Mini Project Report

Human Motion Prediction

Indian Institute of Information Technology, Allahabad

Allahabad, Uttar Pradesh, India { 211012}

Group Details:

Group Members-

➢ Udgam Shah	IIT2019186
➢ Vikram Singh	IIT2019213
➢ Eshan Vaid	IIT2019230
➢ Ayush Khandelwal	IIT2019240

Group ID- G1147585099

Guide- Prof. G.C. Nandi

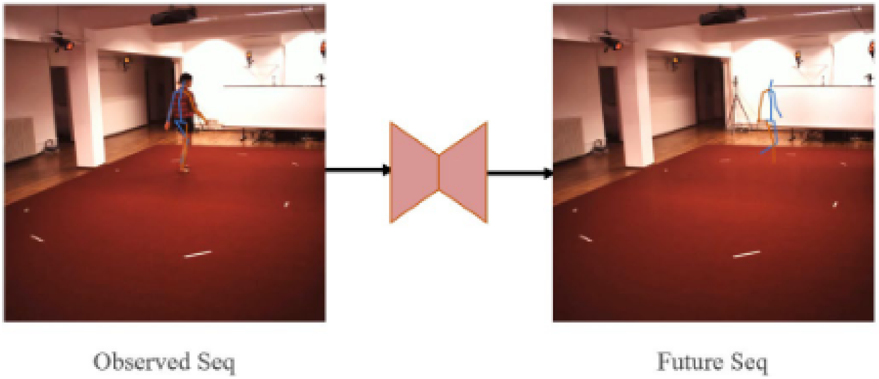
Content

ABSTRACT

Due to the sheer stochasticity and aperiodicity of future postures, predicting human motion is a difficult task. Recently, it has been demonstrated that graph convolutional networks are particularly successful in learning dynamic relations among pose joints, which is useful for posture prediction. On the other hand, a human position may be abstracted recursively to generate a series of poses at various sizes. Furthermore, human pose spatial dependence is conveyed by interpreting a human pose as a generic multi scale graphs (rather than a human skeletal kinematic tree) constructed by connections between every pair of body joints where number of joints differ in each scale and affects the training of other scale graphs so as to have a better outcome due to different range of movement of very joint in different types of motions. Instead of utilising a predefined graph structure, we create novel graph convolutional networks to autonomously learn graph connection. This enables the network to catch long-term relationships that are not captured by the human kinematic tree.

Joint angles and **3D joint coordinates** are the two primary approaches to express human posture in the motion prediction literature. These two representations, on the other hand, are totally static. The input image is transformed in the frequency domain and the model learns parameters based on it, which will give us the image in the frequency domain. We use the IDCT function to convert the output in the original dimension and compare it with the desired output.

Instead of directly creating convolutional graphs we first convert the prediction literature, we suggest encoding the temporal aspect of human motion directly in our representation and working in trajectory space. It is worth noting that, in the end, we must construct human postures in a standard format, and our formalism applies to both of the above-mentioned ones, as proven by our tests.



INTRODUCTION

The goal of human motion prediction is to anticipate future human poses based on a past posture sequence. Predicting human motion is a difficult endeavour because of the stochasticity and aperiodicity of future postures. Human motion prediction is important in several domains, including human-computer interaction, autonomous driving, and video completion. In this project, we use pre recorded human motion data sequences and forecast how the motion will continue for several frames in the future.

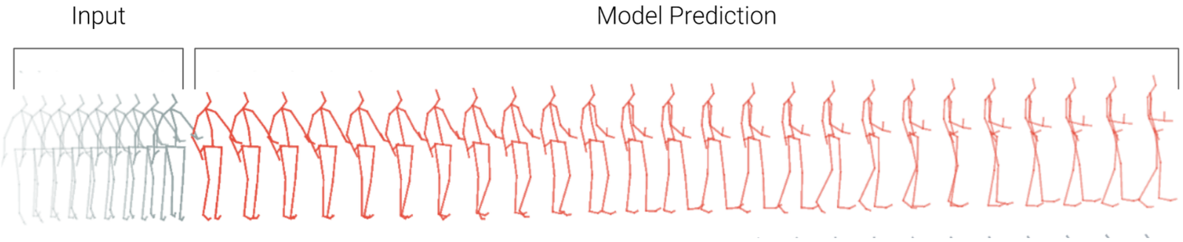


Fig . Prediction of human motion, from given input motion

Prediction of human motions is a key for safe navigation of autonomous robots amongst humans but as we know expecting human movements may be a challenging assignment due to the stochasticity and aperiodicity of future postures, and also the necessity of robustness in the distributional shift. In this paper we have used DCT transformations with MSR-GCN to get better results in future pose prediction.

The input picture is converted to the frequency domain, and the model learns parameters based on it, giving us the image in the frequency domain, which we then compare to the intended output using the IDCT function. The model which we used is a MultiScale Residual Graph Convolution Network (MSR-GCN) which extracts features from input at different scales from fine to coarse scale and then from coarse to fine scale and which is then combined to obtain the residual value between input and target poses.

DATASET

We conducted our experiments of human motion prediction on the H3.6m mocap Dataset which is the largest human motion dataset for 3D body pose analysis. It contains 3.6 million images of 3D human poses with 6 male and 5 females doing 17 different scenarios like smoking ,taking photos,talking on phone etc. We took 22 body joints from the original dataset which had 32 joins and then changed it from exponential mapping format to the 3D joint coordinate space. The other dataset which we used was CMU mocap which has an original of 38 joints. We changed it again to 22 and the rest of the other description is similar to the H3.6m dataset.

Before delving into the implementation, we'll present two common benchmark motion capture datasets that we'll be using: **Human3.6M (H3.6M)** and **CMU Mocap**.

- □ **Human3.6M** : Every subject in the H3.6M dataset has its own set of 15 action categories, while the dataset as a whole comprises 15 action categories for all seven subjects. For each posture in the sequence, we downsample by 2 along the time axis and pick 22 body joints from a pool of 32 joints that were originally in the exponential mapping format. This dataset is utilised for testing and validation, whereas the remainder of the data is used for training. In the descending and ascending sections, we employ four scales, each of which has 22, 12, 7, and 4 joints.

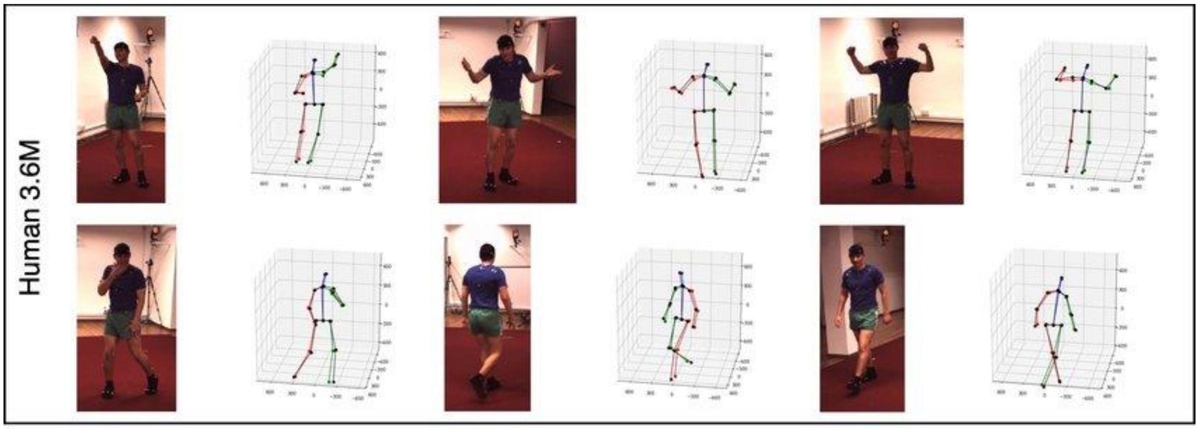


Fig . Left: the input images; Right: the results of 3D pose prediction from a different viewpoint, the black skeleton is the ground truth of the Human3.6M dataset.

□ **CMU Mocap** : Another widely used dataset for predicting human posture is the CMU Mocap dataset, which comprises eight action categories. Each posture has 38 body joints in the original dataset; we abstracted these down to 12, 7, and 4 joints for each of the remaining 25. Other information is comparable to that found in H3.6M.

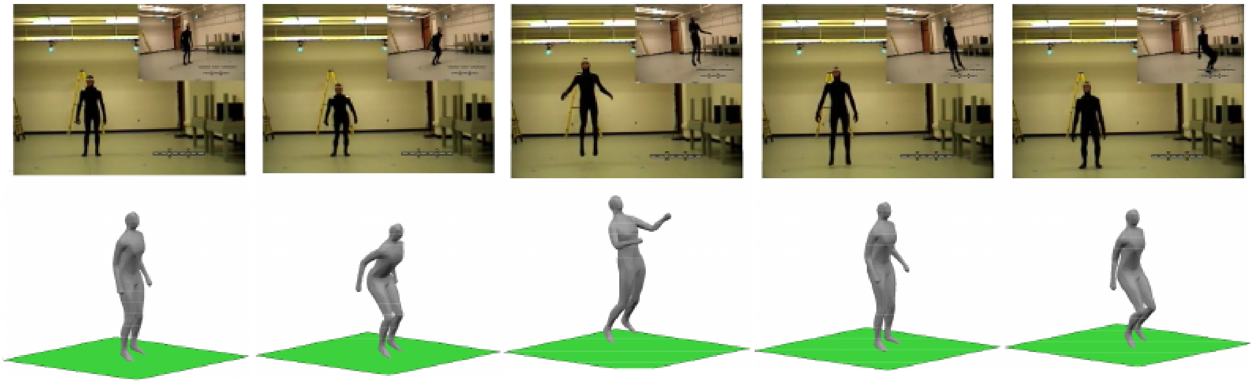


Fig . Running sequence 35/17 of the CMU MoCap data set reconstructed with the non-periodic reconstruction

PREVIOUS WORK

Human motion prediction:

Traditional approaches such as the hidden Markov model, linear dynamic system, restricted Boltzmann machine, Gaussian process latent variable models, and random forests can handle simple periodic motion patterns, but more complicated motion is intractable for these methods. Many deep learning-based approaches for human motion prediction have been proposed. Existing CNN-based studies interpret a pose sequence as a two-dimensional matrix, with one axis representing the spatial axis and the other representing the temporal axis, and then apply spatiotemporal convolutional filters to the posture data, similar to how an image has been done. Pose data, on the other hand, is fundamentally different from images in that it lacks repeating parts that offer a large response to the same filter, limiting the convolutions' efficiency. Although RNN-based algorithms offer advantages when dealing with time-related activities, discontinuity and error accumulation issues are frequently encountered due to the frame-by-frame prediction approach. Furthermore, RNN model training is prone to collapsing due to gradient explosion or disappearance. Furthermore, these networks ignore the inner-frame kinematic relationships that exist between body joints. The generative adversarial networks are thought to produce realistic data with a pattern comparable to the training data.

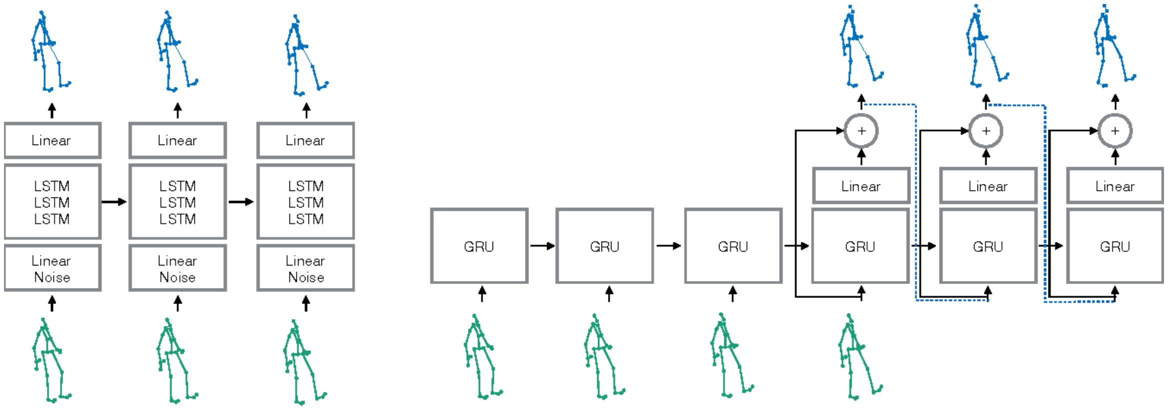
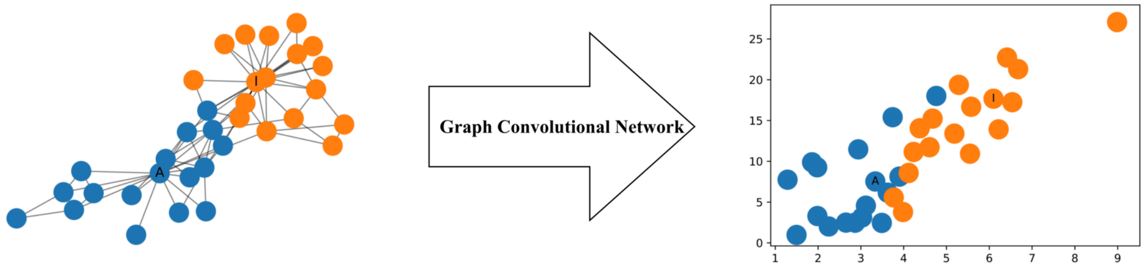


Fig. Overview of training procedure in predicting human motion using Recurrent Neural Networks

Graph Convolutional Networks (GCNs):

Graph Convolutional Networks (GCNs): are well suited for tasks involving non-grid and graph-structural data, such as biological genes, point clouds, human social networks , and human motion prediction due to the skeleton's graph-structure character. Visual recognition, object identification, action localization, trajectory prediction , and picture captioning are just a few of the areas where they've proven effective. Si et al. [Chenyang Si, Wentao Chen, Wei Wang, Liang Wang, and Tieniu Tan. An attention enhanced graph convolutional lstm network for skeleton-based action recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 1227–1236, 2019.] integrate graph convolution with LSTM to improve its capacity to represent temporal connections between human skeleton joints, because graph convolution is more disposed to collect spatial information. Mao et al. [Wei Mao,Miaomiao Liu, Mathieu Salzmann, and Hongdong Li. Learning trajectory dependencies for human motion prediction. In Proceedings of the IEEE International Conference on Computer Vision, pages 9489–9497, 2019.] design a fully connected GCN to allow adaptively learning the connectivity necessary for motion prediction tasks by applying discrete cosine transformation(DCT). According to Cui et al. [Qiongjie Cui, Hua Jiang Sun, and Fei Yang. Learning dynamic relationships for 3d human motion prediction. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 6519–6527, 2020.], all edges of the fully connected graph benefit from natural joint connectivity.



Multi-Scale Residual Graph Convolution Networks

To understand this we first need to understand how a graph neural network is depicted. Human body is basically composed of joints and each joint play a different role in prediction of next pose.Now here we abstract human body as a fully connected graph whose vertices are the pose joints, we employ graph convolution networks to dynamically learn the relations between all pairs of joints flexibly regardless of the physical distance between them. Now for this we use GCN after passing the input through each layer of GCN the vertices know more about their context in the graph and as for our model the more the joints know the relation between them the better. Now we know that after passing through a gcn layer the relation between joints are understood better. But as proposed in previous papers the motion at coarser level is very much stable. Then in finer level therefore multiple gcn’s have been for different scales so that first the motion prediction at coarser level is improved and then the finer.

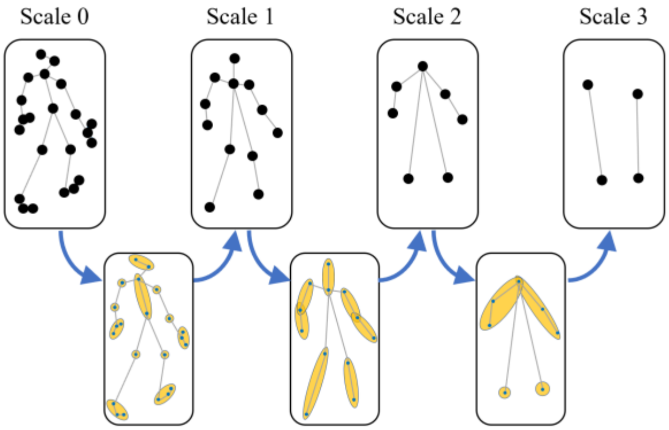


Fig. Step by step, a human position may be abstracted to create a succession of poses ranging from fine to coarse in scale by grouping nearby joints together and replacing the group with a single joint

There are sets of ascending and descending gcn which extract features at different scales and then decode them to get the target output further while decoding the output we apply a residual connection between the input and output at every scale making the whole network learn residuals instead of target poses directly.

DCT

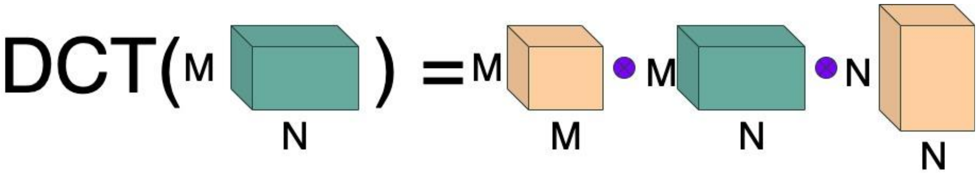
DCT stands for discrete cosine transformation; it converts an image from spatial domain to frequency domain. It basically helps in representing an image as a sum of coefficients of sinusoidal waves and mostly reduces the high frequencies in an image.

It consists of a set of basis vectors that are sampled cosine functions which helps in regenerating back the original image.

We calculate coefficient of how much each vector contributes and we find that the one with lower frequencies have much higher effects than the one with higher ones and then regenerate image after transmission using these coefficients.

$$D(i,j) = \frac{1}{\sqrt{2N}} C(i)C(j) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} p(x,y) \cos\left[\frac{(2x+1)i\pi}{2N} \right] \cos\left[\frac{(2y+1)j\pi}{2N} \right]$$

$$C(u) = \left\{ \begin{array}{ll} \frac{1}{\sqrt{2}} & \text{if } u = 0 \\ 1 & \text{if } u > 0 \end{array} \right\}$$



METHODOLOGY

We have the input historical frame of X_1:T_h = [X_1, …, X_{Th}] ∈ R J×D×T_h with T_h frames, among which X_t depicts a single 3D human pose with J joints in the D-dimensional space (here D is 3) at time t. Firstly, we reformulate our prediction objective by rearranging the input and output pose sequences. Instead of performing prediction based on X_1:T_h, we replicate the last pose X_{Th} for T_f times, obtaining a sequence of length T = T_h + T_f.

The input is transformed using Discrete Cosine Transformations (DCT). In this way each resulting coefficient encodes information of the entire sequence at a particular temporal frequency.

Given a joint, k, the position of k over N time steps is given by the trajectory vector: xk = [xk,1, . . . , xk,N] where we convert to a DCT vector of the form: Ck = [Ck,1, . . . , Ck,N] where Ck,l represents the lth DCT coefficient. For δl1 ∈ R N = [1, 0, · · · , 0], these coefficients may be computed as

$$C_{k,l} = \sqrt{\frac{2}{N}} \sum_{n=1}^{N} x_{k,n} \frac{1}{\sqrt{1+\delta_{l1}}} \cos\left(\frac{\pi}{2N}(2n-1)(l-1)\right).$$

If no frequencies are cropped, the DCT is invertible via the Inverse Discrete Cosine Transform (IDCT):

$$x_{k,l} = \sqrt{\frac{2}{N}} \sum_{l=1}^{N} C_{k,l} \frac{1}{\sqrt{1+\delta_{l1}}} \cos\left(\frac{\pi}{2N}(2n-1)(l-1)\right).$$

Then the prediction task can be translated to compute a residual vector between X_1:T and the ground truth X_1:T, which we also find very effective to improve the prediction accuracy.But before prediction we first have to apply IDCT to the output generated by END GCN's.

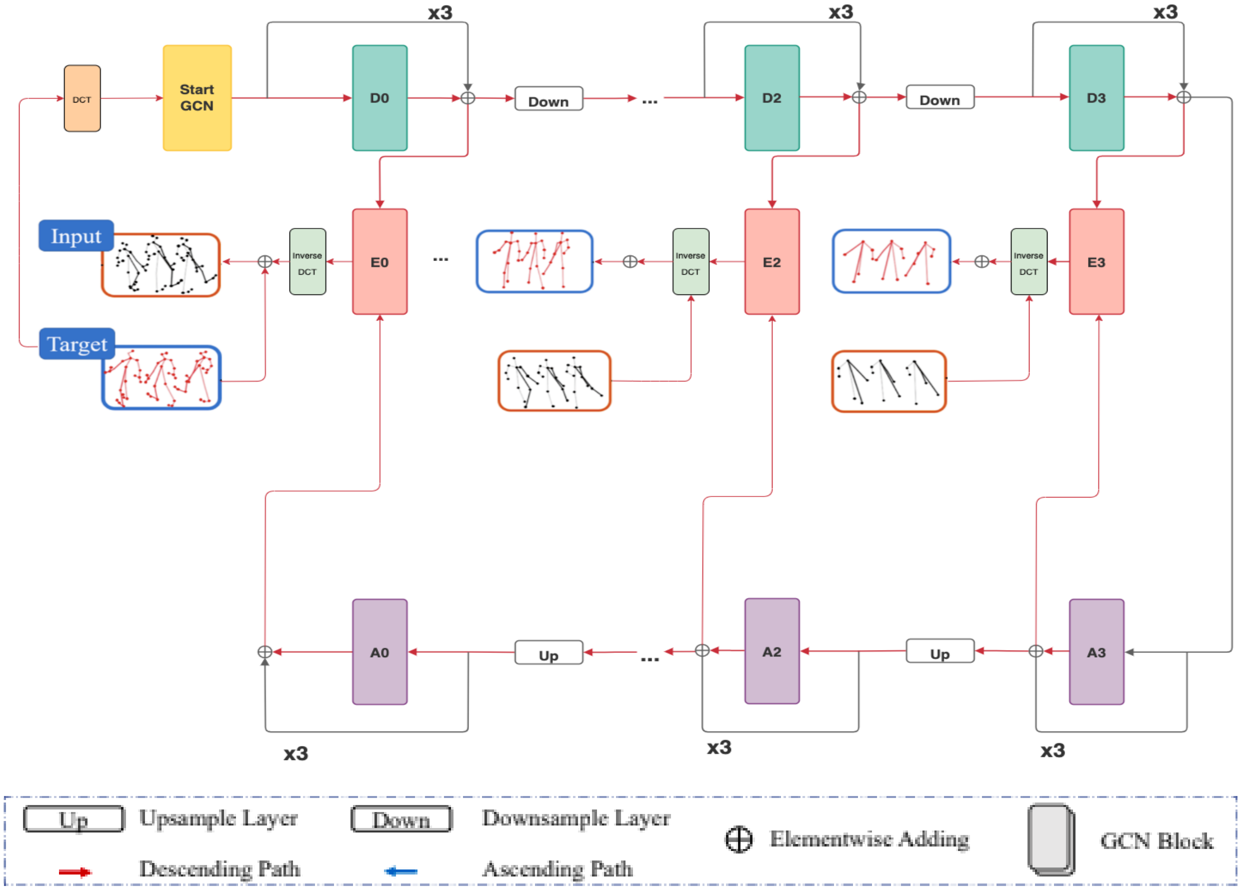
Now we have a graph with K nodes k=joints X dimensions and an adjacency matrix of shape K*K. Now GCN is composed of convolutional layers the output of layer will be

H^ L ∈ R^ K×F ^L ->output of layer l

A ^L ∈ R K×K -> adjacency matrix

W^L ∈ R ^F^L×F ^L+1 ->trainable parameters

Then for layer L+1
H^(l+1)=σ(A^LxW^LxH^L);
Here F=256
The start GCN has 2 graph convolutional layers, projecting the input pose sequence from the space of R^ (K*T) to R^(K*F)
Then 6 residual gcn with 2 layers accepts in space K*F and outputs in the same dimension to end gcn which outputs into K*T dimension and The whole network learns the residual vector between the input and target pose sequences by adding a global skip connection.



Comparison Metrics

We will use Mean Per Joint Position Error (MPJPE) as our evaluation metric. The Mean Per Joint Position Error loss is given by the following equation-

$$\mathcal{L}_{\text{MPJPE}} = \frac{1}{J \times T} \sum_{t=1}^{T} \sum_{j=1}^{J} \left\| \hat{p}_{j,t} - p_{j,t} \right\|^2$$

where p$\hat{}$ $_{j,t}$ ∈ R 3 represents the predicted j-th joint position in frame t, and p$_{j,t}$ is the corresponding ground truth.

We compare our method to three current baselines, namely Residual sup, DMGNN, and Traj-GCN. The RNN-based model is the first, while the GCN-based models are the second and third. To evaluate the Mean Per Joint Position Error, we train the networks to use the entire test dataset in 3D coordinate space.

To verify the accuracy of MSR-predictions, GCN's we will be using 400ms short-term (i.e., 10 frames) and 1000ms long-term (i.e., 25 frames) predicts on H3.6M and CMU Mocap, and compared the results to other techniques.

RESULTS

We undertake tests on two typical benchmark motion capture datasets, Human3.6M (H3.6M) and CMU Mocap, to validate the efficacy of MSR-GCN. Following results were obtained after evaluating on the two datasets:-

Human3.6M

H3.6M-20/50/50-all	80	160	320	400	560	1000
walking	11.474	24.625	38.557	46.486	54.092	63.608
eating	8.242	15.821	31.104	40.263	52.273	77.368
smoking	7.841	14.983	29.7	39.484	49.561	73.082
discussion	12.077	28.17	58.638	71.103	89.779	115.863
directions	7.091	18.84	45.251	54.076	69.729	101.429
greeting	14.868	35.661	77.263	95.32	116.992	145.739
phoning	10.629	20.141	40.124	51.597	69.76	104.33
posing	13.624	28.199	66.866	84.875	116.755	172.549
purchases	15.959	31.471	65.78	81.506	103.297	140.161
sitting	11.3	21.991	46.08	59.528	77.222	119.274
sittingdown	14.841	32.175	62.901	75.486	103.944	157.259
takingphoto	11.121	22.187	46.539	57.239	77.714	122.868
waiting	11.307	21.232	48.534	60.451	76.087	104.833
walkingdog	20.885	41.063	80.328	91.308	112.334	147.369
walkingtogether	11.155	22.078	35.602	43.082	53.304	64.222

CMU Mocap

CMU-20/50/50-all	80	160	320	400	560	1000
basketball	10.243	18.64	36.042	45.856	60.63	86.936
basketball_signal	3.006	6.37	12.951	16.978	27.164	49.904
directing_traffic	6.004	14.004	29.635	37.221	60.489	115.198
jumping	15.685	28.947	57.396	69.063	92.859	126.42
running	16.558	21.852	30.23	33.039	35.692	41.601
soccer	11.794	19.546	35.456	46.932	65.899	100.545
walking	6.39	10.608	16.802	20.673	26.129	36.287
washwindow	6.28	11.555	24.942	29.641	46.03	70.514

CONCLUSION

In this project, we integrated discrete cosine transformation with a multi-scale residual graph convolution network in this project to accurately predict future human movements from recorded histories. To offer intermediate oversight, losses are added to all scales. We predict 25 frames in the future using a brief recorded historical posture sequence of 10 frames as input. On the entire test dataset, we test and compare the suggested strategy with prior state-of-the-art methodologies. On two typical benchmark datasets, our methodology beats the state-of-the-art approaches. In the future, we will investigate multi-scale grouping methods further.

REFERENCES

1. Sünderhauf, Niko et al. "Meaningful maps with object-oriented semantic mapping." *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (2017): 5079-5085.
2. J. Canny, "A Computational Approach to Edge Detection," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679-698, Nov. 1986, doi: 10.1109/TPAMI.1986.4767851.
3. Liu, Wei & Anguelov, Dragomir & Erhan, Dumitru & Szegedy, Christian & Reed, Scott & Fu, Cheng-Yang & Berg, Alexander. (2016). SSD: Single Shot MultiBox Detector. 9905. 21-37. 10.1007/978-3-319-46448-0_2.
4. Wei, L. Miaomiao, and S. Mathieu. History repeats itself: Human motion prediction via motion attention. In ECCV, 2020.
5. Mao, M. Liu, M. Salzmann, and H. Li. Learning trajectory dependencies for human motion prediction. In Proceedings of the IEEE International Conference on Computer Vision, pages 9489–9497, 2019.
6. Bhattacharyya, M. Fritz, and B. Schiele. Long-term on-board prediction of people in traffic scenes under uncertainty. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 4194–4202, 2018.
7. Daxberger and J. M. Hernández-Lobato. Bayesian variational autoencoders for unsupervised out-of-distribution detection. arXiv preprint arXiv:1912.05651, 2019.
8. -Y. Gui, Y.-X. Wang, X. Liang, and J. M. Moura. Adversarial geometry-aware human motion prediction. In Proceedings of the European Conference on Computer Vision (ECCV), pages 786–803, 2018a