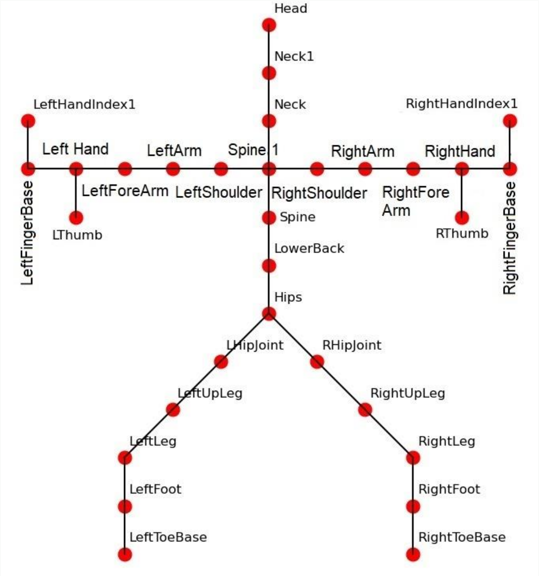
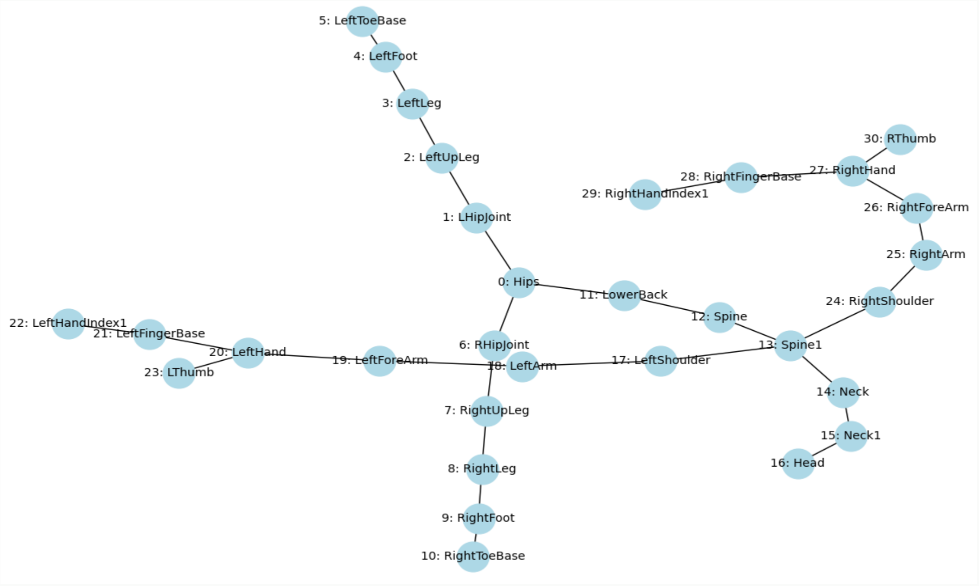
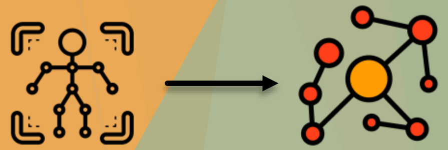
Action and Emotion Recognition by Graph Convolutional Network (GCN)





Instructor - Cesare Alippi
TAs - Tommaso Marzi **and** Gabriele Dominici
Presenter – S. M. Hossein Mousavi

May 2024



Outline

- Introduction
- Action and Emotion Recognition
- Emotion Recognition Modalities
- Body Motion Modality
- Xia Dataset
- Graph Convolutional Network (GCN)
- The Contribution
- Results
- *References*

Introduction

- **Importance [1]**
  - GDL methods offer a deeper understanding of the **data structure** by capturing **subtle details** of the input.
  - Also, it can handle **irregular data structures** like **body skeletons** in which each joint (or node in the graph) can have a varying number of connections to other joints.
  - It normally leads to higher accuracy by **capturing the complex relationship** of the data (extracting relevant features of edges and joints).

- **Challenges**
  - **Feature extraction** is always a problem in traditional methods
  - Even algorithms such as CNN, which extract features, can’t handle different data structures and handle everything **grid-like**
  - Older methods **can’t handle irregular data structures** of the human body perfectly.

Introduction

- **Solution**
  - To extract just **meaning full features** automatically exactly based on the data structure as a graph.
  - To **avoid unnecessary computation** due to having more understanding of the data structure.
  - To **handle irregular shapes** of the body more effectively, by **mapping skeletal data into graphs**.
- **Drawback**
  - The data structure for input must be extracted and defined.

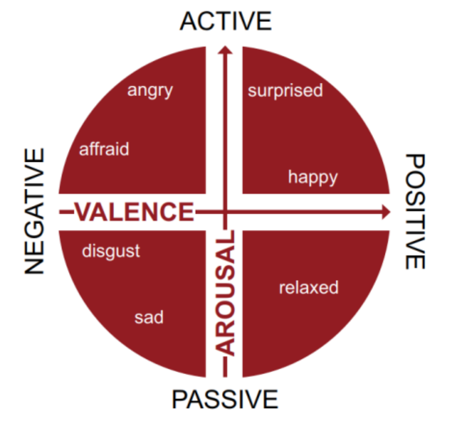
Action and Emotion Recognition

- **Action Recognition (AR):**
  - ➢ It involves using machine learning and pattern recognition to classify human actions such as walking, running, jumping, and more.
- **Emotion Recognition (ER):**
  - ➢ The same techniques are used to determine subjects’ emotional states, such as neutral, anger, joy, and more.
- **In Body Motion**:
  - ➢ Body motions are sequential frames of bodily movement over time, which consist of joints and connected bones/edges.
    - ✓ By investigating **joint angles, positions, rotations**, and the relationships between these joints and edges, actions and emotions could be interpreted [2].

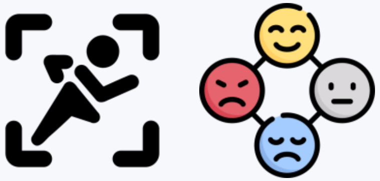
Ekmanian ER Model

1. ***Joy***
2. ***Anger***
3. ***Sadness***
4. ***Fear***
5. ***Disgust***
6. ***Surprise***
7. ***Neutral***

Ekmanian ER Model



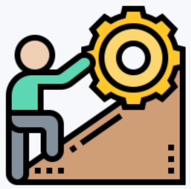
2-D Arousal Valance ER Model [3]



Action and Emotion Recognition

- **Some AR Applications are:**
  - ➢ Security
  - ➢ Sport
  - ➢ Physical Therapy
  - ➢ Entertainment
  - ➢ Smart Home

➢ **Some AR Challenges are:**
  - ➢ Variability in Actions
  - ➢ Real-Time Processing
  - ➢ Data Scarcity
  - ➢ Privacy Issues

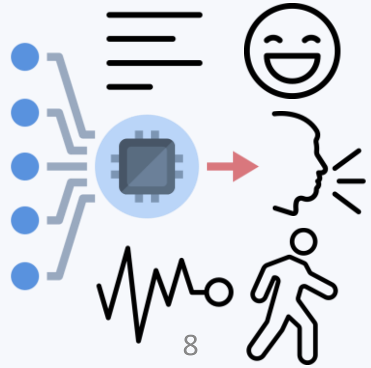



Action and Emotion Recognition

- **Some ER Applications are:**
  - Healthcare
  - Automotive
  - Education
  - Security
  - Marketing
- **Some ER Challenges are:**
  - Variability in Expression
  - Subtlety of Emotional Expressions
  - Data Scarcity
  - Privacy Issues

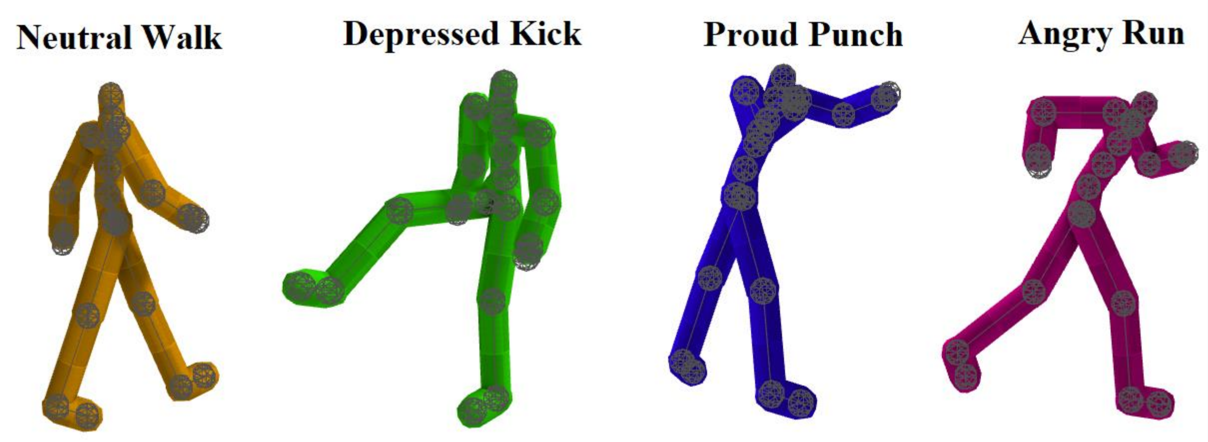
Emotion Recognition (ER) Modalities [4]

- **Facial Expressions:** Image – Change in Facial Wrinkles
- **Vocal Expressions:** Sound Signal – Change in Vocal Tone
- **Physiological Signals for ER:** Vector Signal – Ratio of the Sudden Change
- **Text-Based ER:** String – Change in Writing Style
- **Eye Gaze and Pupil Dilation for ER:** Vector Signal – Change in Eye Direction or Pupil Size
- **Body Motion for ER:** Motion Matrix – Change in Body Posture



Xia Dataset

- **Structure:**
  - 11 Min in 572 Samples and Four Subjects
  - BVH Format and 32 Body Joints
  - Recorded by **Vicon optical** motion capture system
  - Five Actions of **Walking, Running, Jumping, Kicking, Punching**
  - Emotions of **Neutral, Proud, Angry, and Depressed**.
  - Publicly Available by Link



Samples of the Xia dataset generated by BVHView software

Body Motion

- This data type is easily collectible with normal cameras or motion capture technologies; in the latter case, individual identity won't be revealed.
- **Normal color sensors**: **Extracting body joints** and edges by algorithms.
  - ➢ Any smartphone or digital camera.
- **Mocap infrared sensors**: These are the same as color sensors, but **subject's identity is secured**.
  - ➢ Kinect, OptiTrack, and Vicon.
- **Mocap Wearable sensors**: Same as above, and the most precise and the **subject's identity is secured**.
  - ➢ VR headset and handheld controller and Xsens.


Smartphone-Digital Camera



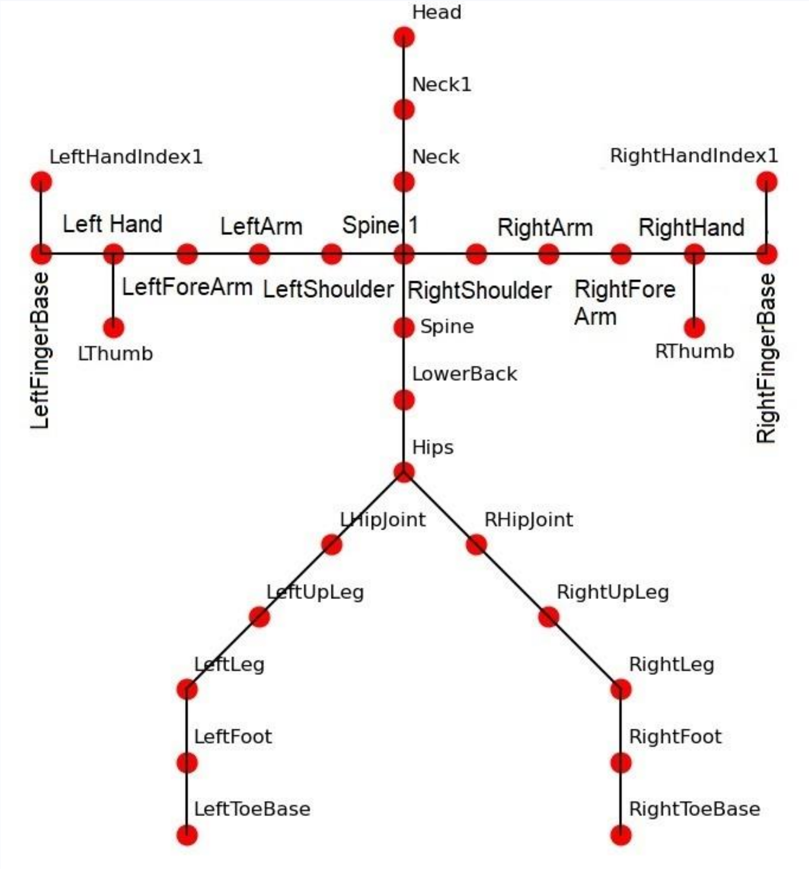
Kinect - OptiTrack - Vicon


VR headset - Xsens

Body Motion

BioVision Hierarchy (BVH) file [5]**:** BVH file format stores motion capture data in the form of **position and rotation of joints over time**.

- It has a **hierarchical skeletal structure** of the body in which each joint has a name.
- A lot of software like Blender and Maya use it as **animation** files.



Skeletal structure of the Xia dataset samples

Graph Convolutional Network (GCN) [6-8]

- GCN **extends the concept of convolution**, traditionally used to process **grid-like data** (such as images), to operate on **graphs**.
- This allows GCNs to effectively **capture the relationships and interactions between nodes** (entities) in a graph.
- Powerful tools for tasks involving networks, such as **social network analysis, molecular structure analysis, and body skeleton**.
- **GCN Input:**
  - ➢ For instance, it is feature matrix X and an adjacency matrix A.
  - ➢ X contains the feature vectors of each node in the graph, with dimensions [N×F], where N is the number of nodes and F is the number of features per node.
  - ➢ A represents the connections between nodes, with dimensions [N×N], where a value of 1 indicates a connection between nodes, and 0 indicates no connection.

Graph Convolutional Network (GCN)

- **GCN Convolution Layers:**
  - ➢ A graph convolution layer **updates the feature vector of each node** by aggregating features from its neighbors and its own features, often using the adjacency matrix A and a set of learnable weights. All followed by ReLU.
- **Hidden Layers**:
  - ➢ A GCN can have multiple graph convolution layers stacked to learn increasingly abstract representations of the graph data. Tasks such as **dropout, normalization, and attention mechanism**.
- **Output Layer**:
  - ➢ The final layer of a GCN transforms the learned representations into the desired output format, which can be node-level predictions (e.g., node classification). It is **fully connected**.

Graph Convolutional Network (GCN)

- **Training Loss Function**:
  - GCNs are trained using **gradient descent** to minimize a loss function, **adjusting the weights** W to improve the model's predictions.
- **Why GCN in Body Motion?:**
  - Applying GCNs to classify body motion resembles the **natural graph structure of human anatomy**, where **joints and limbs can be represented as nodes and edges**, respectively.
  - By understanding these relationships, GCNs can accurately classify various types of body motions, benefiting applications in **sports science, physical therapy, and human-computer interaction**.
  - For emotion recognition, it is superior for **subtle movement representing emotions** over other methods.

The Contribution

- Defining joint and body skeletal structure
- Loading the dataset for Action/Emotion
- Interpolating the number of frames to the maximum to have unified samples
- Converting body skeleton to graph
- Defining GCN model structure
- The recording node features and edge list
- T-SNE plot of Nodes
- Splitting dataset to 70 % train and 30% test
- Training the model
- Testing the model
- Piloting results (acc/loss plot, classification report, and confusion matrix)

The Contribution

Defining joint and body skeletal structure

Joint_Names (31) = [
"Hips", "LHipJoint", "LeftUpLeg", "LeftLeg", "LeftFoot", "LeftToeBase", "RHipJoint", "RightUpLeg", "RightLeg", "RightFoot", "RightToeBase", "LowerBack", "Spine", "Spine1", "Neck", "Neck1", "Head", "LeftShoulder", "LeftArm", "LeftForeArm", "LeftHand", "LeftFingerBase", "LeftHandIndex1", "LThumb", "RightShoulder", "RightArm", "RightForeArm", "RightHand", "RightFingerBase", "RightHandIndex1", "RThumb"
]

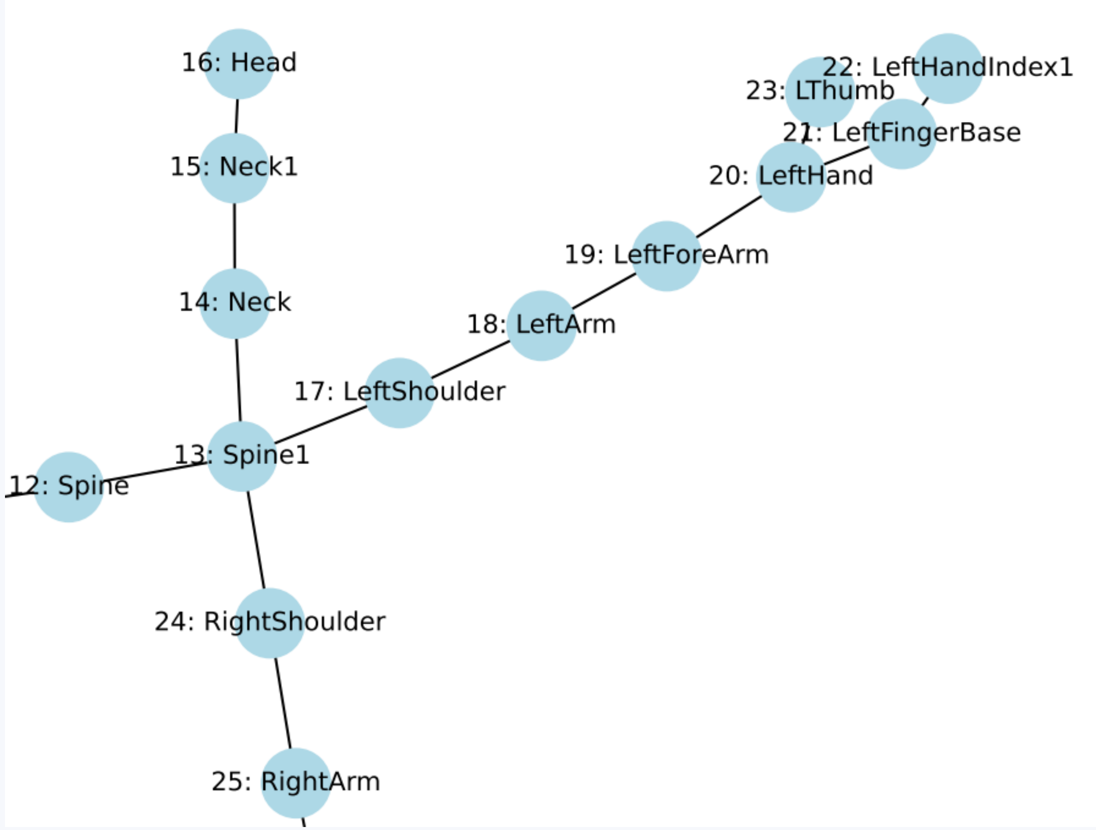
- **Pairs of joints that form the skeleton's connections**

Skeletal_Connections (28) = [
("Hips", "LHipJoint"), ("LHipJoint", "LeftUpLeg"), ("LeftUpLeg", "LeftLeg"),
("Hips", "RHipJoint"), ("RHipJoint", "RightUpLeg"), ("RightUpLeg", "RightLeg"),
("RightLeg", "RightFoot"), ("RightFoot", "RightToeBase"),
("Hips", "LowerBack"), ("LowerBack", "Spine"), ("Spine", "Spine1"),
("Spine1", "Neck"), ("Neck", "Neck1"), ("Neck1", "Head"),
("Spine1", "LeftShoulder"), ("LeftShoulder", "LeftArm"),
("LeftArm", "LeftForeArm"), ("LeftForeArm", "LeftHand"),
("LeftHand", "LeftFingerBase"), ("LeftFingerBase", "LeftHandIndex1"),
("LeftHand", "LThumb"),
("Spine1", "RightShoulder"), ("RightShoulder", "RightArm"),
("RightArm", "RightForeArm"), ("RightForeArm", "RightHand"),
("RightHand", "RightFingerBase"), ("RightFingerBase", "RightHandIndex1"),
("RightHand", "RThumb")
]

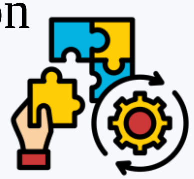
The Contribution

Converting body skeleton to graph

- Graph **nodes** represent **body joints**
- **Edges** represent **skeletal connections**



- **Node** in body motion is the multiplication of the number of body joints by the number of frames
- **Node features** are descriptions of nodes/joints (how many ways a node/joint could be described)
    - For instance, 3000*1000 means there are 3000 nodes which each could be described in 1000 ways
- **Edges** are connections of joints multiplied by the number of frames
    - For instance, 2*3000 means we have 3000 edges, and each row represents two nodes connection
    - If the first column is [0, 1], it indicates there is an edge from node 0 to node 1



The Contribution

- **Input Layer**:
  - ➢ The input consists of node features (x) and edge information (edge_index) from a graph data structure that represents body motion.
- **First Graph Convolutional Layer (conv1):**
  - ➢ **Type**: GCNConv
  - ➢ **Input Features**: The number of node features is dynamically determined based on the input data.
  - ➢ **Output Features**: 16 features. This layer **transforms the input node features** into a 16-dimensional feature space.
  - ➢ **Activation Function**: ReLU (applied after this layer in the forward method). This introduces **non-linearity** to the model, allowing it to **capture complex patterns** in the data.
  - ➢ **ReLU** says if the neuron is important or not to be sent to the next layer.

The Contribution

- **Dropout Layer (from layer 1):**
- After the first GCN layer and its ReLU activation, a dropout operation is applied to **prevent overfitting** by randomly **zeroing some of the features**.
- Dropout rate is 0.5. This rate means that each unit (neuron) in the layer has a **50% chance** of being set to zero during training. Preventing overfitting by **reducing dependency in a single or small group of neurons**.

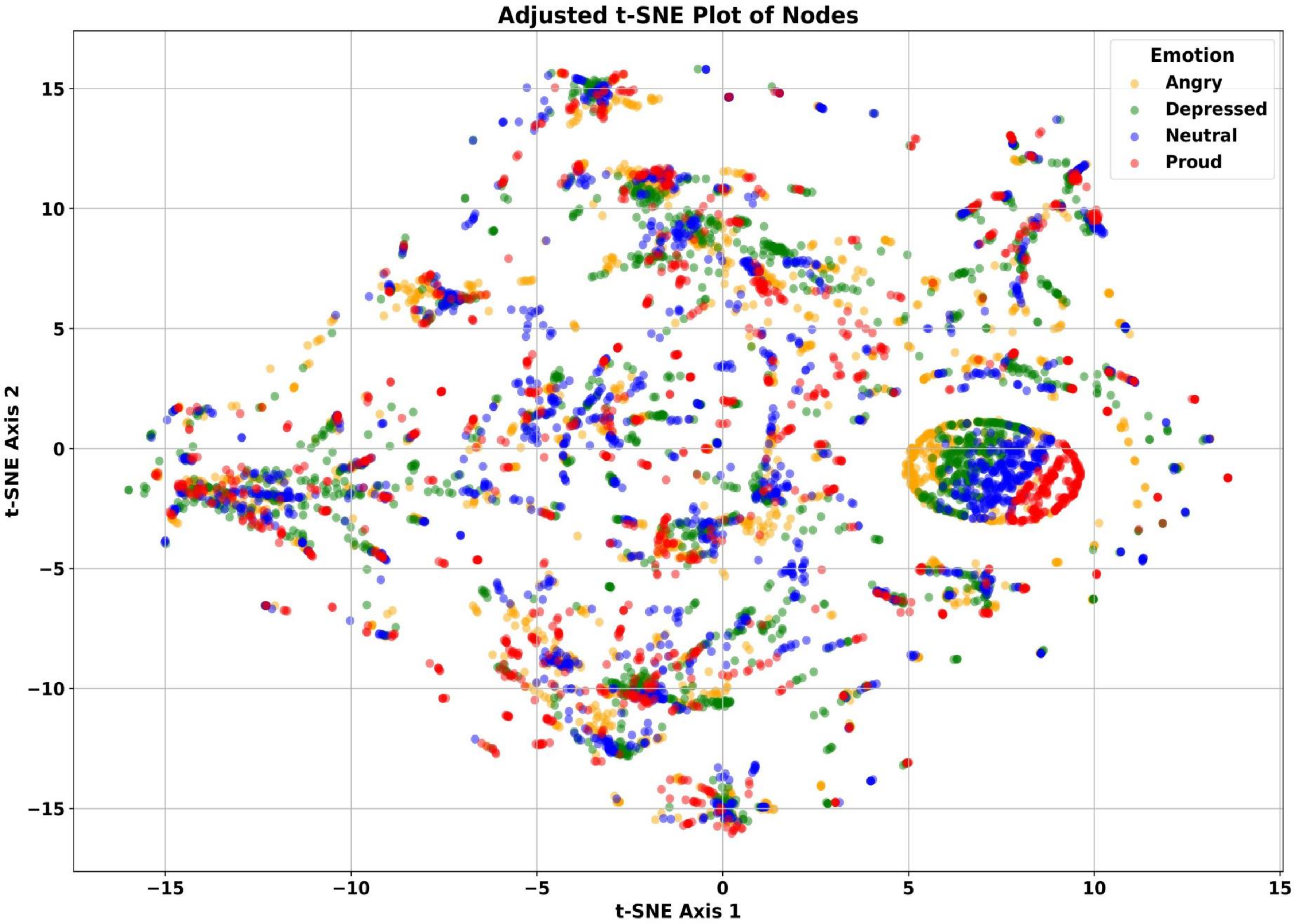
• **Second Graph Convolutional Layer (conv2):**

- **Type**: GCNConv.
- **Input Features**: 16 (the **output features** of the previous GCNConv layer).
- **Output Features**: The number of **target classes** for the motion classification task.
  - This layer aims to prepare the features for classification.

The Contribution

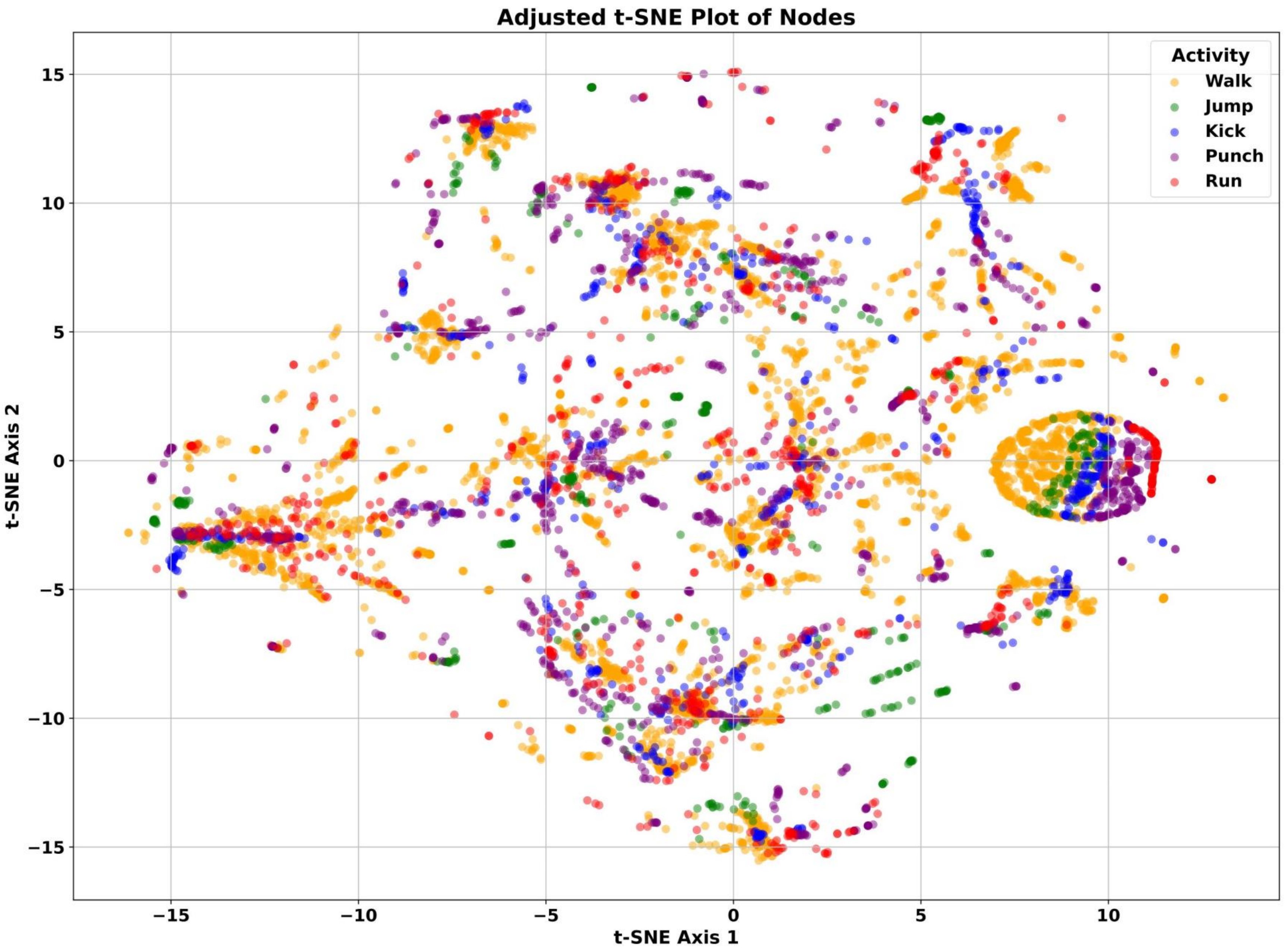
- **Global Mean Pooling:**
  - After the second GCN layer, the **global mean pool** is applied to **aggregate node features into a single graph-level feature vector**.
  - For graph-level predictions, such as **classifying entire graphs rather than individual nodes.**
- **Output Layer:**
  - The output of the model is passed through a **log_softmax function**, which is used for **multi-class classification** tasks.
    - This function provides the **probabilities of each class**, making it easier to determine the class with the highest probability.
- **Training Objective:**
  - A **log_softmax** output for training classification models. It calculates the **loss between the predicted and the ground truth by adjusting the weights**.

The Contribution



Emotion	Sample
Angry	59
Depressed	58
Neutral	58
Proud	47
Sum	**252**
Nodes	**6882**
Edges	**6660**
Features	**2202**

The Contribution



Activity	Sample
Walk	135
Jump	18
Kick	25
Punch	44
Run	30
Sum	**252**
Nodes	**7812**
Edges	**7560**
Features	**2202**

Results

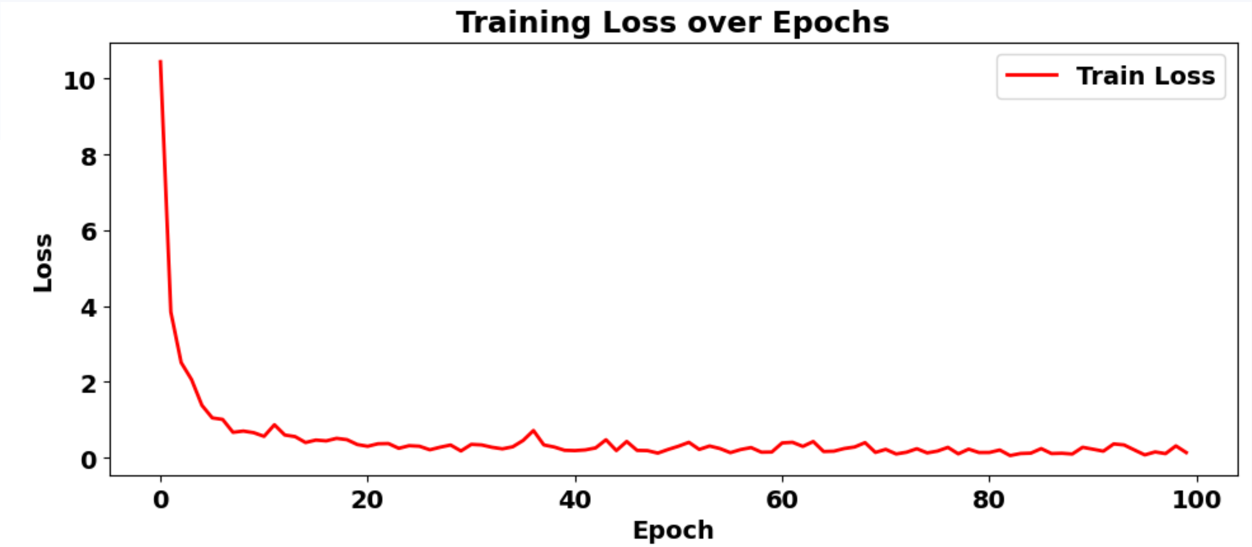
Classification Report:

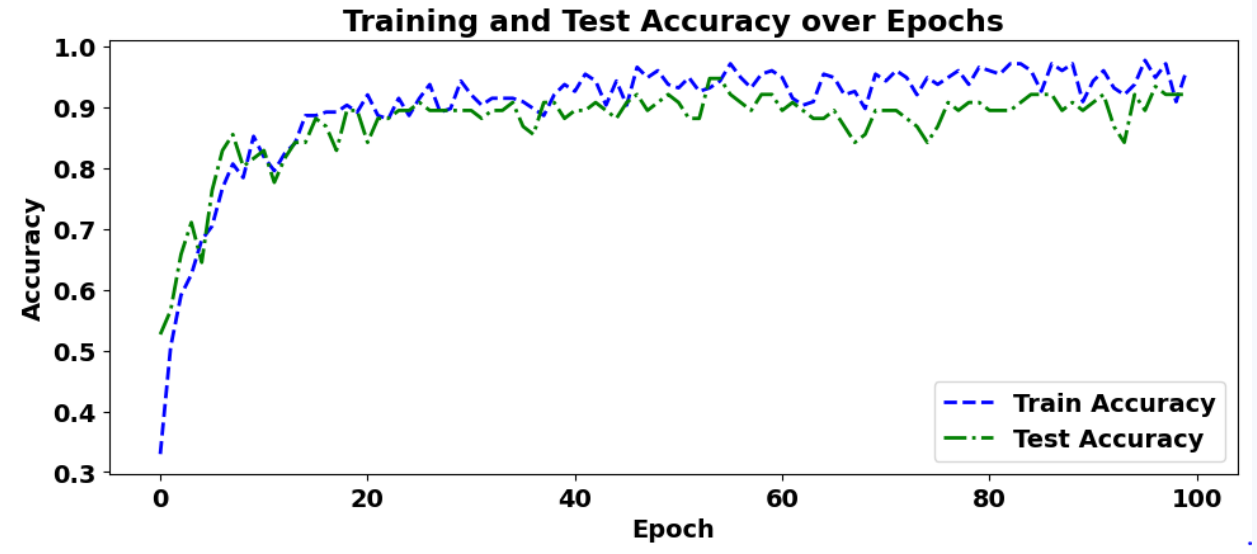
	precision	recall	f1-score	support
Walk	0.97	0.91	0.94	43
Jump	0.71	1.00	0.83	5
Kick	1.00	1.00	1.00	8
Punch	0.85	1.00	0.92	11
Run	0.88	0.78	0.82	9
accuracy			0.92	76
macro avg	0.88	0.94	0.90	76
weighted avg	0.93	0.92	0.92	76

Confusion Matrix:

```
[[39  2  0  1  1]
 [ 0  5  0  0  0]
 [ 0  0  8  0  0]
 [ 0  0  0 11  0]
 [ 1  0  0  1  7]]
```

Train Loss: 0.1319
Train Acc: 0.9602
Test Acc: 0.9211





Results

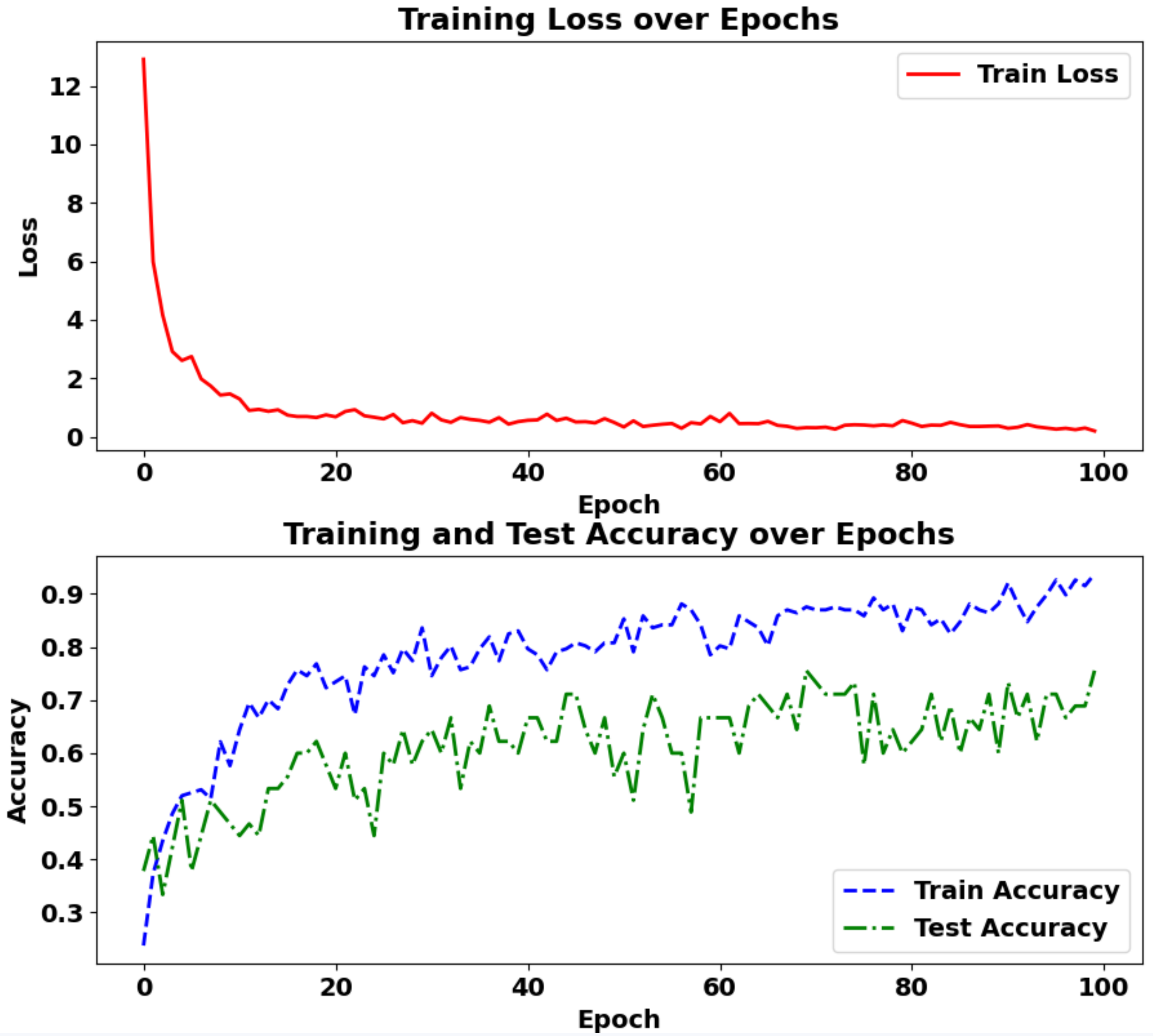
Classification Report:

	precision	recall	f1-score	support
Angry	0.75	0.64	0.69	14
Depressed	0.69	0.92	0.79	12
Neutral	0.75	0.60	0.67	10
Proud	0.89	0.89	0.89	9
accuracy			0.76	45
macro avg	0.77	0.76	0.76	45
weighted avg	0.76	0.76	0.75	45

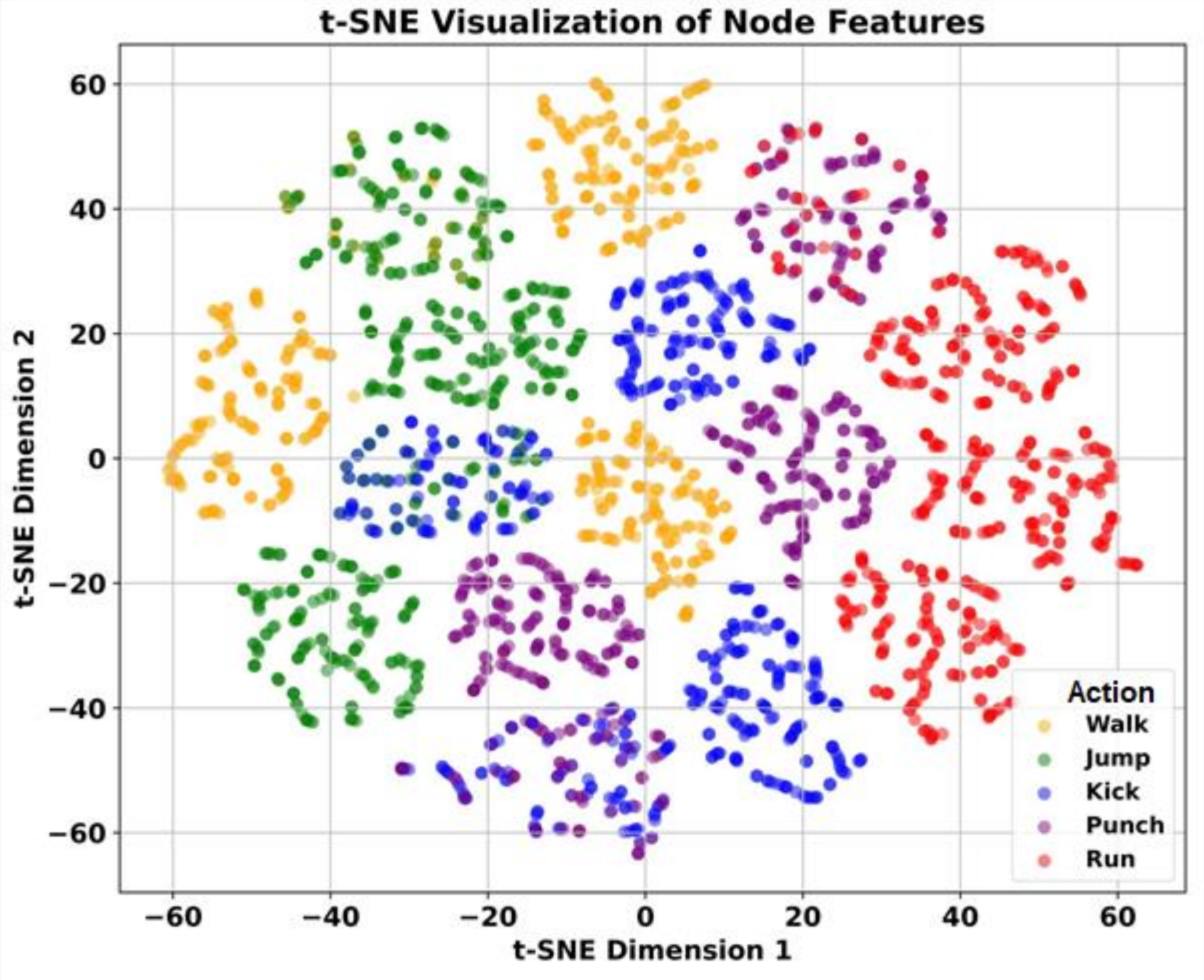
Confusion Matrix:

```
[[ 9  3  1  1]
 [ 0 11  1  0]
 [ 2  2  6  0]
 [ 1  0  0  8]]
```

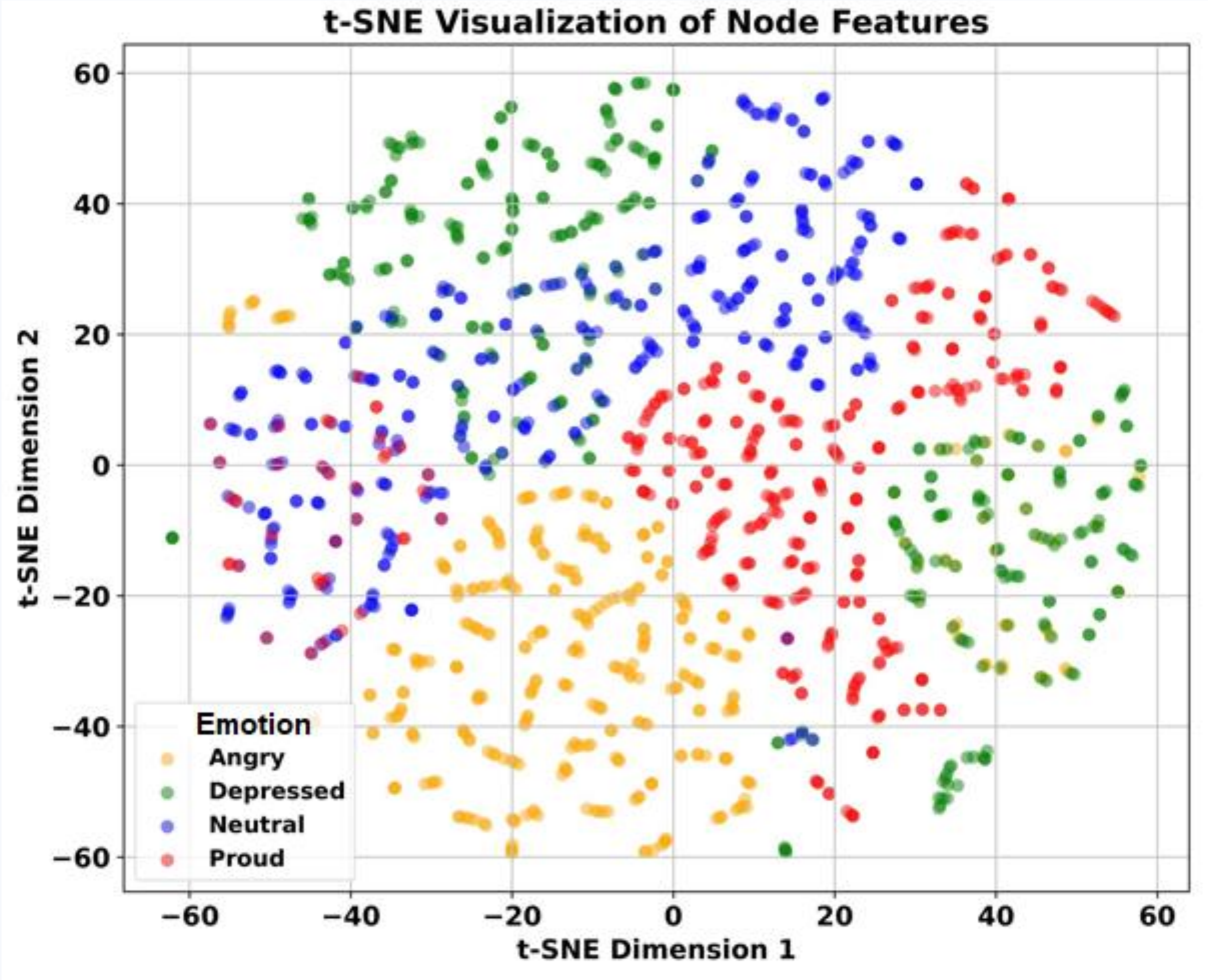
Train Loss: 0.1959
Train Acc: 0.9379
Test Acc: 0.7556



Results

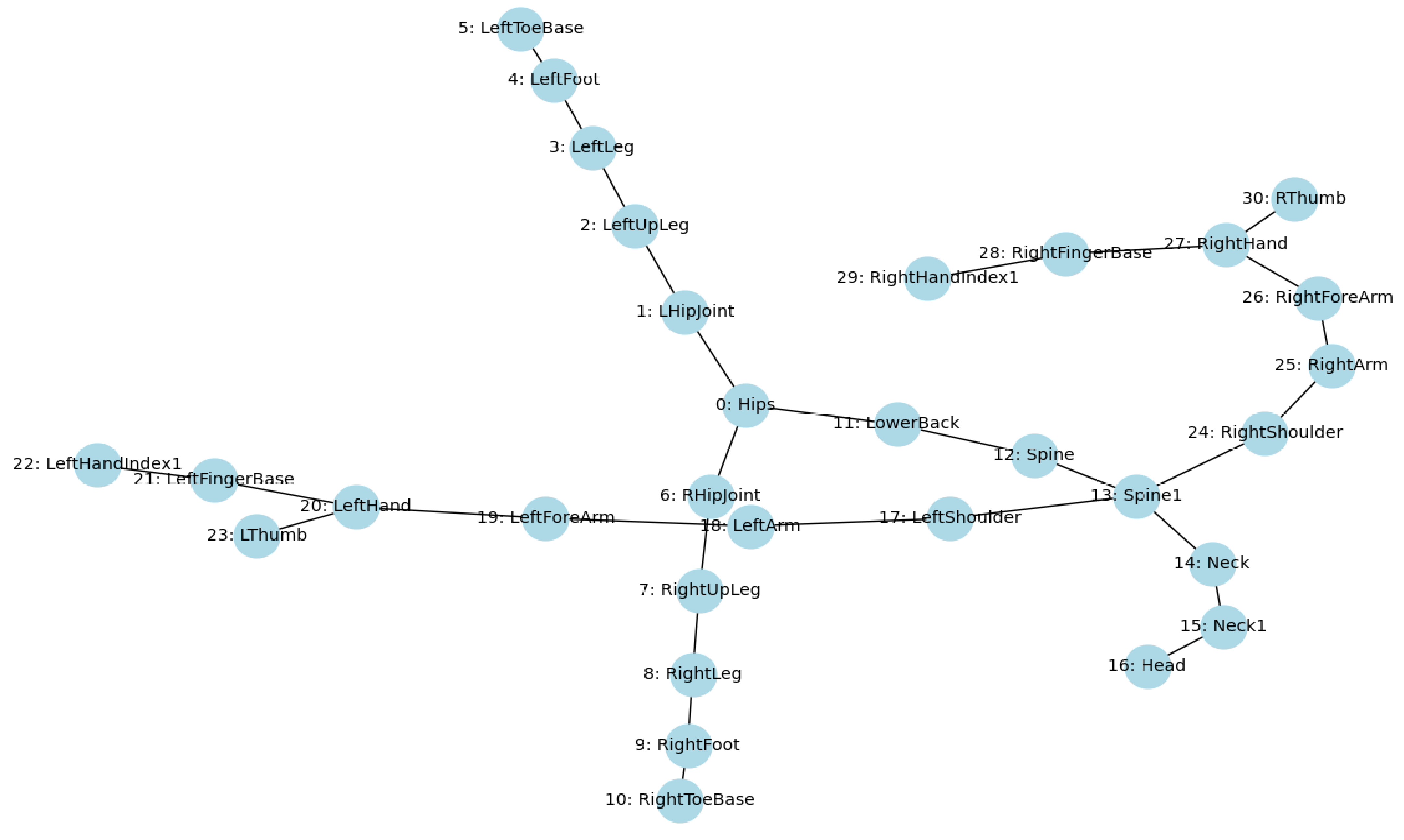


t-SNE Visualization of Node Features
t-SNE Dimension 1
t-SNE Dimension 2
Action
Walk
Jump
Kick
Punch
Run

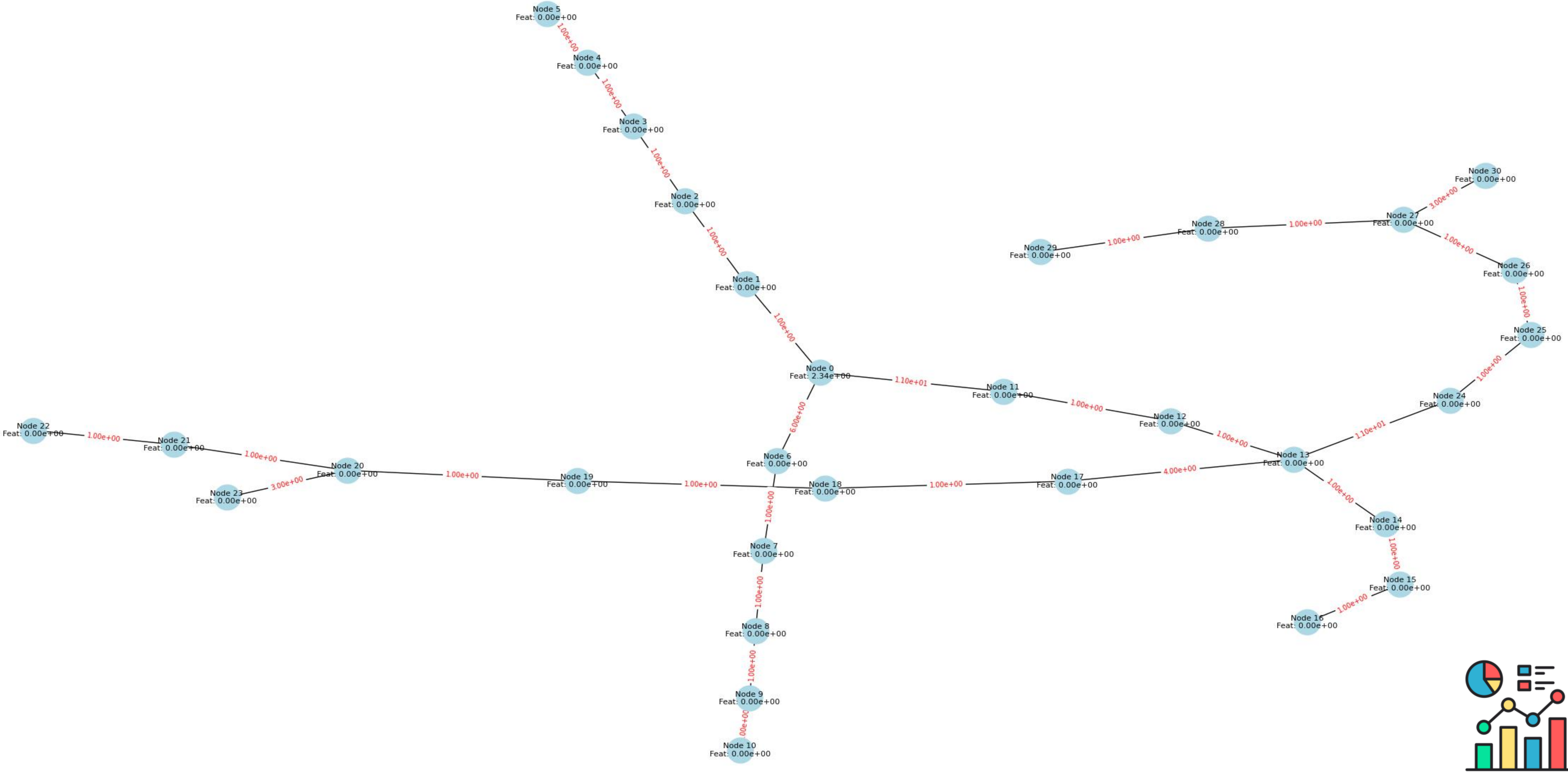


t-SNE Visualization of Node Features
t-SNE Dimension 1
t-SNE Dimension 2
Emotion
Angry
Depressed
Neutral
Proud

Results

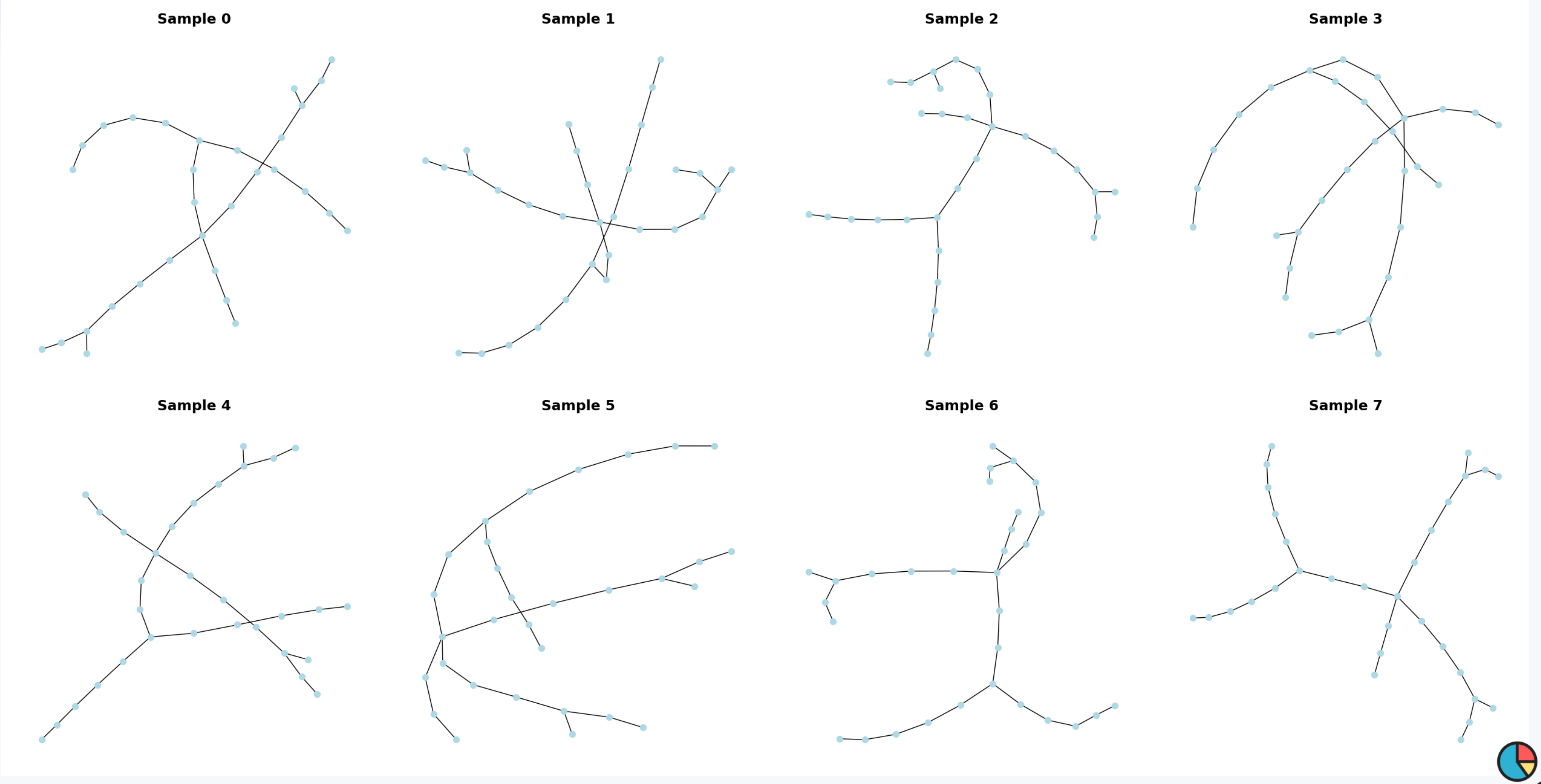

5: LeftToeBase
4: LeftFoot
3: LeftLeg
2: LeftUpLeg
1: LHipJoint
0: Hips
6: RHipJoint
7: RightUpLeg
8: RightLeg
9: RightFoot
10: RightToeBase
11: LowerBack
12: Spine
13: Spine1
14: Neck
15: Neck1
16: Head
17: LeftShoulder
18: LeftArm
19: LeftForeArm
20: LeftHand
21: LeftFingerBase
22: LeftHandIndex1
23: LThumb
24: RightShoulder
25: RightArm
26: RightForeArm
27: RightHand
28: RightFingerBase
29: RightHandIndex1
30: RThumb

Results



Node 5
Feat: 0.00e+00
1.00e+00
Node 4
Feat: 0.00e+00
1.00e+00
Node 3
Feat: 0.00e+00
1.00e+00
Node 2
Feat: 0.00e+00
1.00e+00
Node 1
Feat: 0.00e+00
1.00e+00
Node 30
Feat: 0.00e+00
3.00e+00
Node 27
Feat: 0.00e+00
1.00e+00
Node 28
Feat: 0.00e+00
1.00e+00
Node 29
Feat: 0.00e+00
1.00e+00
Node 26
Feat: 0.00e+00
1.00e+00
Node 25
Feat: 0.00e+00
1.00e+00
Node 0
Feat: 2.34e+00
1.10e+01
Node 11
Feat: 0.00e+00
1.00e+00
Node 24
Feat: 0.00e+00
Node 12
Feat: 0.00e+00
1.00e+00
1.10e+01
6.00e+00
Node 22
Feat: 0.00e+00
1.00e+00
Node 21
Feat: 0.00e+00
1.00e+00
Node 6
Feat: 0.00e+00
Node 13
Feat: 0.00e+00
Node 20
Feat: 0.00e+00
1.00e+00
4.00e+00
Node 19
Feat: 0.00e+00
Node 17
Feat: 0.00e+00
1.00e+00
Node 23
Feat: 0.00e+00
3.00e+00
1.00e+00
Node 18
Feat: 0.00e+00
1.00e+00
1.00e+00
Node 14
Feat: 0.00e+00
1.00e+00
Node 7
Feat: 0.00e+00
Node 15
Feat: 0.00e+00
1.00e+00
Node 16
Feat: 0.00e+00
1.00e+00
Node 8
Feat: 0.00e+00
1.00e+00
Node 9
Feat: 0.00e+00
1.00e+00
Node 10
Feat: 0.00e+00

Results


Sample 0
Sample 1
Sample 2
Sample 3
Sample 4
Sample 5
Sample 6
Sample 7

References

[1]. Shi, Jiaqi, et al. "Skeleton-based emotion recognition based on two-stream self-attention enhanced spatial-temporal graph convolutional network." Sensors 21.1 (2020): 205.

[2]. Ahmed, Ferdous, ASM Hossain Bari, and Marina L. Gavrilova. "Emotion recognition from body movement." IEEE Access 8 (2019): 11761-11781.

[3]. Bota, Patricia J., et al. "A review, current challenges, and future possibilities on emotion recognition using machine learning and physiological signals." IEEE access 7 (2019): 140990-141020.

[4]. Picard, Rosalind W. Affective computing. MIT press, 2000.

[5]. https://www.cs.cityu.edu.hk/~howard/Teaching/CS4185-5185-2007-SemA/Group12/BVH.html

[6]. Kipf, Thomas N., and Max Welling. "Semi-supervised classification with graph convolutional networks." arXiv preprint arXiv:1609.02907 (2016).

[7]. Shi, Henglin, et al. "Multiscale 3D-shift graph convolution network for emotion recognition from human actions." IEEE Intelligent Systems 37.4 (2022): 103-110.

[8].Ghaleb, Esam, et al. "Skeleton-based explainable bodily expressed emotion recognition through graph convolutional networks." 2021 16th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2021). IEEE, 2021.