Project Proposal: 3D Pose Estimation with Graph Neural Network (GNNs) for Activity Recognition

Team Members (Group 16): Chong Jun Rong Brian (A0290882U), Parashara Ramesh (A0285647M), Ng Wei Jie Brandon (A0184893L)

Project Motivation

Accurately predicting 3D human poses from 2D keypoints is a critical task for many applications such as motion capture and activity recognition. Traditional methods that use direct regression or lifting techniques often struggle to fully capture the complex spatial relationships between body joints. By treating the 2D pose keypoints as graphs, we can leverage the underlying connectivity between joints to improve the 3D pose estimation. Additionally, recognizing and classifying human activities from these poses is an essential task in fields like surveillance and healthcare. Therefore, this project seeks to explore how GNNs can enhance 3D pose estimation and activity recognition.

Project Description

The primary objective of this project is to predict 3D human poses from 2D pose keypoints accurately using GNNs. First, we will develop a basic Neural Network (NN) and Convolutional Neural Network (CNN) for 3D pose estimation. Second, we will reimplement the SemGCN model, which treats the body joints of a 2D pose as nodes in a graph, with edges representing the connectivity between them. Third, we will design an improved version of the SemGCN model by exploring different GNN architectures and modifications to enhance its performance. The secondary objective is to classify human activities based on 2D pose keypoints. We will use custom datasets to validate this task, allowing us to assess the generalization capabilities of GNN-based models for activity recognition.

Proposed Solution

3D Pose Estimation: Firstly, we will build a simple baseline model using deep learning techniques such as NN and CNN. This will be compared against the SemGCN model, which utilizes GNNs to process the 2D pose as a graph and predict the corresponding 3D pose. Finally, we implement a custom GNN model by taking inspiration from the SemGCN model and then compare with the other methods as well.

Activity Recognition: We will collect a custom dataset of 2D poses for a fixed set of activities. GNN-based models will be employed for activity classification, leveraging the graph structure of the poses. Accuracy will be the primary metric to evaluate activity recognition performance.

Dataset: We intend to use the Human3.6M Dataset to train and evaluate the 3D pose estimation and activity recognition tasks. We also intend to build a custom dataset of 2D pose keypoints from a predefined set of activities for the activity recognition task.

Metrics: Average Mean Per Joint Position Error (*MPJPE*) will be used to measure the performance of 3D pose estimation tasks. Accuracy will be used to measure the performance for activity recognition tasks.

Project Milestones

Week 8 - **Literature Review**: Conduct a review of relevant research on 3D pose estimation and GNNs. Set up the development environment and familiarize the team with the Human3.6M dataset. Begin preliminary data exploration and preprocessing.

Week 9 - Baseline And Custom Model Implementation: Start with implementing baselines using simple NN and CNNs. Implement the SemGCN model for 3D pose estimation. Implement a custom GNN model. Evaluate all implementations using the Human3.6M dataset, focusing on MPJPE.

Week 10 - Custom Dataset Collection for Activity Recognition: Design and implement a process to collect 2D pose data from various activities. Annotate the dataset for use in the activity recognition task.

Week 11 - Activity Recognition Implementation: Implement GNN-based models for activity classification using the custom dataset. Fine-tune models and optimize hyperparameters.

Week 12 - Model Evaluation and Comparisons: Compare the performance of the SemGCN and traditional models for 3D pose estimation. Compare GNN-based models with other methods for activity recognition, focusing on accuracy.

Week 13 - Final Report and Presentation: Compile the results and insights from both tasks. Prepare a comprehensive report and presentation on the findings, including future improvements and applications.

References

Zhao, Long and Peng, Xi and Tian, Yu and Kapadia, Mubbasir and Metaxas, Dimitris N., Semantic Graph Convolutional Networks for 3D Human Pose Regression, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 3425--3435, 2019

Julieta Martinez and Rayat Hossain and Javier Romero and James J. Little, A simple yet effective baseline for 3d human pose estimation, https://arxiv.org/abs/1705.03098

Eric Hedlin and Helge Rhodin and Kwang Moo Yi, A Simple Method to Boost Human Pose Estimation Accuracy by Correcting the Joint Regressor for the Human3.6m Dataset, https://arxiv.org/abs/2205.00076

Team Contract

By signing below, each team member agrees to the tasks assigned and commits to contributing equally to the success of this project.

Team Member Name	Signature	Date
Chong Jun Rong Brian	Brian	12 October 2024
Parashara Ramesh	Parashara	12 October 2024
Ng Wei Jie Brandon	Brandon	12 October 2024

Task Allocation

Group Member	**Tasks**
Chong Jun Rong Brian	**III. Model Development – NN and CNN Model:** ● Implement the baseline linear and CNN model for pose estimation. ● Optimize the model architecture and hyperparameters for improved performance.
Parashara Ramesh	**IV. Model Development – SemGCN Reimplementation:** ● Reimplement the SemGCN model for 3D pose estimation. ● Implement performance evaluation metrics post estimation.
Ng Wei Jie Brandon	**V. Model Development – Improved SemGCN Model:** ● Implement the improved model with SOTA GNN architectures. ● Implement GNN-based models for pose estimation
All	**I. Data Preprocessing**: ● Preprocess Human3.6M dataset for activity recognition ● Curate and preprocess custom dataset for activity recognition **II. Evaluation Metrics** ● Implementing performance evaluation metrics such as MPJPE for 3D pose estimation and accuracy for activity recognition. **VI. Building the Activity Recognition** ● Modifying our models to perform activity recognition.