CS5284 Final Project: Activity Recognition with Graph NN

Names: Chong Jun Rong Brian, Ng Wei Jie Brandon, Parashara Ramesh
Student ID: A0290882U, A0184893L, A0285647M

Project Motivation

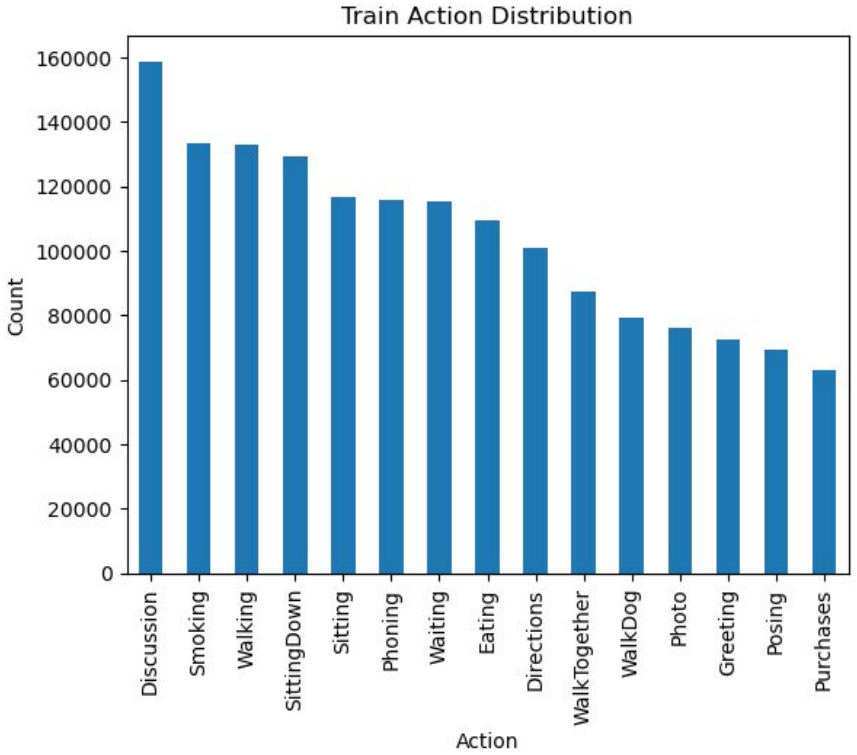
- Predicting 3D Human Poses from 2D Keypoints is a critical task for applications such as motion capture or activity recognition
- Traditional methods often relies on Computer Vision or Computer Graphics methods to reliably perform this task but there might be edge cases where poses are unable to be detected.
- Classifying human poses are also important tasks for human surveillance and healthcare.
- Skeleton poses are naturally an undirected graph where nodes are positions of the human features such as arms, hands and legs and edges can represent bones or connectivity between two nodes.
- Therefore, we will want to explore utilizing GNNs to improve 3D pose estimation and activity recognition.

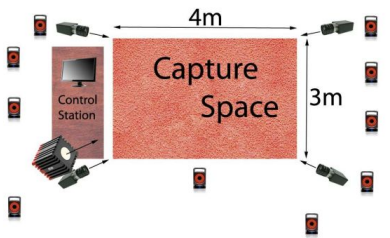
Project Description / Task Formulation

- **Objective**: To predict 3D Human Poses and Activity Labels
- **Datasets**:
  - Human 3.6M Dataset
  - Custom Dataset
- **Models**:
  - **SimplePose**: Basic Neural Network Architecture
  - **SimplePoseGNN**: Graph Convolutional Neural Network Architecture
  - **SimplePoseGAT**: Vanilla Graph Transformer Neural Network Architecture
  - **SimplePoseTAG**: Topological Adaptive Graph Convolutional Neural Network Architecture

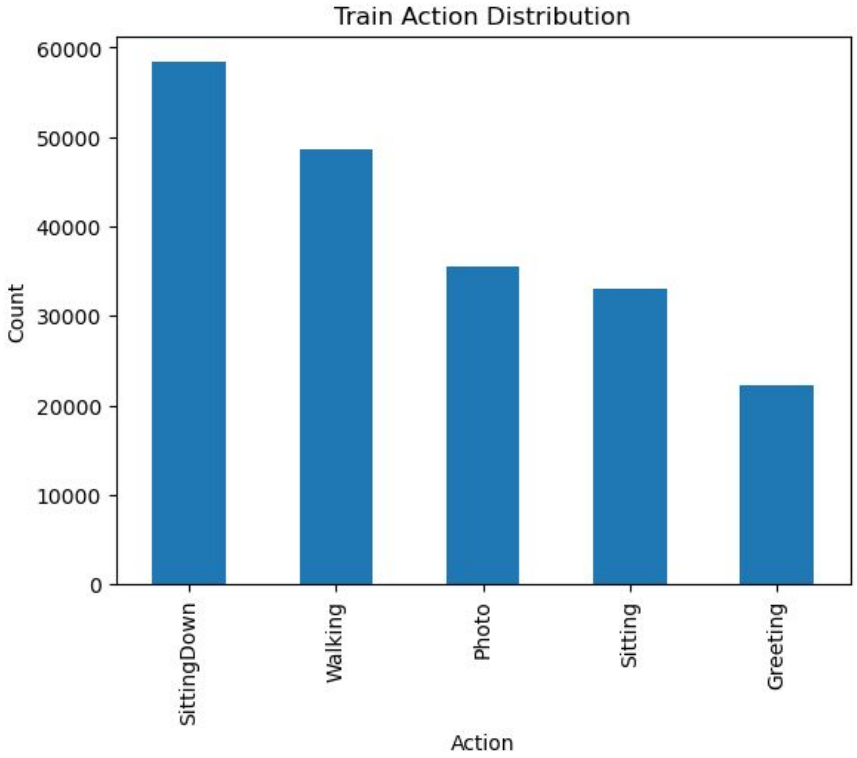
Dataset: Human 3.6M

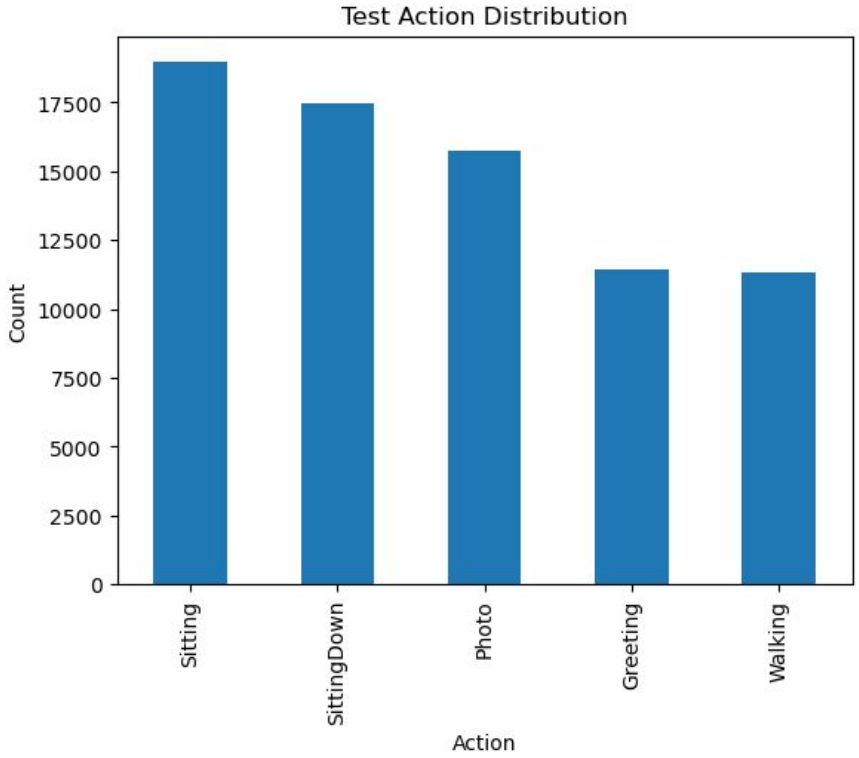
- Contains 3.6 million human poses captured from 11 professional actors in 17 daily scenarios
- Utilises 3D motion capture system (10 cameras) to track reflective markers on body to label poses
- Captures actions of subjects starting from initial pose and with some freedom to move naturally





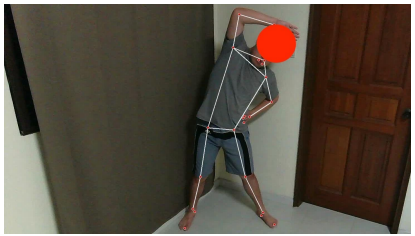
Dataset: Modified Human 3.6m Dataset Distribution

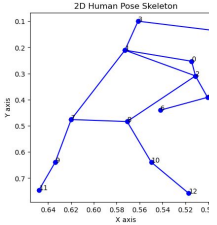


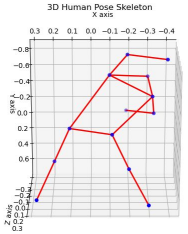


Dataset: Custom Dataset

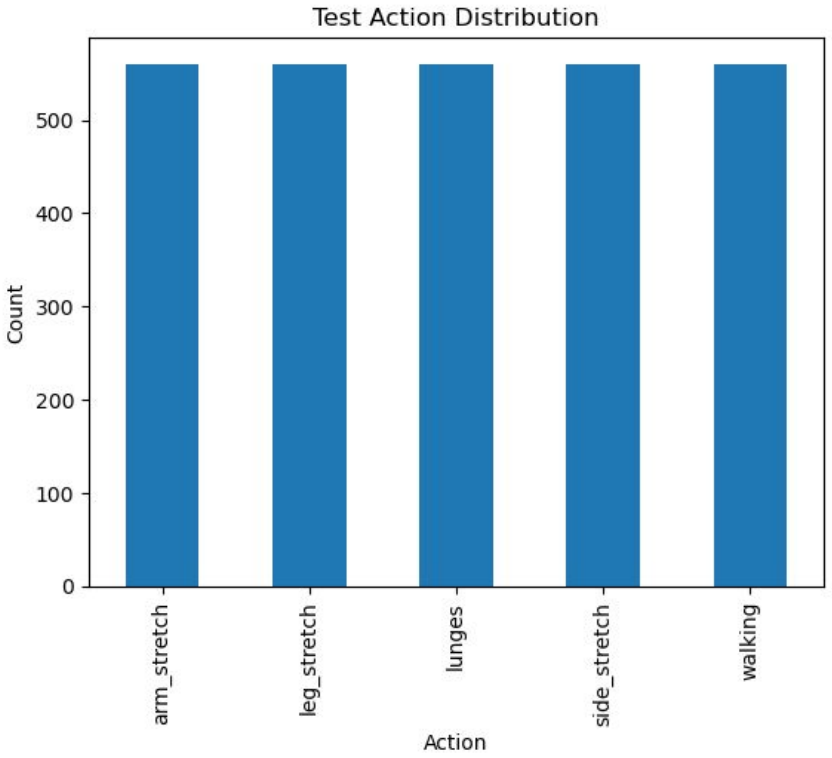
- Contains 34594 human poses captured from 1 person doing 5 actions at 10 different camera angle
- Utilises digital camera with global shutter to record videos to label 2D and 3D poses with Mediapipe
- Captures actions of subjects while moving to add variability in movements



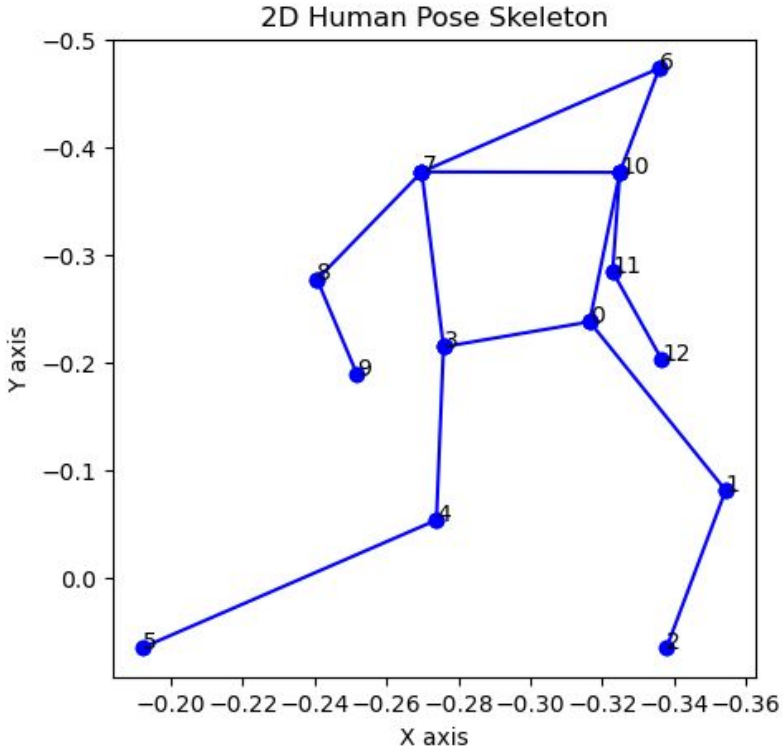


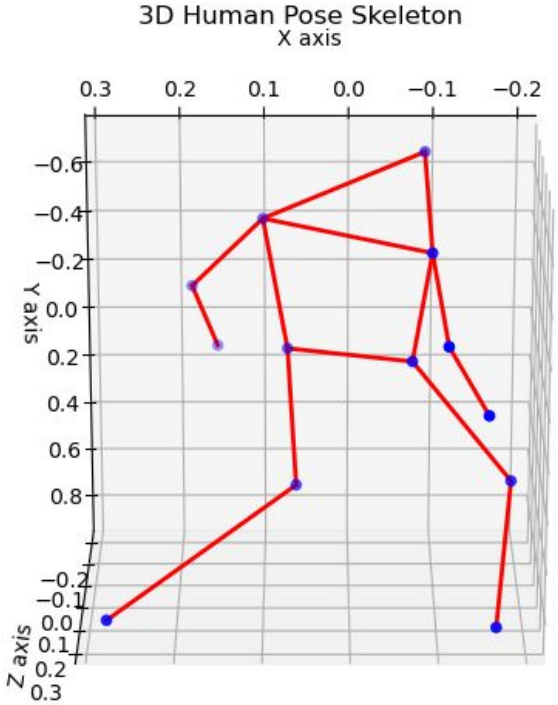


Dataset: Custom Dataset Distribution

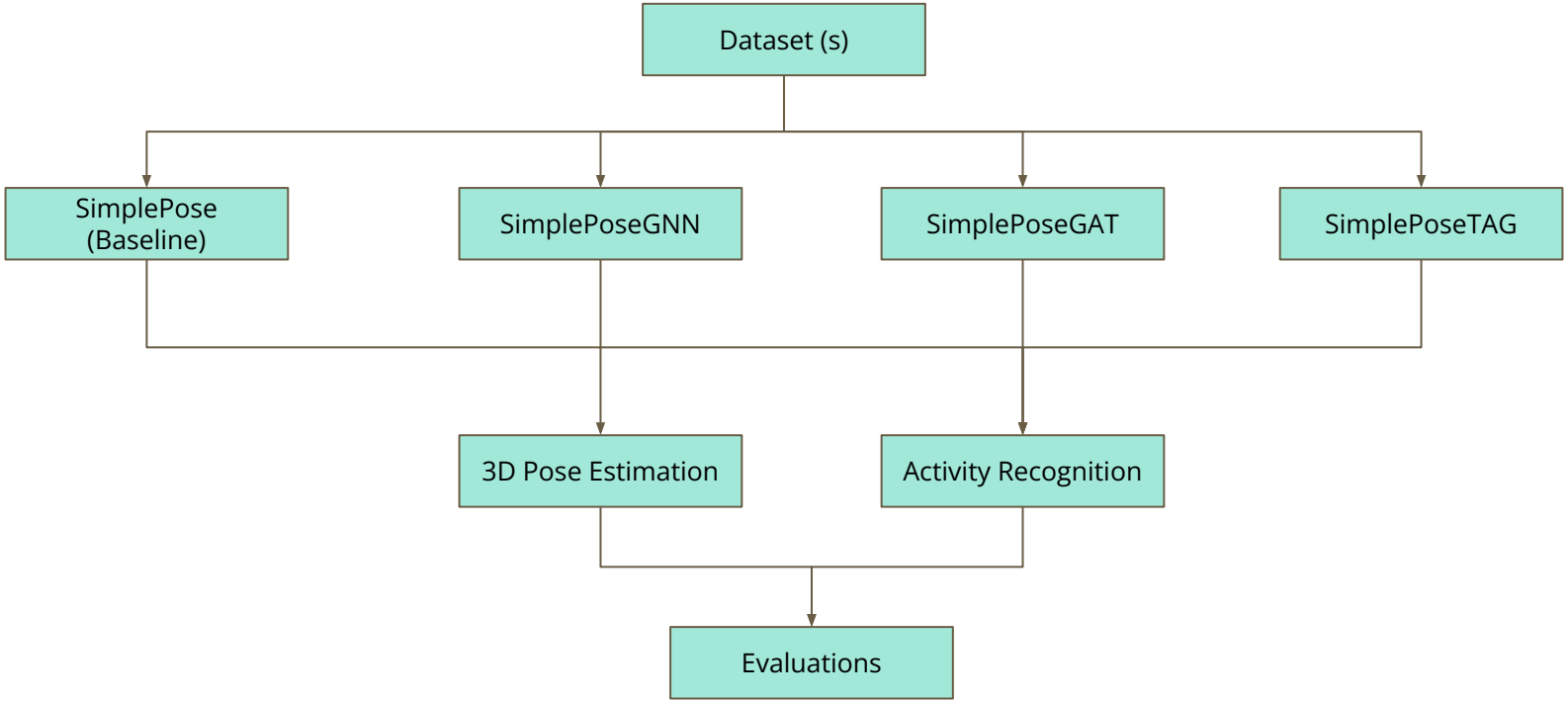


Dataset Skeleton (Human 3.6M & Custom dataset)





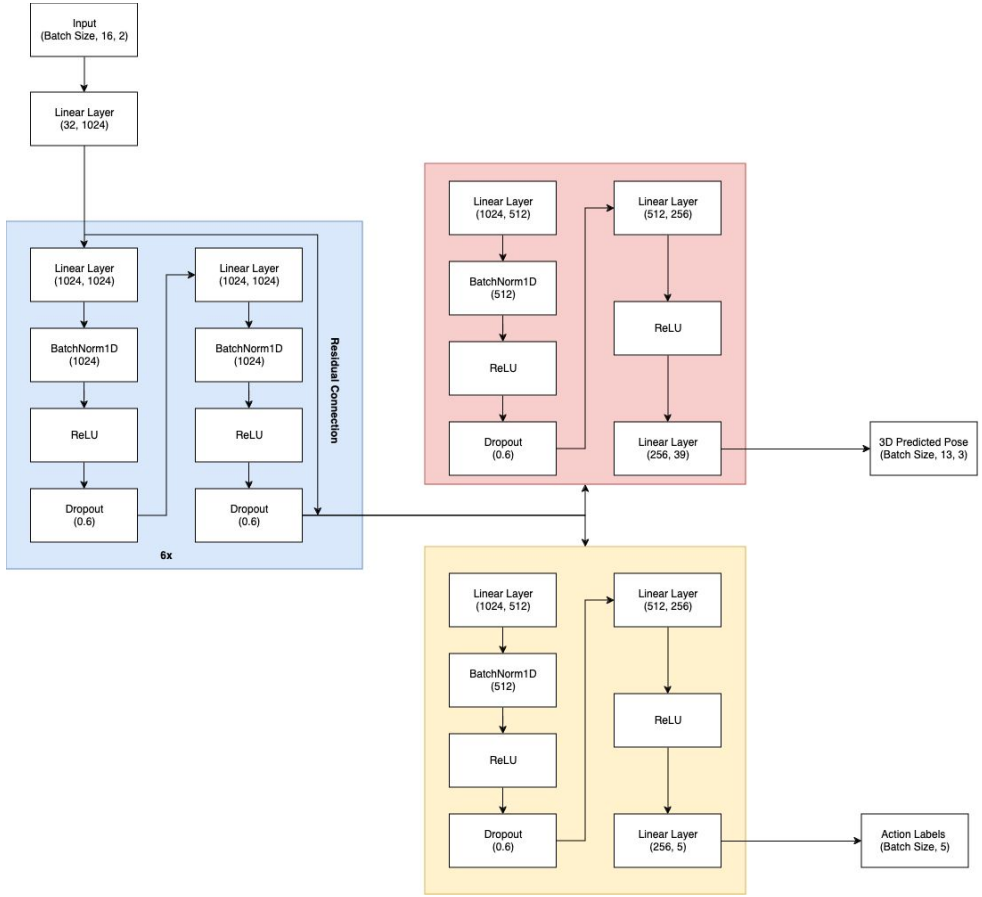
Experiment Setup



Neural Network Implementation (Baseline)

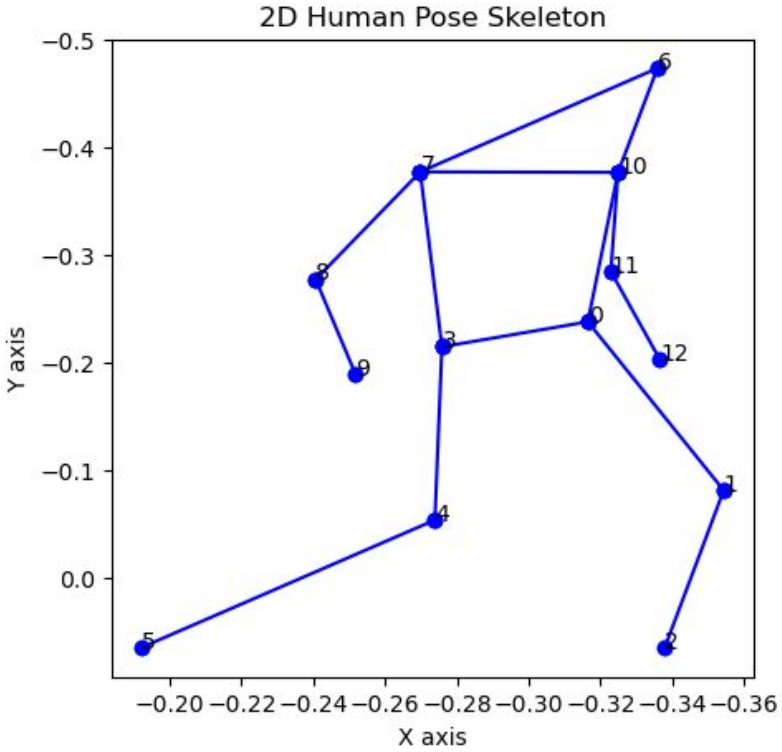
Model: SimplePose

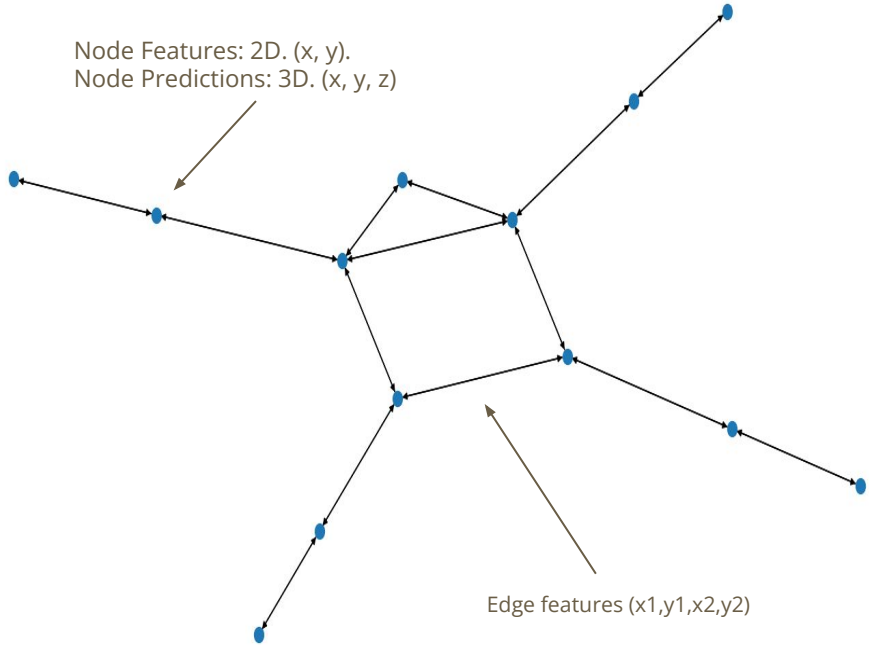
- 13,973,036 Trainable Parameters
- This is our best model iteration for SimplePose where it matches the performance of SimplePoseTAG in H36m dataset.



Graph Neural Network Implementations

DGL Graph Preprocessing for all Graph Models

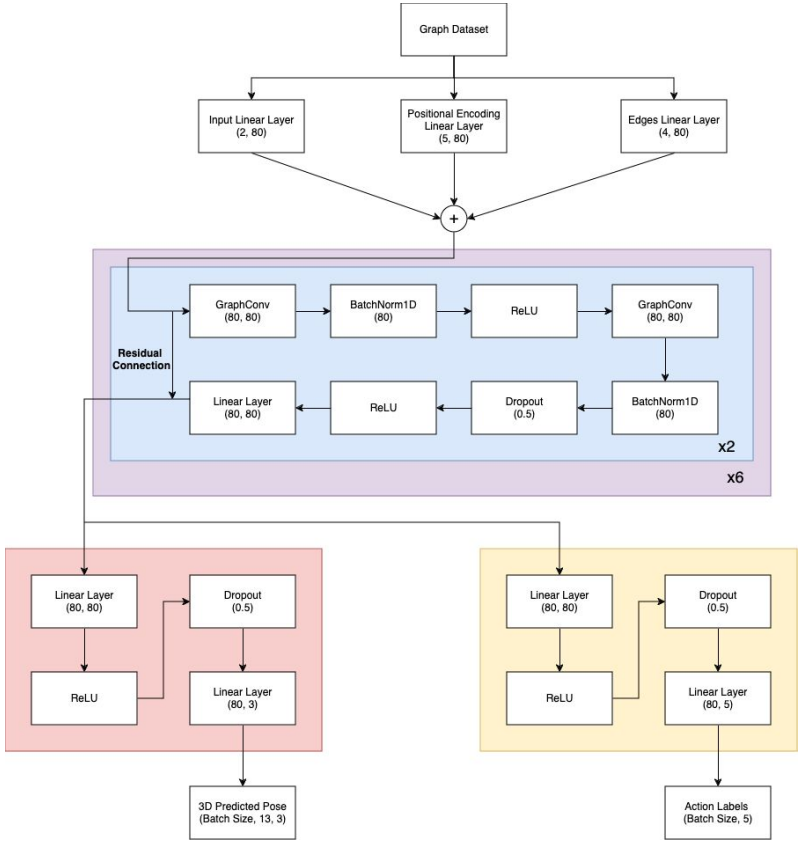




Pose Label: Standing

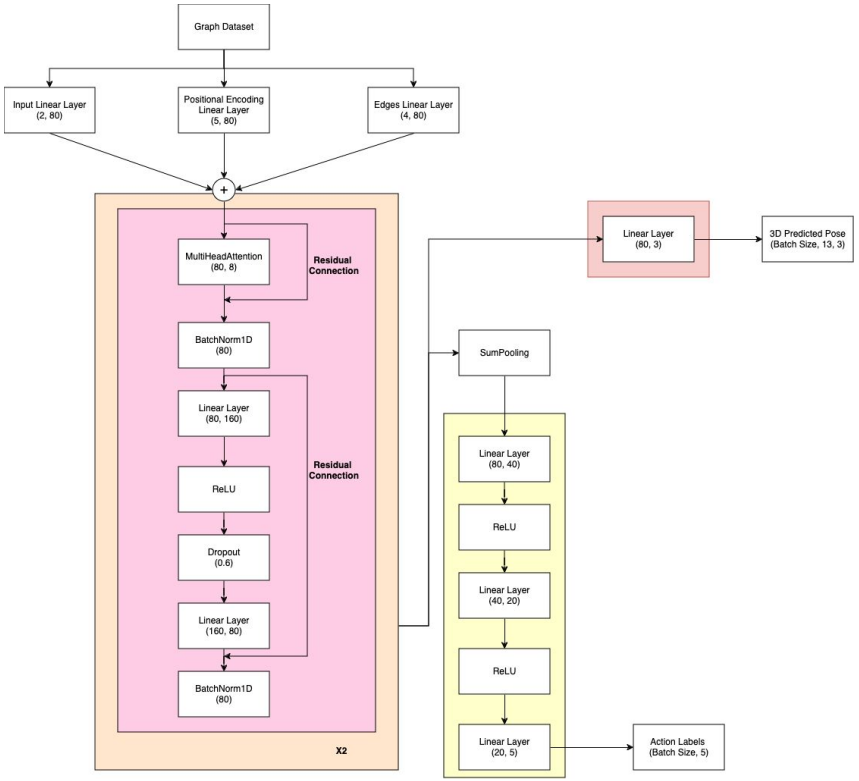
Model: SimplePoseGNN

- 251,848 Trainable Parameters
- Calculate the graphs' Laplacian Positional Encoding with k = 5



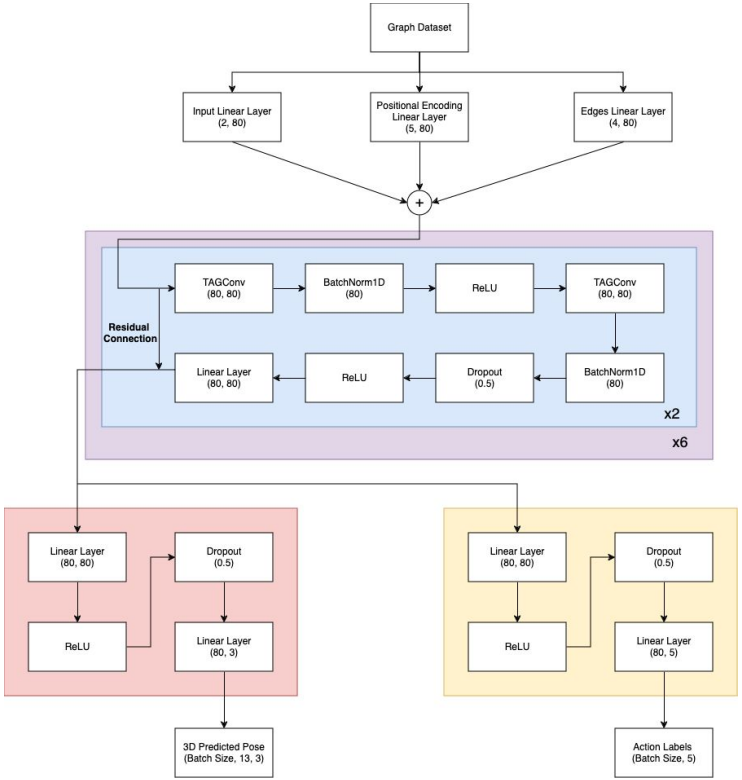
Model: SimplePoseGAT

- 110,888 Trainable Parameters
- Calculate the graphs' Laplacian Positional Encoding with k = 20



Model: SimplePoseTAG

- 1,019,848 Trainable Parameters
- Calculate the graphs' Laplacian Positional Encoding with k = 5
- The multi hop parameter is also set to 5



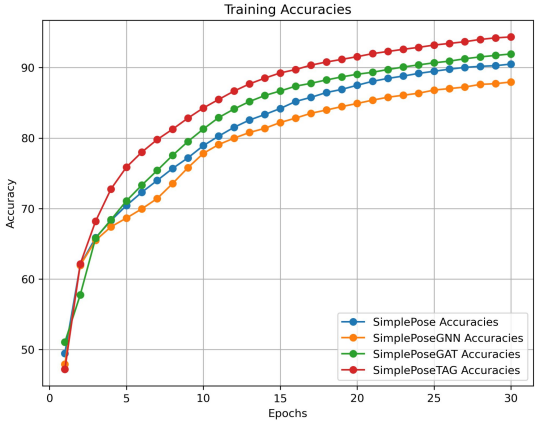
Training setup

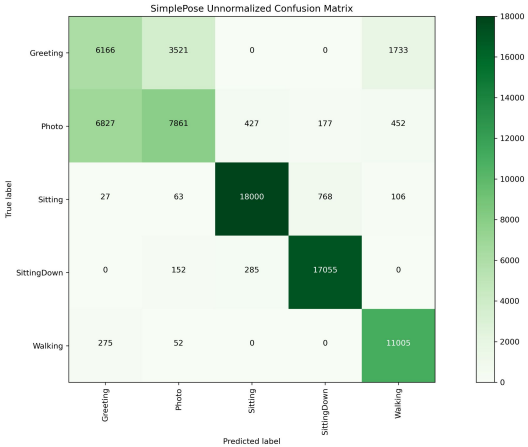
- Our experiments evaluate the 4 models mentioned before on both the Human3.6M dataset as well as the custom dataset.
- Therefore we train all the models using the same set of hyperparameters
- These multipliers ensure that the absolute loss value of both of these loss components are of the same magnitude thereby ensuring that the model trains on both

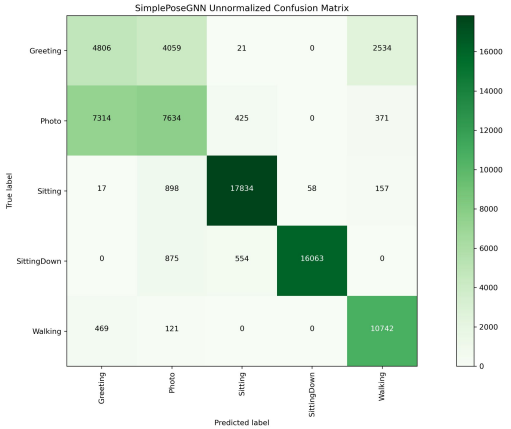
Learning Rate	Batch Size	Epochs	Pose loss multiplier	Action loss multiplier
3e-5	256	30	100	1

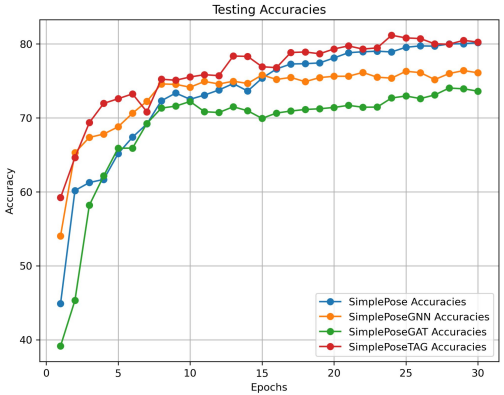
Using the Human 3.6M Dataset

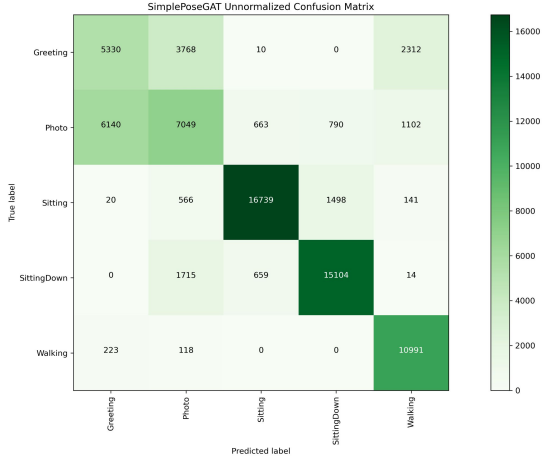
Accuracies across different models (Human 3.6M dataset)

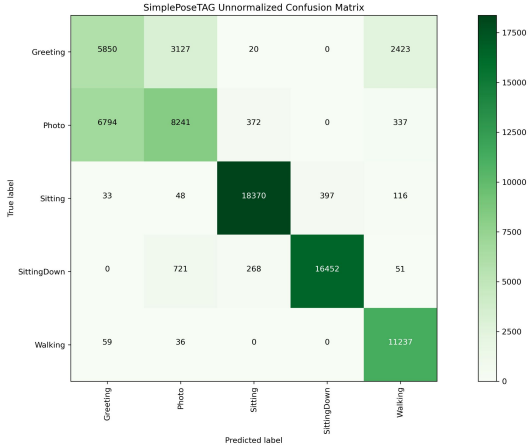




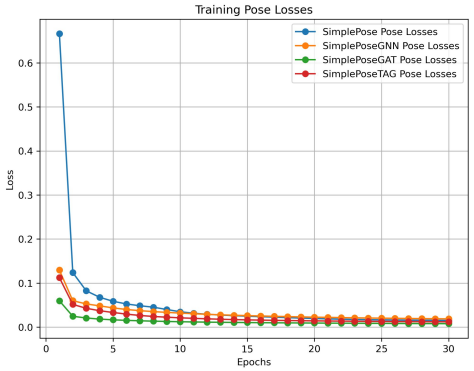


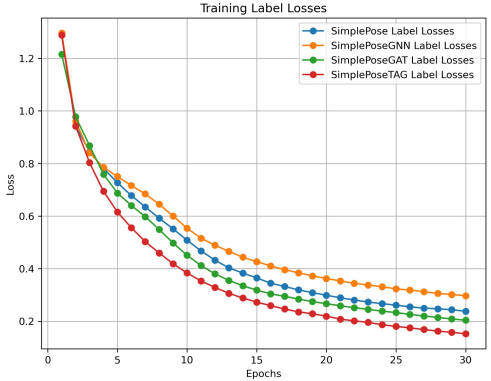


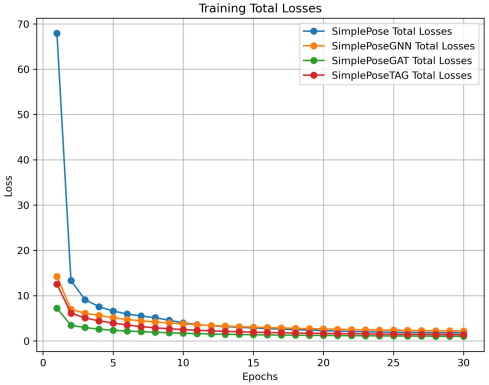


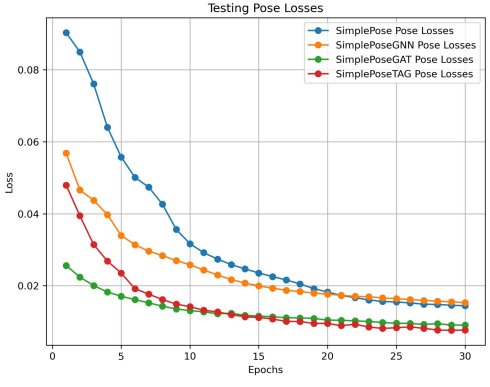


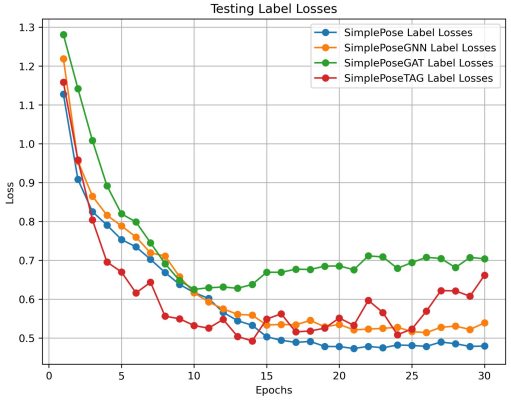
Losses across different models (Human 3.6M dataset)

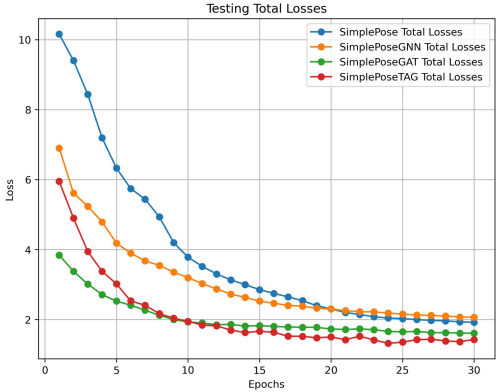






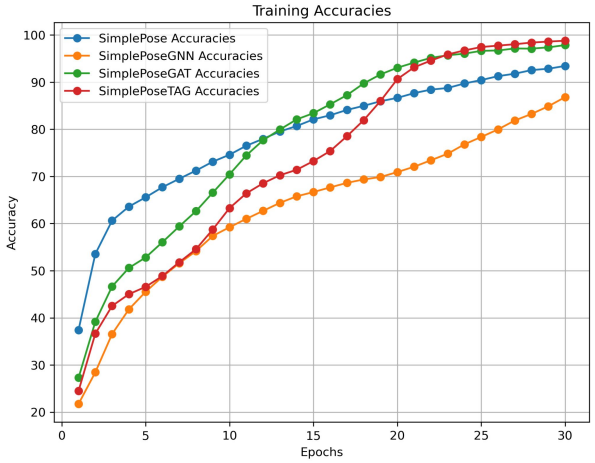


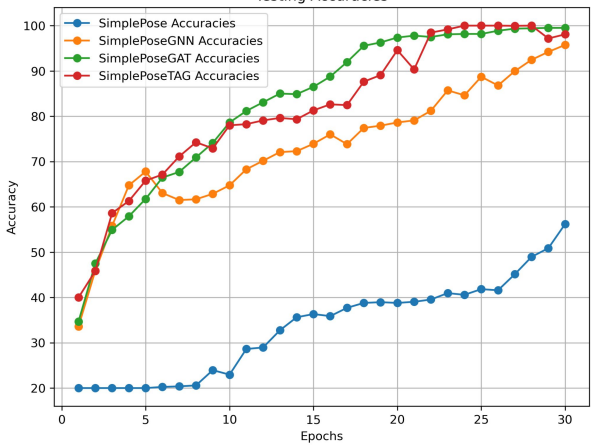


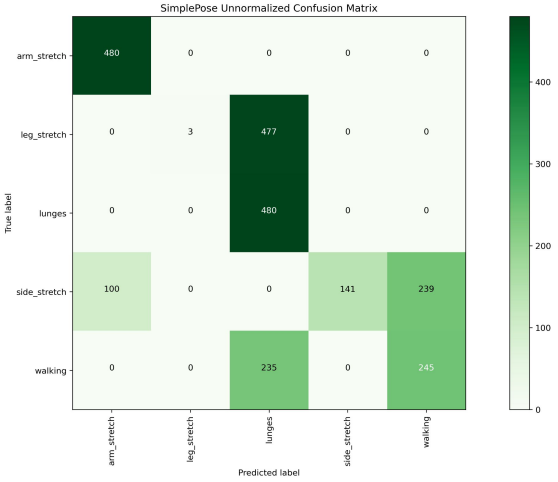


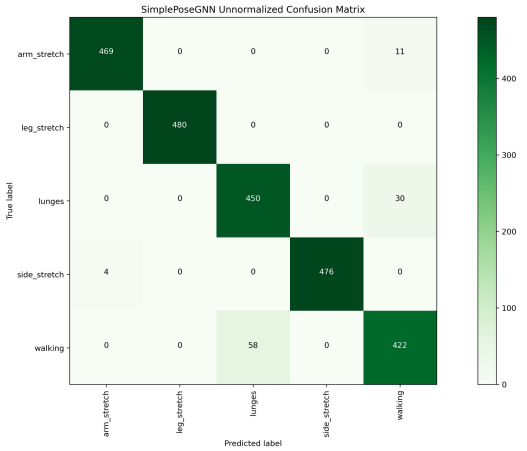
Using the Custom Dataset

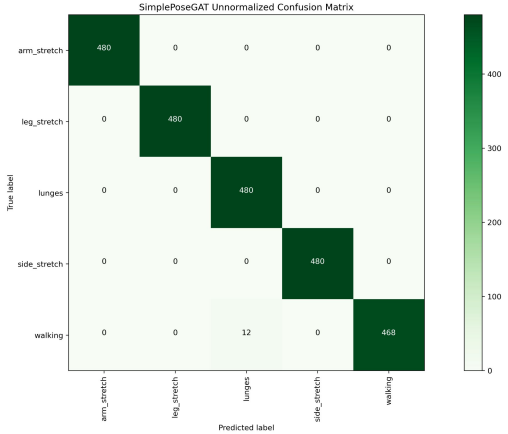
Accuracies across different models (Custom dataset)

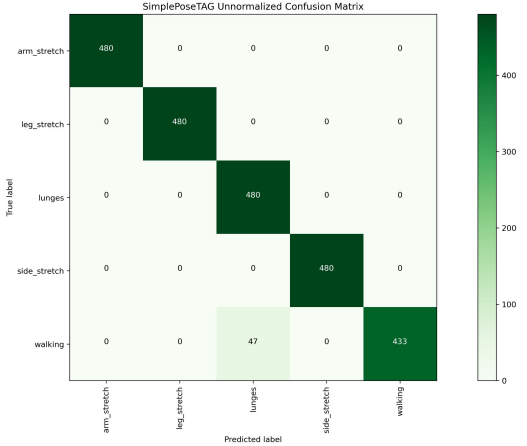




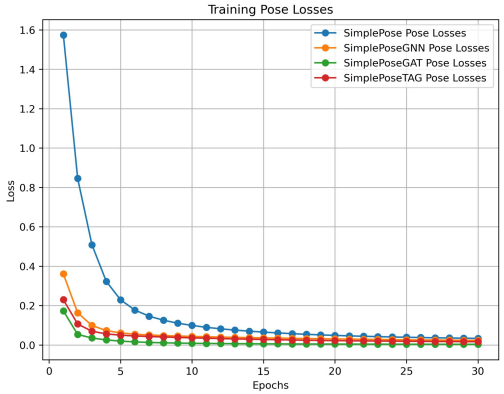


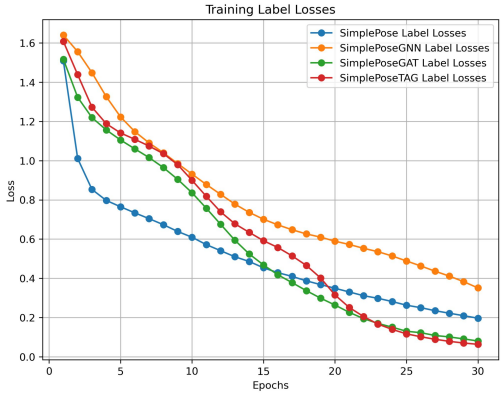


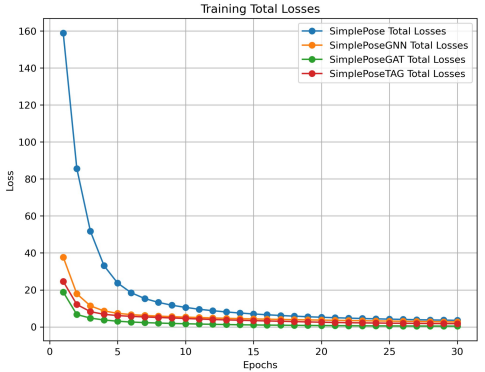


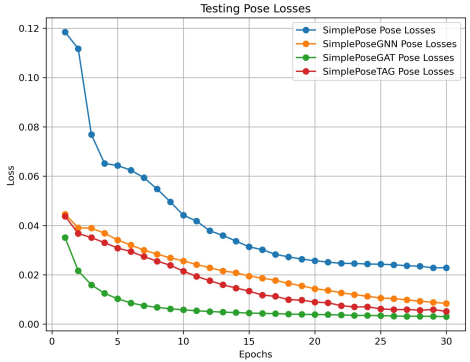


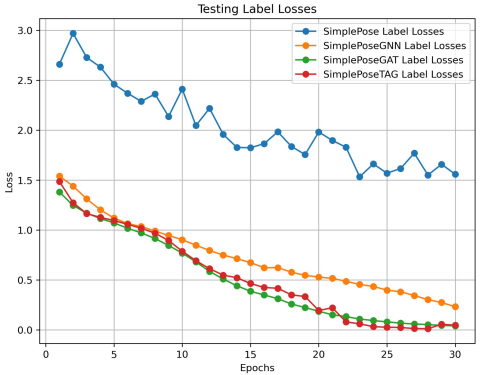
Losses across different models (Custom dataset)

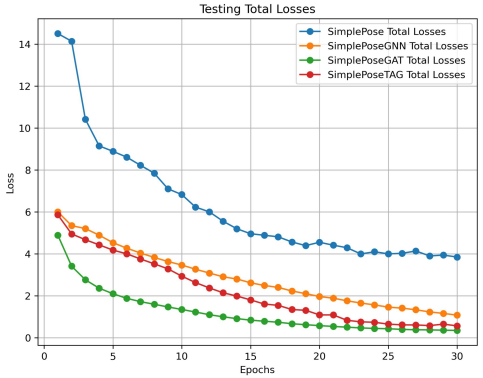












Overall Performance Analysis

- Judging based on the Human 3.6M dataset and our own custom dataset, we believe it is fair to say that some poses favours different models based on the results.

- Throughout our results, SimplePoseTAG performed really well across these dataset as compared with the other methods. This indicates that the graph topology plays a huge role in Pose Estimation and Activity Recognition using Graphs.

- Nevertheless, we proved that when data can be modeled naturally as a graph, Graph Neural Network models can match or even outperform basic Neural Network approaches.

Limitations

- We believe that SimplePoseGAT can perform well if not better than SimplePoseTAG if we were to train it for longer or have better computational resources. This also tells us that given limited computational resources, training SimplePoseTAG will be a wise choice.

- Another limitation we believe is our own custom dataset, we believe that if we are able to collect more data, the results could have been different.

Thank You