experiment 2 (FL - ICP)



# Experiment 2 (FL - ST)



# Conclusion

- Using BCE over a dataset pre-processed by ICP, we obtain some results but not sufficient for stating that the model works well
- Using BCE over a broken dataset, performance degradates
- Using FL on a pre-processed dataset from ICP results in worse performance than using BCE
- Using FL on a broken dataset, performance are completely absent

**In conclusion the model doesn’t work! :(**