Journal of Data Acquisition and Processing

ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.778180

HUMAN TRAJECTORY PREDICTION: FUTURE LOCATION AND TRACKING WITH COMBINED DEEP LEARNING ARCHITECTURE AND YOLOV7

N. Venkata SubbaReddy
Assistant Professor, Dept of CSE, SreeNidhi Institute of Science & Tech. Hyderabd-501 301, India

Dr. D. S. R. Murthy
Professor, Dept of CSE, Anurag University, Hyderabad - 500088, India.

Abstract

In recent times, the significance of next location prediction in different fields of application has gained the attention of investigators. Traffic flow forecasting, tracking devices development are some of the applications related to this fields. With huge directional trajectory information, researchers can predict where humans will travel up next. Humans use distinct routes to prevent obstructions and give space to other pedestrians. Researchers have an interest in predicting human future trajectories according to the previous locations. This paper suggests a combined deep learning model that may train human movement patterns and predict future trajectories. Here, three steps are involved including Preprocessing step, Feature extraction step and Future location Prediction step. In the preprocessing step, input video from the dataset is converted into the frames (images) and then filtered by an improved wiener filtering process to enhance its quality. Subsequently, Texton based features, Resnet based features, VGG 16 based features, Improved Semantic features and Improved LTP features are extracted from the preprocessed image. The last step in this model is future location prediction, a hybrid deep learning architecture is proposed in this step, which is the combination of improved LSTM and Bi-GRU model. Finally, Yolo V7 technique is used for the tracking purpose. Experiments on an available public dataset demonstrate that the proposed work is efficient than the conventional models on predicting the trajectories.

Keywords- Improved filtering, Deep Learning, ILTP, Human Trajectory, Yolo V7

1.0 Introduction

Predicting the trajectory of moving humans in congested environments is crucial for autonomous vehicles as well as sociable robotic routing [11] [12]. However, the prediction issue is to create a list of future locations using information of previously recorded trajectories of a given duration [9]. Trajectory prediction of humans has gained great attention in recent years because of the development of deep learning, which allows the collection of long-term time dependencies [10] [13]. The complexity and uncertainties associated with human populations ensure that, despite an extensive number of hopeful papers, trajectory prediction issues remain unsolved. Humans organize all interactions with each other in overcrowded situations in their own unique ways and give way to those who are going in the same direction as them. The characteristics of human mobility have always made trajectory prediction a difficult task [14].

However, everyone has a goal in mind when they arrange their journey. People typically walk smoothly and by observing their prior trajectories over time, researchers might learn more about where they could finish up. Occasionally, people will suddenly change their direction of walking. According to "standard" historical tracks, such instances are unavoidable [24]. Also, socially acceptable norms, such as maintaining a secure distance behind other people can't be measured [15] [17]. Humans are able to incorporate all of their interactions with efficiency and modify their course properly [23] [25]. However, machines find it difficult. Individual preferences also influence the way individuals respond to one another. However, predicting a single course in complicated situations is irrational. It is challenging to estimate likely future trajectories using an analysis of previous trajectories [20].

The above-mentioned challenges are intended to be addressed by the advanced research work. Their primary goal is to simulate social interactions [19] [21] [22]. The development of deep neural networks like LSTM, which is a turning point for predicting actual human motion pathways, has demonstrated promise in moderating the long-term temporal dependency of trajectories [16] [18]. This paper suggested a hybrid deep leaning architecture (Bi-GRU and improved LSTM) to predict the future location. The key contribution of this proposed model is as follows:

- Proposing an Improved Wiener filter to restore the anti-noise of image (frames) by using power spectral density and Improved SNR with improved PSNR for enhancing the image quality.
- Proposing an enhanced CNN-based semantic feature extractor to capture the content of semantic information and contextual comprehension.
- Proposing an Improved Local Ternary Pattern feature extractor with improved LBP to improve the accuracy and performance of the detection.
- Proposing an improved LSTM model along with Bi-GRU model for the future location prediction process. In the improved LSTM model, the attention layer incorporates three mechanisms considered such as Score alignment, Weights and Context vectors. Additionally, their loss function is replaced by the hybrid loss function with the combination of balanced entropy loss and Dice loss function

Overall, the organization of remaining sections are as follows. Section 2.0 presented the literature review, Section 3.0 presented a future location and tracking model for the hu7man trajectory prediction by using deep learning architecture and Yolo V7 model. Section 4.0 shows the investigated output and their analysis. Section 5.0 represents the conclusion of the proposed work.

2.0 Literature Review

In 2023, P. Kothari and A. Alahi [1] focussed on the difficulties of modelling social interactions and producing a collision-free bidirectional distribution emerge while predicting human trajectories in groups. Recent research suggests a number of GAN-based designs to more accurately simulate human movements in crowds of people, building on the success of SGAN. High collisions in model predictions show that present networks fall short of producing socially acceptable paths, even though they perform better in minimizing distance-based metrics. In response, the authors provided SGANv2, an enhanced safety-compliant SGAN framework with a transformer-based detector and spatio-temporal interaction modeling. Improved temporal modelling of sequences was achieved through transformer-based discriminator design, while enhanced spatiotemporal modeling facilitates a deeper understanding of social interactions between people. Furthermore, SGANv2 makes use of the trained detector even during testing using a cooperative sampling technique that not just improves collision trajectory but also averts mode collapse, a frequent occurrence during GAN training. By means of comprehensive testing on various synthetic and actual data sets, this paper exhibits the effectiveness of SGANv2 in generating multimodal trajectories that are socially acceptable.

In 2022, Zhiquan He et al., [2] suggested an MCDIM approach for learning and predicting future paths in humans. In order to simulate the structure of human–human relationships, researchers particularly develop multiple layers of GAT. A second set of LSTM systems was created to record the relationships between these human-human exchanges across various time periods. In order to simulate the interactions among individuals and the environment, they explicitly collect and encode the regional characteristics in the individual's neighbourhood at every point in time as well as the overall scene architecture features. Then record both the temporal and spatial data of these interactions. For precise trajectory prediction, the multi-level GAT-based system incorporates human–human as well as human–scene interactions. This paper tested the strategy using two benchmark datasets. The outcomes indicate the MCDIM approach operates better than other approaches, producing human trajectories that are both more precise and believable. Regarding the standard movement errors and ultimate movement error, an average improvement was two and three percent points, correspondingly.

In 2022, Yanyan Fang et al., [3] presented a unique high-order GCN for predicting pedestrian trajectories. In particular, an intended pedestrian's traveling phase depends upon its neighbours' movements in addition to its previous trajectory, which contains information on its acceleration, traveling instructions, and velocity. In order to forecast the target pedestrian's progress, researchers therefore suggest using GCNs to combine the trajectory characteristics of the pedestrian and other people. This paper suggests using a high-order GCN for modeling human–human interaction because the motion of one pedestrians' neighbors influences the mobility of the target pedestrian's neighbor, which in turn influences the mobility of the target pedestrians indirectly. This kind of high-order GCN takes into account both the neighbors of the target individual and the neighbours of those around it. Moreover, a pedestrian is able to prevent collisions by anticipating the whereabouts of both itself as well as its neighbors. Also, it could change its course or reduce its speed if it senses that a collision is likely, particularly in densely populated areas. Given this, researchers suggest modelling anticipation-based decision-making as focus while integrating this into the high-order GCN. Therefore, using a straightforward technique, initially determine the general future motions of every pedestrian. Compute the focused attention in the attention-based high-order GCN and forecast future trajectories utilizing the coarse anticipated future trajectories and GCN output. The method's efficiency was validated through numerous tests. Furthermore, the model demonstrates increased data effectiveness. Just 5 percent of the initial training data for the ETH&UCY dataset was used.

In 2022, Pei Lv et al., [4] proposed a straightforward and understandable method of characterizing movement known as a trajectory distribution, that converts the human trajectory's values into a two-dimensional Gaussian pattern in area. This paper creates a new trajectory forecasting technique that refer to as the social likelihood approach according to this inventive definition. The technique incorporates deep combined RNNs with trajectory distribution. This approach uses trajectories distribution as input as well as output, giving the RNN enough spatial and randomized data to identify pedestrians in motion. Additionally, in order to produce accurate and reliable predictions, the societal probabilistic approach actively drives spatiotemporal elements from the updated motion specification. Tests conducted using publicly available benchmark datasets demonstrate the efficacy of the suggested approach.

In 2021, Ronny Hug et al., [5] stated that the evaluation of human trajectory forecasting algorithms requires techniques for quantifying the numerous variables of trajectories datasets. A technique for estimating the quantity of data found in a data set using a prototype-based data set illustration was suggested for the purpose to further comprehend the complexities of trajectory forecasting assignments as well as adhere to the perception which higher-level data sets include additional data. In order to facilitate a later LVQ step, a non-trivial spatial sequencing alignment was carried out to initially obtain the information's structure. Following an overview on indicators used in human trajectory forecasting and evaluation, an extensive complexity evaluation was carried out on multiple benchmark datasets for human trajectory forecast.

In 2020, Akif Hacinecipoglu et al., [6] stated that Human-aware navigation becomes essential as mobile AI begin to operate in human-populated areas in order to ensure their safe, effective, and socially acceptable navigation. In order to locate an approach to a location even with the absence of a defined method, individuals travel in an engaging and cooperate manner. Considerable work was done to address this issue for mobile robots; yet, learning-based systems tend to be highly dependent on the situation and arrangement of the set of training data, and they weren't extensible for high individuals' populations. The crowd devise a technique that uses a cost-based interactions method to collaboratively modify the starting paths for all representatives, derived from Gaussian methods.

In 2020, Pedro A. Peña & Ubbo Visser [7] focussed the challenging to monitor and predict individuals in all three dimensions so that one can determine their location and direction inside the environment. If this problem gets solved, however, a robotic agent could be able to traverse its surroundings and recognize wherever it may securely be without endangering the person it speaks with. Researchers suggest a brand-new probabilistic structure for artificial intelligence that forecasts human motion by fusing numerous algorithms into a revolving probability map.

In 2022, Quan T. Ngo et al., [8] provided a two stages paradigm which makes full use of the spatiotemporal constraints inherent in the movement of a person and any companions in order to forecast a person's eventual placements. After identifying the POI's supporters, the structure uses mobility data regarding the POI as well as a few chosen supporters to forecast the POI's potential. In this study, two companion ways to choose were presented. By comparing the geographical locations of the humans, the initial approach utilizes SC to identify the relatives of that POI. The POI's partners were selected using cosine correspondence in the subsequent technique, which creates PIE matrices. It additionally employs a stacked automatic encoder, that reduces a highly dimensional input attribute to create a low-dimensional storage vectors, to lessen the effects of dimension.

In 2021, Dapeng Zhao and Jean Oh [26] concentrates on the Robots should be capable to anticipate the motions of humans in order to move in a secure and straightforward manner, as increasingly robotics plan to work alongside humans in shared spaces. According to this paper, the authors introduced Social-PEC, a CNN-based method for learning, identifying, and extracting patterns from sequential trajectory information. According to a series of studies about the human trajectory forecasting issue, this model behaves on level with or even better than the current state of the field in several situations. In addition, the suggested method clarifies the confusion around the earlier application of pooling layers and offers an understandable explanation for the method used to make decisions.

In 2021, Dongho Choi et al., [27] In this research, the authors have provided a trajectory prediction approach using the LSTM encoder-decoder framework using the RF algorithm. The study was carried out under varied driving conditions, and the experiment's vehicle was outfitted with a camera, LIDAR sensors, and vehicular wireless communication devices for the gathering of data for training. The outcomes of the car testing indicate that, in comparison to current trajectory prediction techniques, the suggested method offers more reliable trajectory forecasting.

Table 1: Features and challenges of existing human trajectory prediction methods

Author [Citation]	Methodologies	Features	Challenges
P. Kothari and A. Alahi [1]	SGAN	It shown experimentally that SGANv2 is in minimizing models' errors without reducing distance-based measures.	Need to enhance the method for zero collision of human trajectory based on the metrics like safety critical collision
Zhiquan He et al., [2]	GAT, LSTM	The outcomes show that our MCDIM approach performs better than other approaches, producing more accurate and precise trajectories for humans.	Need to reduce the computational complexity of GAT in the real time applications for quick prediction

Yanyan Fang et al., [3]	GCN	This approach exceeds the existing state of the art based on the ETH&UCY dataset, utilizing just 5 percent of the initial training data for every training period.	Have to consider the future location prediction foe accuracy and enhance the proposed model
Pei Lv et al., [4]	2D Gaussian distribution	From new moment description. The spatial feature was extracted based on generating robust and predicting accuracy.	Need to involve physical environment in the proposed model for accurate prediction
Ronny Hug et al., [5]	LVQ	Several datasets are used in this paper to obtain the accuracy of long-term human trajectory prediction	Have to consider the tracking of the location in human trajectory prediction model with various datasets for better accuracy
Akif Hacinecipoglu et al., [6]	Gaussian approach	This proposed method has attained great accuracy in human-likeness of predicted trajectories for realistic navigation performance	Need to measures the future location predicting parameters for accurate human navigation techniques
Pedro A. Peña & Ubbo Visser [7]	Robotic system (ITP-IRL framework)	The performance of the suggested model attained more accurate probability map’s parameterization.	Need to considered the future pose of the proposed model with suitable dataset
Quan T. Ngo et al., [8]	POI, PIE	The performance of the recommended	Have to involve the semantic context of

		model outperforms over other techniques for predicting human future location through two real-world datasets	human mobility prediction and also enhance the accuracy of model in terms of time and personal preferences
Dapeng Zhao and Jean Oh [26]	CNN, Social-PEC	This proposed model has overcome the prediction problem for human trajectory as compared to the other traditional methods	Need to involve the limitation of physical environment in the recommended model for accurate prediction in human trajectory
Dongho Choi et al., [27]	RF, LSTM	This model provided more reliable trajectory prediction as compared to the other prediction model	Have to create the model of EV trajectory prediction and sound (warning sound) is created before collision for better prediction.

2.1 Problem Statement

While there is literature addressing the issue of identifying common trends in the trajectories of human, the majority of research that have been conducted to this point has focused solely at the spatial aspects of user trajectories. Predicting the location of human trajectory has become a popular area of research, although there are issues with prediction accuracy and under optimized forecasting algorithms. When modelling human motion, one must take into account people's patterns of motion in the places. Complex sequence change regularities are observed in human motion. Organizing, evaluating, and analyzing trajectory information for predicting a person's future locations is difficult. In comparison to most advanced future location predicting methods, storing and processing large amounts of trajectory data is a challenging function requiring complex algorithms for data processing to yield accurate location predictions. Table 1 shows the features and challenges involved in the traditional human trajectory prediction model. According to Table 1, researchers also identified other issues, such as where to effectively combine the trajectories which best fit various behavior of users along with ways to improve characteristics and crucial features representation. The ability to store time as a location, or temporal information, is a significant aspect that influences the manner in which a future location prediction occurs. Therefore, a combined deep learning model for future location based human trajectory prediction and tracking with Yolov7 model is proposed in this paper.

3.0 Proposed human trajectory for future location and tracking prediction model

Predicting future location using previously recorded locations is the goal of human trajectory prediction, which is typically regarded as a time sequence challenge. It is widely applicable to mobile robots, intelligent automobiles, and other applications where it can reduce accident risks in an efficient manner. Humans walk in a highly stochastic manner since they are not constrained by rules while stopping, turning, or interacting with their neighbor. Certain techniques acquire spatiotemporal information, or the social restrictions among people, to simulate the human trajectories. This paper proposes a new human trajectory prediction and tracking model along with combined Deep learning Architecture and Yolo V7 model. The following description shows the step-by-step process for the proposed model.

- At first, pre-processing phase is carried out. Here, initially, video to frame conversion takes place and then applying Improved Wiener filter to enhance the input image (frame).
- The next step is the feature extraction process. Here, features like Texton based features, Resnet based features, VGG 16 based features, Improved Semantic features, Improved Local Ternary pattern features are extracted from the preprocessed image (frame).
- At last, Future location prediction and tracking process is carried out by using hybrid model and Yolo v7 respectively. Here, the hybrid prediction model is the combination of Improved LSTM and Bi-GRU model. Figure 1 shows the step-by-step process of the prediction model.

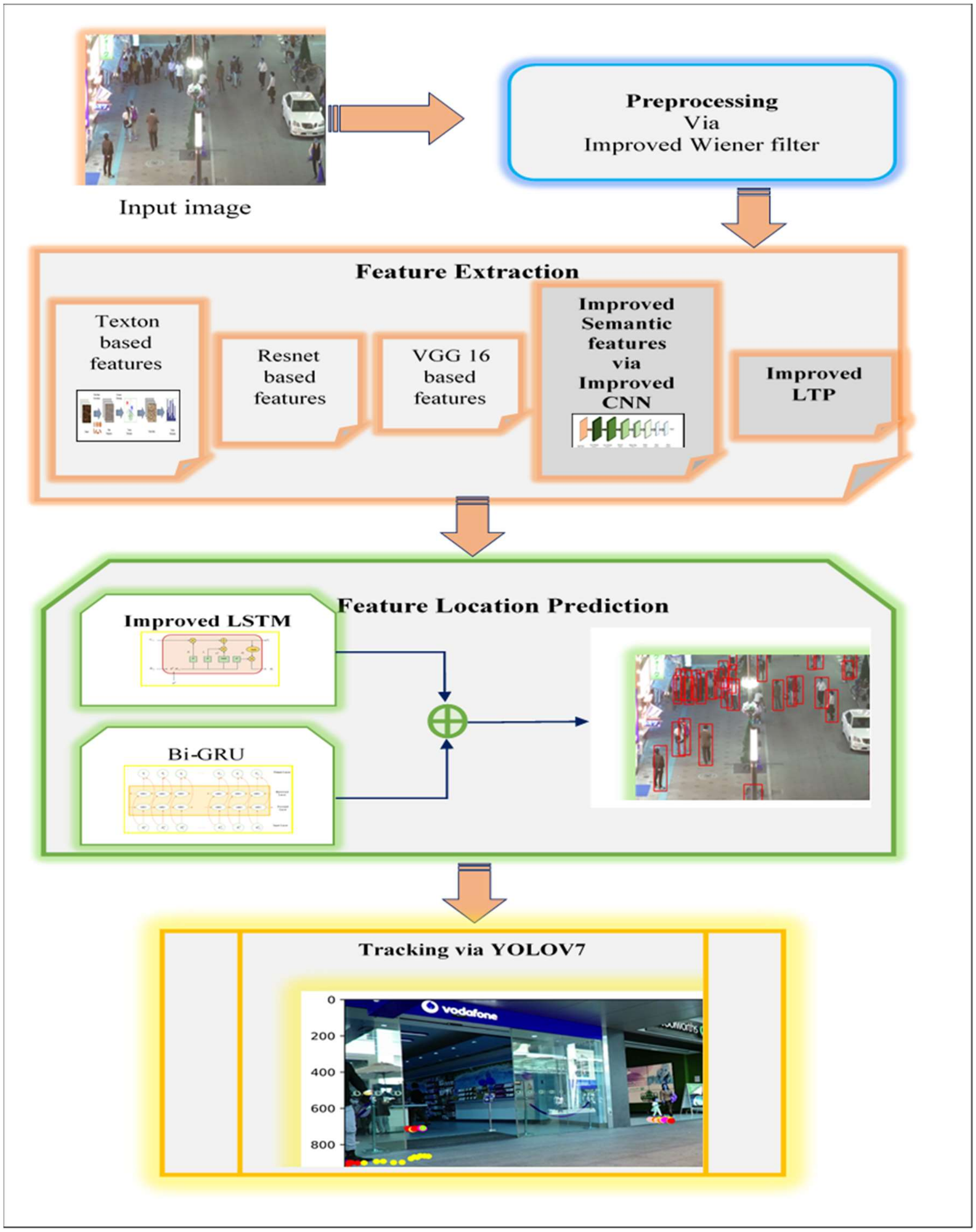


Figure 1: Architecture for the prediction model

3.1 Pre-processing via Improved Wiener filter

In this human trajectory prediction model, pre-processing is the initial step to reduce the noise and image enhancement. At first, input video to frame (image) conversion takes place and then filtered by using Improved Weiner filter. Here, the input video is represented as $I' = [\hat{I}, ...\hat{I}_n]$ with frames $\hat{I}$ and then applying improved wiener filter to enhance it. Normally, wiener filtering method [28] is used for extra noise elimination. Based on the information gathered from every pixel's local region, the low pass wiener filters employ the pixel-pixel adapting techniques. As it filters, it calculates the local mean and variance. It uses inverse filtering to accomplish the deconvolution as well as a compression technique in order to get removal of the noise. The working process of wiener filter are as follows:

Step (i)- Calculate the original and noisy image's power spectra, which are the Fourier transforms of the auto-correlation function.

Step (ii)- Use a mask in order to cover a noisy pixel in an image.

Step (iii)- Determine the variance, σ^2 as well as local mean, μ .

Step (iv)-Use the variance, mean and noise power to calculate the new pixel value.

Step (v)- Go through step (ii) to step (iv) again for every pixel in a noisy image.

According to this work, an Improved wiener filtering process is proposed to enhance the image and adaptive effective solution for reducing the noise in the image. Also improve the visual fidelity in several applications. The following steps are used in the improved wiener filtering process [28]

Step (1): Calculate the power spectral density (Fourier transform of auto correction function) and improved SNR with Improved Peak Signal to Noise Ratio which is used to estimate the image quality.

In the first step, additionally improved SNR with improved PSNR is calculated to improve the efficiency and applicability in evaluating image quality. By comparing the amount of the intended image content to the amount of noise from the background that exists in the image, the SNR, a metric applied to measure image quality, is determined. A greater SNR [29] value in the processing of images typically denotes a decrease in noise interference and improved quality of the image. To enhance the PSNR [30] metric for enhance the quality of image as compared to the original image, Eq. (1) shows the improved PSNR formula. where, $IPSNR$ represents the improved PSNR, $G'(i,j)$ represents the gray value of pixel in ith row and jth column of restored images, $G_o(i,j)$ represents the pixel's gray value of original image in ith row and jth column. P and Q represents the matrix dimension of the image. The mathematical expression of the improved SNR is shown in Eq.(2). Where, $ISNR$ denotes the improved SNR, $R.V$ represents the Relative variance, which is defined as the ratio between the variance to mean value. RMS indicates the Root Mean Square value, which is used to measuring the varying quantity's magnitude. First, the arithmetic mean of squares of value in the set and then take the square root of the obtained mean to get the RMS value, which is shown in Eq. (3). Where N represents the number of measurements, x_i represents each value.

$$IPSNR = \left\{ \frac{10\log_{10}\left[\frac{255^2 \times P \times Q \times \left|G'(i,j)\right| - \left|G_o(i,j)\right|}{\sum_{i=1}^{P}\sum_{j=1}^{Q}\left|G'(i,j) - G_o(i,j)\right|^2}\right]}{\left(1+e^{-\left|G'(i,j)\right|}\right)} \right\} \tag{1}$$

$$ISNR = 10 \times \log\left[\frac{\left|RMS\right|^2}{2 \times R.V}\right] + IPSNR \tag{2}$$

$$RMS = \sqrt{\frac{1}{N}\sum_{i=1}^{N} x_i^2} \tag{3}$$

Step (2): Over a noisy image pixel, apply a mask.
Step (3): Determine the median and put it in the center pixel of the mask.
Step (4): Calculate the variance and truncated mean.

The truncated mean formula can be applied for this calculation. Following the sorting of the pixel values, certain percentage of the highest and lowest values are eliminated, and the mean is then calculated using the values that remain. The mathematical expression of Truncated mean is shown in Eq.(4). Where, $l$represents the number of pixels to be truncated from both ends. N represents the total number of pixels in the given image ($\hat{I}$). The primary benefits of this is that it is used to decrease the impact caused by anomalies, noise, and outliers in pixel values. Moreover, Eq. (5) shows the variance of the image. Variance measures is used for the image enhancement and estimation of noise. Where, i, j denotes the rows and columns of a given frame or image $\hat{I}$. μ indicates the mean intensity value of all pixels in the image.

$$T_{N,l} = \frac{1}{N-2l}\sum_{i=l+1}^{N-2l} x_i \tag{4}$$

$$\sigma^2 = \frac{1}{ij}\left(\hat{I}(i,j) - \mu\right)^2 \tag{5}$$

Step (5): Calculate the new pixel value $D(i,j)$

As per Eq. (6), the new pixel value is estimated. where, V^2 denotes the noise variance, M denotes the median value, σ^2 denotes the variance

$$D(i,j) = M + \frac{\sigma^2 - V^2}{\sigma^2}\left(\hat{I}(i,j) - M\right) \tag{6}$$

Step (6): For all the pixels in the noisy image, repeat the step (2) to Step (6)

Therefore, the resultant enhanced filtered image from the improved wiener filter process (preprocessed image) is denoted as $\hat{I}^F$, which is subjected to the next process.

3.2 Feature Extraction process

After the Preprocessing phase, certain features are extracted from the pre-processed image, $\hat{I}^F$. The features include Texton based features, Resnet based features, VGG 16 based features, Improved Semantic features and Improved Local Ternary Patter features are extracted. In this section, each feature extraction process is explained as follows.

3.2.1 Texton based features $\left(T^F\right)$

The basic idea of texton [31] is that it is simple visual features or structures that can be utilized to define and explain the textural features of a preprocessed image. This feature is indicated as $\left(T^F\right)$,. Usually, texton-based feature extraction involves the steps given below, and the diagrammatic representation is shown in Figure 2.

(a) Filter Bank Generation: Generating a filter bank from a collection of filters (e.g., steerable or Gabor filters) to capture various textures and frequency in the image. Textons are particularly the center points of cluster in a localised filtering responses area that are produced by convolutions of a training group of images using spatially orientated functions placed in the filter bank.

(b) Filter Response Computation: The existence and intensity of distinct texture pattern at various spatial positions are represented by the filter responses, which are computed by applying the bank of filters to the image.

(c) Texton Dictionary Development: In this process, k-means clustering is usually used to find the centers of the cluster in the feature area to which every pixel in the image is mapped.

(d) Histogram Representation: This involves making histograms that show how textons are distributed throughout the image as well as frequently various textons appear.

Therefore, the Texton-based feature extraction, which is represented by $\left(T^F\right)$, makes it possible to characterize complicated patterns and textures in images and offers a reliable representation of the image details.

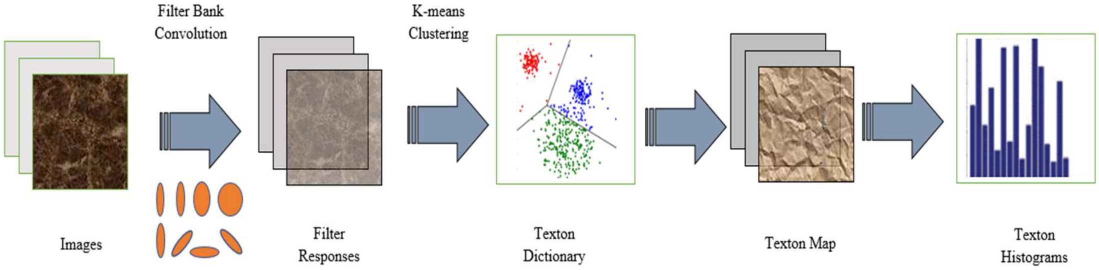


Figure 2: Texton based feature extraction process

3.2.2 ResNet based features $\left(R^F\right)$

A DCNN structure called ResNet [32], or Residual Network, has produced significant advances in classification of images tasks. Because it solves the gradient vanishing issue, ResNet has become known for the efficient training of highly dense networks. By introducing the idea of residual training into the design, the network can be trained to acquire residual mappings rather than trying to acquire the intended underlying mapping directly. Skip connections are used to accomplish this, allowing the gradient to propagate and making the learning of deep networks easier.

When a pre-trained ResNet is used for feature extraction, it functions as a feature extractor; the deeper layers extract deeper and sophisticated features, while the earliest layers retrieve low-level features like boundaries and textures. Usually, the result of one of the intermediate layers is used as a feature structure that represents the preprocessed image ($\hat{I}^F$), and the fully connected layers of the neural network are eliminated in this process.

For instance, based on the demands of the particular task, features can be extracted from various layers in a pre-trained ResNet model, including the final layer of convolution or a layer nearer to the input. Subsequently, the prediction step of the human trajectory prediction process might employ these extracted features R^F as input.

3.2.3 VGG 16 based features $\left(VGG^F\right)$

Deep convolutional neural network structure, VGG-16 [33] has overall 16 layers like 13 layers of convolution and 3 fully connected layers, and is recognized for being simple and uniform in architecture. applying an already trained VGG-16 system as a feature extractor, feature extraction entails utilizing the network's numerous layers to gather various kinds of image features from a preprocessed image. Deeper layers typically capture more complicated and complex features, whereas the beginning layers usually catch low-level features like borders and texture. The result obtained from any of the intermediate layers is usually employed as a feature representation of the input image, $\hat{I}^F$ when extracting features via the pre-trained VGG-16 model. This involves removing the last fully connected layers. Subsequently, the DL prediction model can be trained with these features. In TL scenarios, the trained VGG-16 model is used as a feature extraction tool for novel dataset or alternatives image-related tasks, where VGG-16 derived features are commonly used. As the VGG-16 model has trained to identify and describe complicated features inside images, it frequently performs better on tasks with fewer details when it is able to utilize the numerous feature representations that learned through initial training. The extracted VGG 16 feature is denoted as VGG^F.

3.2.4 Improved Semantic features $\left(IS^F\right)$

The technique of identifying and expressing the fundamental semantic content and essential concepts contained in an image is known as semantic feature extraction. By analyzing the general structure and significance of the visual information, this procedure seeks to make more sophisticated image analysis, categorization, and understanding possible. In this research, an improved semantic feature is proposed to get better semantic content and higher accuracy in the visual data. Here, the preprocessed image's semantic properties are extracted [34] using an improved CNN model.

With the use of improved CNN structure, objects, scenes, and sophisticated visual structures can be recognized with a higher level of context and comprehension as the semantic information. There are five layers in the traditional CNN model [35] like the input, convolution, pooling, fully connected, and output layers. Improved CNNs, based to the description, are developments, adjustments, or improvements made to the structure or training procedure of traditional CNN models to solve particular inadequacies or boost the models' performance in a variety of tasks. In this case, the preprocessed image $\hat{I}^{F}$ serves as the improved CNN model's input. which, using 32 filter sizes, is delivered to the convolution layer 1. The output of convolution layer 1 is then transmitted to convolution layer 2, which has a filter size of 64. In order to prevent overfitting, their result after the second convolution layer is to enter the dropout layer at 0.5. After that, the max pooling layer receives the outcome of the dropout layer before it is passed on to the flatten layer. Next, the two dense layers with 100 and 1, respectively, receive the result that comes out of the flatter layer. This work improves the activation function in the first dense layer, and the second dense layer serves as the output layer for the improved CNN model. An enhanced activation function is used to get over the vanishing gradient issue and the output's restricted range. It is made up of the Leaky ReLU activation function combined with Softmax. Specifically, the Leaky ReLU activation function incorporates the softmax activation function. Figure 3 depicts the improved CNN architecture.

A kind of activation function utilized in neural computing is the Softmax function. It is applied to the computation of the distribution of probabilities from a real number vectors. The result of the Softmax function is a range of value from 0 to 1, where the total of the probability values equals 1. The Softmax function [36] is computed by using Eq.(7) for $i = 1, 2, ..., L$. Where, the exponential of the ith element is denoted by e^{x_j}. The base of the natural logarithm is denoted by e, and the ith element of the input vector x from the flatten layer is indicated by x_i.

$$S\left(f\left(x\right)\right) = \frac{e^{x_i}}{\sum_{j=1}^{L} e^{x_j}} \tag{7}$$

An activation function called Leaky ReLU is frequently employed in NNs, particularly when addressing the vanishing gradient issue. It is a variation on the standard ReLU function that allows a tiny gradient to be transmitted over a tiny slope at a negative value, so addressing the problem of dead neurons. Eq. (8) illustrates the way to calculate the leaky ReLU activation function. Whereas x indicates the function's input and χ indicates a tiny constant slope that is usually set to a tiny value of 0.01 or 0.001. The Leaky ReLU function [36] avoids the problem of dead neurons and permits some gradients to move while backpropagation, in contrast to the ReLU function, that sets every value that is negative to zero.

$$LR\left(f\left(x\right)\right) = \begin{cases} x & if\, x < 0 \\ \chi x & if\, x \le 0 \end{cases} \tag{8}$$

According to the improved activation function, Eq. (7) of Softmax function is incorporated in Eq. (8) of Leaky ReLU activation function, the calculation of this activation function is shown in Eq. (9). Here, the value of χ is 0.01.

$$S-LR\left(f\left(x\right)\right)=\begin{cases} x & if\ x<0 \\ \dfrac{e^{x_i}}{\sum_{j=1}^{L} e^{x_j}} & if\ \ 0>x\geq -1 \\ \chi x & if\ x\leq 0 \end{cases} \tag{9}$$

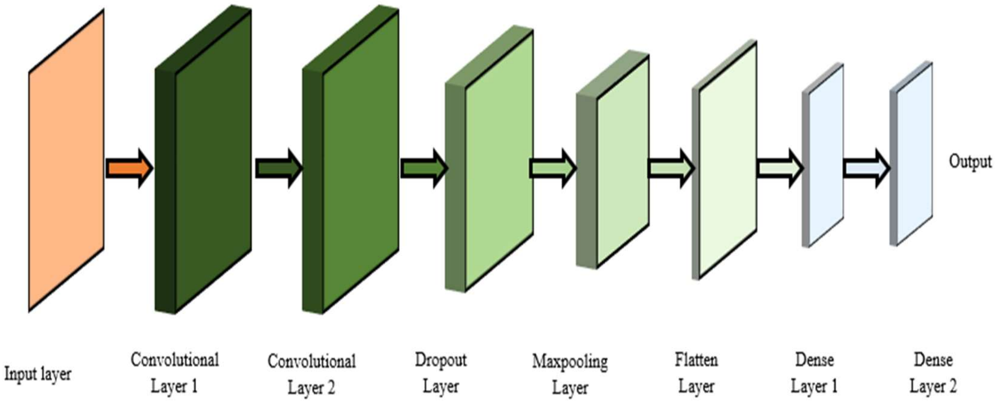


Figure 3: Improved Semantic features based on improved CNN model

3.2.5 Improved Local Ternary Pattern features $\left(ILTP^{F}\right)$

For image analysis and recognizing patterns, the LTP texture descriptor is employed. By capturing more complex local patterns in images, it expands on the concept of the LBP [37]. A better variant of the conventional LTP, or Improved LTP, is utilized in image analysis to extract features. Accordingly, from the $\hat{I}^{F}$, this work extracts the improved LTP features. It seeks to extract textures and more complex local designs from image data. Recognition of patterns and textural classification are two common uses for LBP, a well-known feature classifier. It uses a circular sequence to encode the connection that exists between each pixel and its surrounding pixels. It explains the image's local spatial arrangement in Eq. (10). Here, g_i indicates the gray value of the neighbouring pixels surrounding the center pixel, g_c indicates the gray value of the center pixel. In this case, LBP uses a fixed-size local neighborhood for pattern analysis, which might not be appropriate for extracting texture data at various granularities or scales. This constraint may limit the extent to which LBP features respond to textures with different spatial properties. Improved LTP is suggested as a solution to these problems. It is based on the neighboring pixel value and the center's gray value, which are determined using the following formulas.

$$f_{N,R}=\sum_{i=0}^{N-1}2^{i}s\left(g_i-g_c\right), \qquad s(x)=\begin{cases}1, & x\geq 0\\ 0, & x<0\end{cases} \tag{10}$$

From Eq. (10), the value of neighbor pixel is calculating as per Eq. (11). Likewise, Eq. (12) shows the calculation of the centre pixel value. Where L_x^g indicates the left pixel and R_x^g indicates the right pixel.

$$g_i = \left[x^2 - \left(\frac{1}{|x|} \right) * \left(1 + e^{-x} \right) \right] \left[\frac{L_x^g + R_x^g}{2} \right] \tag{11}$$

$$g_c = \left[\frac{1}{N} \sum_{i=0}^{N-1} g_i \right] \tag{12}$$

The threshold for the LBP is the central pixel, and it is not noise-resistant. Triggs and Tan work on the LTP. The sign function $s(x)$ is replaced by a 3-valued functional as Eq.(13) by LTP, which expands LBP to 3-values code.

$$s'(x) = \begin{cases} 1, & x \geq T \\ 0, & |x| < T \\ -1, & x \leq -T \end{cases} \tag{13}$$

where T indicates the specified threshold and can strengthen the noise resistance of the LTP code. Positive and negative components are further divided into every ternary pattern. The two components are handled as two independent LBP descriptor channel. Determine the positive as well as negative channels' corresponding histograms. As the last feature definition of the initial image, combine all the histograms. Thus, the total feature set is indicated as E^F, $E^F = \left[T^F \;\; R^F \;\; VGG^F \;\; IS^F \;\; ILTP^F \right]$.

3.3 Future Location prediction

This is the final stage of the proposed work, which explains about the prediction of future location. Here, a new hybrid model is proposed for predicting the future location, which is the combination of improved LSTM and Bi-GRU model that is explained as follows.

3.3.1 Improved LSTM

The structure of LSTM is an RNN. The gradient explosion as well as gradient vanishing issues with RNN are primarily resolved by LSTM [38]. LSTM is a member of the feedback NN group in the discipline of DL. Three gates make up an LSTM: an input gate, an output gate, and a forget gate. The Eq. (14), Eq. (15) and Eq. (16) represent the LSTM input gate, where E^F stands for a given input, H_t for the actual output, and H_{t-1} for the past output. The terms C_t and C_{t-1} denote the present and past states of the cell. The determination of new data stored within the cell state is achieved by using Eq,(15) and Eq.(16), which include sending H_{t-1} and E^F via a sigmoid layer and a tanh layer, correspondingly. Weight matrix and input gate bias are denoted by the terms P_i and Q_i, respectively. By applying Eq. (15) to the outcomes of sigmoid in Eq. (14) and tanh in Eq. (16), a new cell state is produced.

Eq. (17) represents the forget gate, where Q_F indicates the offset and P_F is considered as the weight matrix. To determine the probability of forgetting certain details from the previous cell, the sigmoid and dot product are utilized.

The output gate weight and bias of the LSTM are denoted by P_O and Q_O in Eq. (18). The final output is calculated using H_{t-1} and E^F, and is then multiplied by the tanh of the present state of the recently acquired data C_t through Eq. (19). Figure 4 shows the improved LSTM architecture.

$$I_t = \sigma\left[P_I \cdot \left(H_{t-1}, E^F\right) + Q_B\right] \tag{14}$$

$$C_t = I_t * C_t'' + F_t * C_{t-1} \tag{15}$$

$$C_t'' = \tanh\left[P_C \cdot \left(H_{t-1}, E^F\right) + Q_C\right] \tag{16}$$

$$F_t = \sigma\left[P_F \cdot \left(H_{t-1}, E^F\right) + Q_F\right] \tag{17}$$

$$O_t = \sigma\left[P_O \cdot \left(H_{t-1}, E^F\right) + Q_O\right] \tag{18}$$

$$H_t = \tanh\left(C_t\right) * O_t \tag{19}$$

By employing a weighted total of all prior hidden states, the attention mechanism leads the decoder's attention to the most relevant aspects of the input pattern. The input value of the attention mechanisms layer is $\hat{H}_t$, given the hidden state of the LSTM layer $\hat{H}_t = \left[H_1, ..., H_t\right]$ at each time step. Three phases are carried out by the attention layer: scores alignment, weights, and context vector [38]. Here, the attention mechanism was employed.

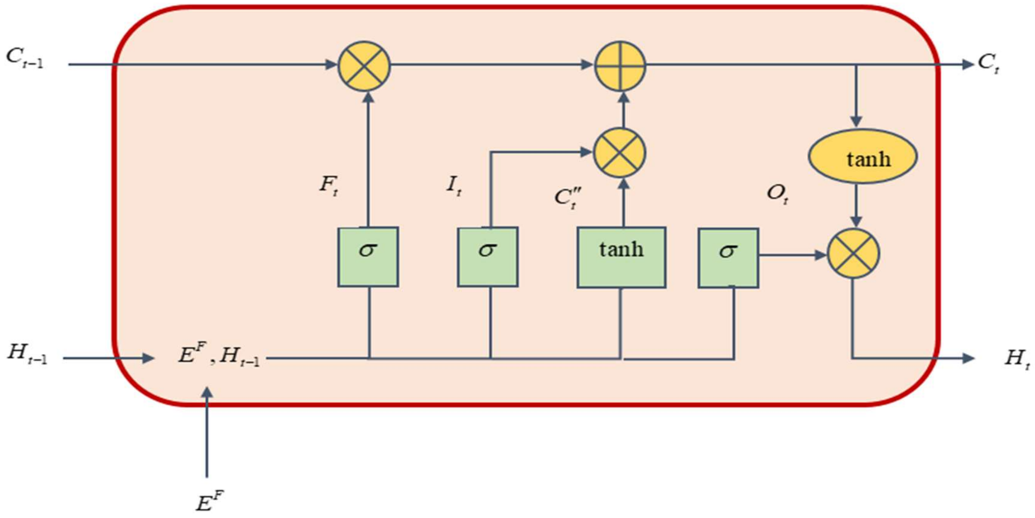


Figure 4: Improved LSTM architecture

(i) Improved Score Alignment

The alignment of the scores is indicated using Eq.(20). According to the proposed model, an improved LSTM is proposed based on the improvement in the attention layer mechanism. Improving an alignment or weighting procedure to represent the significance of specific components more accurately within a sequence is commonly referred to as "more effective score alignment." To accurately align the input sequence with the context vector, this can be accomplished by changing the scoring system or adding improvements to the attention mechanism, which is expressed in Eq.(21). Here, SA_t indicates the improved score alignment mechanism.

$$SA_t = \tanh\left[\left(\hat{H}_t \times P_{att}\right) + Q_{att}\right] \tag{20}$$

$$SA_t = \frac{\tanh\left[\left(\hat{H}_t \times P_{att}\right) + Q_{att}\right]}{\sqrt{\left(P_{att} * Q_{att}\right)}} \tag{21}$$

(ii) Improved Weights

The trainable weights and bias of the attention layer are denoted by P_{att} and Q_{att}, accordingly. Eq. (22) is used to calculate the attention weights W_t by passing the scores SA_t via the Softmax function. To improve the model's representation of features abilities as well as the learning mechanism, improved weights mechanism is proposed in this paper as per Eq.(23). Here, W_t indicates the improved weight mechanism. It is the combination of swish, Softmax and tanh function, SA_t indicates the improved score alignment, which is calculated as per Eq. (21).

$$W_t = Soft\max\left(SA_t\right) \tag{22}$$

$$W_t = Swish\left(Soft\max\left(\tanh\left(SA_t\right)\right)\right) \tag{23}$$

(iii) Improved Context Vector

The context vector, which is equal to the weighted total of T hidden states, is calculated following attention weights calculation, and is given by the notation attention vector in Eq. (24). Eq. (25) shows an improved context vectors by including positional encodings, the attention mechanism can better consider ordered sequences and dependencies while developing the context vector by giving the representation details. It is regarding the comparative or actual placements of the elements inside the input sequence. Here, CV_t' denotes the improved context vector.

$$CV_t = \sum_{i=1}^{T} W_t \hat{H}_t \tag{24}$$

$$CV_t' = \sum_{i=1}^{T} \frac{W_t \hat{H}_t}{\exp\left(\hat{H}_t / 2\right)} \tag{25}$$

A fully connected NN for feature learning receives the attention layer's output and consists of one layer triggered by the function that activates ReLU, and then a regularization term (Dropout at 0.5).

In the improved LSTM model, the loss function is replaced by hybrid combination of loss such as balanced cross entropy and Dice loss. Enhancing the loss function of LSTM networks can result in improved learning, improved generalizing, and improved predictive accuracy in time-series data estimation and sequencing prediction.

The aim of balanced entropy loss functions is to reduce the impact of unequal class distribution on learning, hence addressing the problem of class imbalance in the data. A weighting technique that takes into consideration the class frequencies is incorporated into the normal cross-entropy loss to create the balanced entropy loss function. It is represented as per Eq. (26). Where, $\beta = \frac{1}{2}$, y indicates the actual value and $\hat{y}$ indicates the predicted value.

$$L_{BLE}\left(y,\hat{y}\right) = -\left[\left(\beta * y\log\left(\hat{y}\right)\right) + \left(\left(1-\beta\right) * \left(1-y\right)\log\left(1-\hat{y}\right)\right)\right] \tag{26}$$

By evaluating the degree of similarity or difference among actual patterns and predicted patterns, the Dice loss function in LSTM networks is developed with the objective of reducing the difference among both of them. According to the task and type of data being handled, several Dice lost function formulations and implementations in LSTM networks for data processing could be used. It supports the assessment and improvement of the model's efficacy in preserving sequential relationships and patterns in the data, which is denoted in Eq.(27).

$$L_D = 1 - \frac{2y\hat{y}+1}{y+\hat{y}+1} \tag{27}$$

Proposed Hybrid Loss calculation: For the hybrid loss function, Eq.(26) and Eq,(27) are averaged together to determine the final loss function in the improved LSTM, which is represented in Eq.(28), where, $L_{BLED}\left(y,\hat{y}\right)$ represents the hybrid loss function.

$$L_{BLED}\left(y,\hat{y}\right) = \frac{\left\{\begin{matrix}\left(-\left[\left(\beta * y\log\left(\hat{y}\right)\right) + \left(\left(1-\beta\right) * \left(1-y\right)\log\left(1-\hat{y}\right)\right)\right]\right) + \\ \left(L_D = 1 - \frac{2y\hat{y}+1}{y+\hat{y}+1}\right)\end{matrix}\right\}}{2} \tag{28}$$

3.3.2 Bi-GRU model

Sequential data can be processed using RNNs. When working with the present data, RNNs can also pick up some information from earlier data. Both GRU and LSTM are enhanced RNN models with strong modelling skills for long-term dependencies, GRU is a simpler version than LSTM. Reset and update gates, respectively, make up a GRU unit [39]. Both gates manage their output h'_t, which is regulated by the prior state h'_{t-1} and the present input E^F. The following Eq.(29) is used to determine the gate and GRU unit outputs. Where, Q_r, Q_z, Q_H represents the synthesis of bias vectors for input E^F and the previous state h'_{t-1}, P_r, V_r, P_z, V_z, P_h, $V_{h'}$ indicates the weight matrices, $\tanh$ indicates the hyperbolic activation function and σ indicates the logistic sigmoid function.

$$\begin{aligned} GR_t &= \sigma\left(P_r E^F + V_r h'_{t-1} + Q_r\right) \\ GZ_t &= \sigma\left(P_z E^F + V_z h'_{t-1} + Q_z\right) \\ \overline{h'_t} &= \tanh\left[P_{h'} E^F + V_{h'}\left(GR_t \square\ h'_{t-1}\right) + Q_{h'}\right] \\ h'_t &= \left(1 - GZ_t\right) \square\ h'_{t-1} + GZ_t \square\ \overline{h_t} \end{aligned} \tag{29}$$

When utilizing the present data, models with a bi-directional structure are able to gain from both prior and subsequent data. Figure 5 displays the Bi-GRU model [39] diagram's structure. Two GRUs that are unidirectional in opposing directions are used for establishing the bi-GRU model. Two GRUs: one travels forward and starts at the starting point of a data series, while the other moves backward and starts at the end of the data sequence with 64 units. This enables knowledge from the past and the future to affect the states that are in place currently. Linear activation function is used to produce a linear combination of weights and inputs, which is obtained from the output of Bi-GRU layer Eq. (29) shows the definition of the Bui-GRU. Where, $\oplus$ represents the operations of the concatenating two vectors., $\vec{h}_t'$ indicates the forward state GRU and $\overleftarrow{h}_t'$ indicates the negative state GRU.

$$\begin{aligned}&\vec{h}_t' = GRU^{fwd}\left(E^F, \vec{h}_{t-1}'\right)\\&\overleftarrow{h}_t' = GRU^{bwd}\left(E^F, \overleftarrow{h}_{t-1}'\right)\\&h_t' = \vec{h}_t' \oplus \overleftarrow{h}_t'\end{aligned} \qquad (30)$$

Final Prediction outcome: Thus, the final prediction results will be determined by averaging the output of the improved LSTM and Bi-GRU, which will be the final prediction outcome of future location.

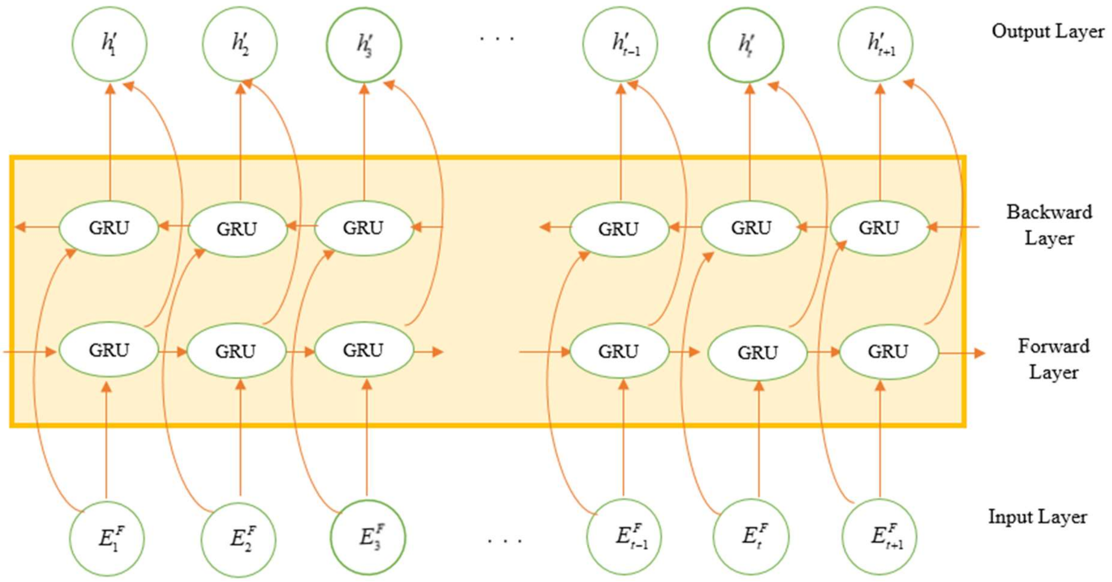


Figure 5: Architecture of Bi-GRU

3.4 Tracking using Yolov7 model

In this work, Yolov7 [40] is used for the tracking the location. Yolo is termed as the ability to track location quickly and effectively, particularly in situations where real-time processing is essential. YOLO utilizes a single NN's structure to split the image into several layouts and forecast numerous bounding boxes (each including class data as well as object positions) for each grid. As a result, YOLO forecasts four coordinates' variables for each bounding box, which stand for the locations of the upper-left and lower-right surrounding corners as well as the likelihood that every value belongs to a particular category. This prediction process can be categorized as a regression issue because it requires calculating regression among the parameters of the model and the input data.

YOLO has introduced several versions, each with unique enhancements and adjustments. In this paper YOLOv7 is used, which is regarded as the most stable and dependable of the versions that have been made public. YOLOv7 is greatly outperformed than the previous YOLO models in terms of both detection accuracy and speed. Expanding on its predecessor, YOLOv7 creatively presents the expanded ELAN architecture, which enhances the network's capacity for self-learning without erasing the gradient route. ELAN maximizes the gradient length of the entire network using the stack structure in the computing block and is mostly made up of VoVNet and CSPNet combined. Deeper networks can efficiently train and converge by optimizing gradient trajectories. Furthermore, YOLOv7 has a cascade-based model scaling technique that dynamically modifies the model size to meet the particular needs for detecting the location. Model scaling serves the primary function of modifying some model features and producing models at various scales to accommodate varying inference rates. For instance, breadth, depth, and resolution are taken into account in the EfficientDet scaling model.

Two essential components are integrated into the YOLOv7 input component namely, adaptive scaling and adaptive anchor box. The main purpose of adaptive scaling is to change the input image's size. This strategy has a number of benefits. First off, when working with large-sized images, it can be stored in memory. Second, it improves the model's generalization capabilities by allowing the network to adjust to input images of different sizes. Lastly, by guaranteeing that small items take up a larger percentage of the image, adaptive scaling helps to increase tracking localization and detection accuracy. The adaptive anchor box is in-charge of choosing the quantity and dimensions of earlier boxes automatically. The adaptive anchor box improves trajectory location detection accuracy during testing by adapting to various object scales and aspect ratios. Additionally, it provides adaptability in handling various situations and assignments by employing the K-means method. The model's capacity to adapt to varying item scales and aspect ratios enhances its overall effectiveness across a range of detecting circumstances.

Three modules make up the core of YOLOv7, which is in charge of feature extraction such as MP, ELAN and CBS. The Conv, BN and SiLU layers are among the layers that make up the CBS module. It can collect features at different scales since it uses three distinct convolutional kernel sizes and step sizes. In order to extract useful characteristics from the input data, the CBS module is essential. The ELAN module is a productive network architecture that manages the network's longest and shortest gradient pathways. This improves the resilience of the network by pushing it to learn more varied and discriminative characteristics.

4.0 Results and Discussion

4.1 Experimental Setup

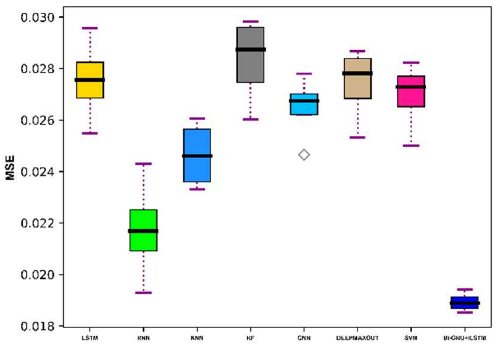
The hybrid (Bi-GRU + ILSTM) model is developed by using PYTHON 3.7.9 on an Intel i5 PC with 16GB RAM. Bi-GRU + ILSTM model is successfully developed and verified by comparing conventional techniques such as RNN, KNN, RF, DEEPMAXOUT, SVM, CNN [26], LSTM [27]. Datasets used for training and testing of Human Trajectory prediction and tracking is MOT17Det.

4.2 Dataset Description

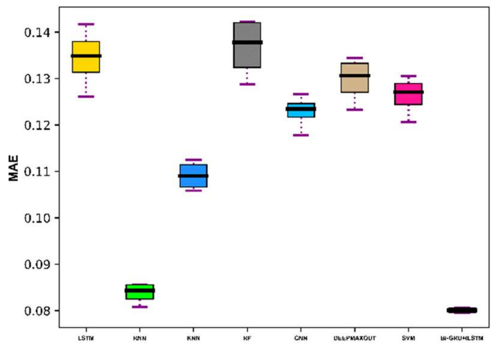
The Multiple Object Tracking Benchmark provides MOT17Det [41]. This approach was developed for the objective assessment of multi-person monitoring methods. There are a lot of datasets in the collection, some of which are now in use and some of which are brand-new, challenging sequences. Every sequence in the dataset has already been located. It also contains an assessment tool that offers several metrics, including recall, precision, and running time. A simple way for assessing the efficacy of modern tracking approaches is provided in the dataset. A training set and a testing set are both included in the dataset. The resolution of the entire video sequence is 1920x1080.

4.3 Comparative Analysis

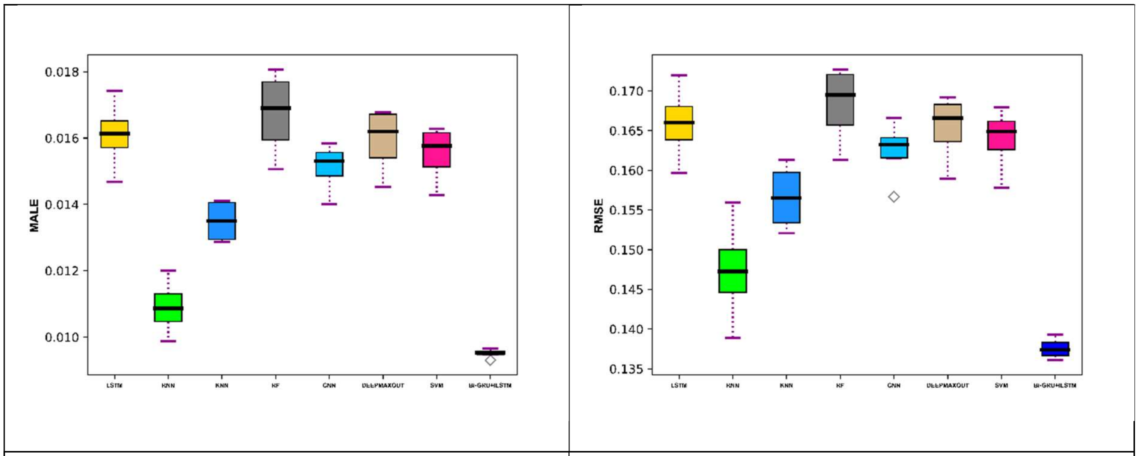
We analyse the Bi-GRU + ILSTM model to evaluate its performance with least prediction error than the existing models. If the prediction error reduces, the human trajectory estimate will be more accurate. The Bi-GRU + ILSTM model for Human Trajectory Prediction is examined using the following error metrics: MSE, MAE, MALE, RMSE, MPL and ME, and FDE. Figure 6 shows the error value occurred by the Bi-GRU + ILSTM model and conventional techniques for the MSE, MAE, MALE, RMSE. To investigate the Bi-GRU + ILSTM model's error rate, a comparison is made between the proposed model and many conventional approaches, such as RNN, KNN, RF, DEEPMAXOUT, SVM, CNN, LSTM. As shown in Figure 6, MSE is 0.019, MAE is 0.080, MALE is 0.009, RMSE is 0.139 for the Bi-GRU + ILSTM model at learning percentage 90 indicates that it has occurred less error than other traditional procedures. At learning percentage 60 for MAE, the Bi-GRU + ILSTM model has 0.079 errors, while traditional optimization algorithms like the RNN, KNN, RF, DEEPMAXOUT, SVM, CNN, LSTM all have errors of 0.085, 0.105, 0.141, 0.132, 0.128, 0.123 and 0.126, respectively. Figure 6(d) shows that at a learning percentage of 80, the Bi-GRU + ILSTM model has a less significant 0.009 error for MALE, whereas other conventional techniques have higher MALE error value.



(a)



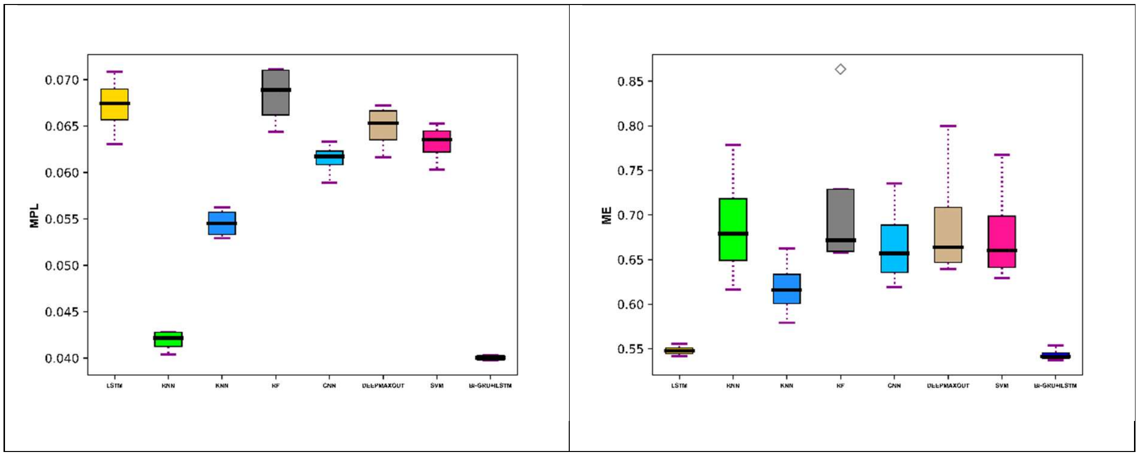
(b)



(c) (d)

Figure 6: Illustration of the Bi-GRU + ILSTM model's error analysis in relation to traditional methods for (a) MSE (b) MAE (c) MALE (d) RMSE

In Figure 7, the error analysis for the Bi-GRU + ILSTM model using the Bi-GRU and Improved Long Short-Term Memory is graphically shown and contrasted with the MPL, ME, and FDE schemes. The Bi-GRU + ILSTM model obtains less error in the following learning percentages: 60, 70, 80, and 90, in all error detection metrics. The Bi-GRU + ILSTM model's feature extraction phase enables the model to extract Improved Local Ternary Pattern (ILTP) based features, Improved semantic features from image characteristics. This aids in the model's conversion of low-level data into high-level semantics, lowering model error and increasing the likelihood of a human trajectory prediction and tracking over that of other traditional methods. Fig 7(a) illustrates the MPL error value at learning percentage 90, which is 0.040. The inaccuracy of other methods is approximately 2% more than that of the Bi-GRU + ILSTM model. The Bi-GRU + ILSTM model has a lower FDE scheme than the pre-existing methods. At the learning percentage of 90, the FDE value of the Bi-GRU + ILSTM model is 4.50. The error incidence for the Bi-GRU + ILSTM model for human trajectory prediction and tracking decreases as the learning percentage increases from 60 to 90. Also, the better performance is assured with the incorporation of improved filtering process that enhances the image quality. Hence, proved the performance of proposed work over the conventional models.



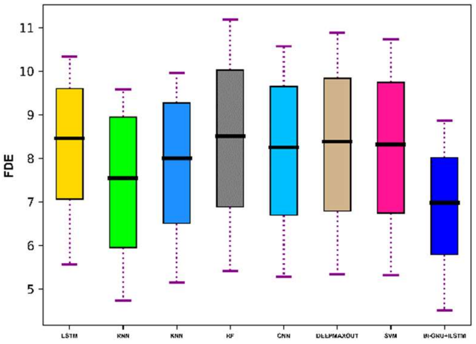


Figure 7: Illustration of the Bi-GRU + ILSTM model's error analysis in relation to traditional methods for (a) MSE (b) MAE (c) MALE (d) RMSE

Also, the proposed Bi-GRU + ILSTM model is contrasted with the recent models like hybrid model + BMCOA, Hybrid model + SU-TSO. Table 2 unequivocally demonstrates that the Bi-GRU + ILSTM model performs better than the Hybrid model+BMCOA and Hybrid model +SU-TSO. The Bi-GRU + ILSTM technique has all the error metrics at a low level; the Bi-GRU + ILSTM model's MSE is 0.019, which is less than the hybrid model + BMCOA and Hybrid model +SU-TSO. While the MALE value of the Bi-GRU + ILSTM model is 0.009, that of the Hybrid model + BMCOA is 0.067 and Hybrid model + SU-TSO is 0.017. The Bi-GRU + ILSTM model yields a 7.73 FDE, while the previous models obtain 7% higher error rate. The enhancement made in the Bi-GRU + ILSTM model makes the model to predict and track human trajectory more accurately with lesser error.

Table 2: Comparative Analysis of Hybrid model+BMCOA, Hybrid model +SU-TSO and Bi-GRU + ILSTM approach

	Hybrid model+BMCOA	Hybrid model +SU-TSO	Bi-GRU + ILSTM
MSE	0.133427	0.033601	0.019032
MAE	0.259441	0.111325	0.080654
MALE	0.067262	0.017532	0.009531
RMSE	0.365277	0.153305	0.137956
MPL	0.12972	0.055662	0.040327
ME	0.969471	0.843769	0.542725
FDE	15.21318	14.05892	7.731894

4.4 Performance Analysis

The performance of Bi-GRU + ILSTM work is evaluated in terms of ablation study, which is given in subsequent section. Moreover, Figure 8 displays the image results of the Bi-GRU + ILSTM work. Figure 9 displays the Tracked image.

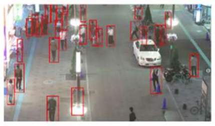
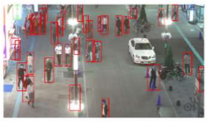
Original	Detected Image
	
	

Fig 8: Image results (a) original image, (b) Detected image

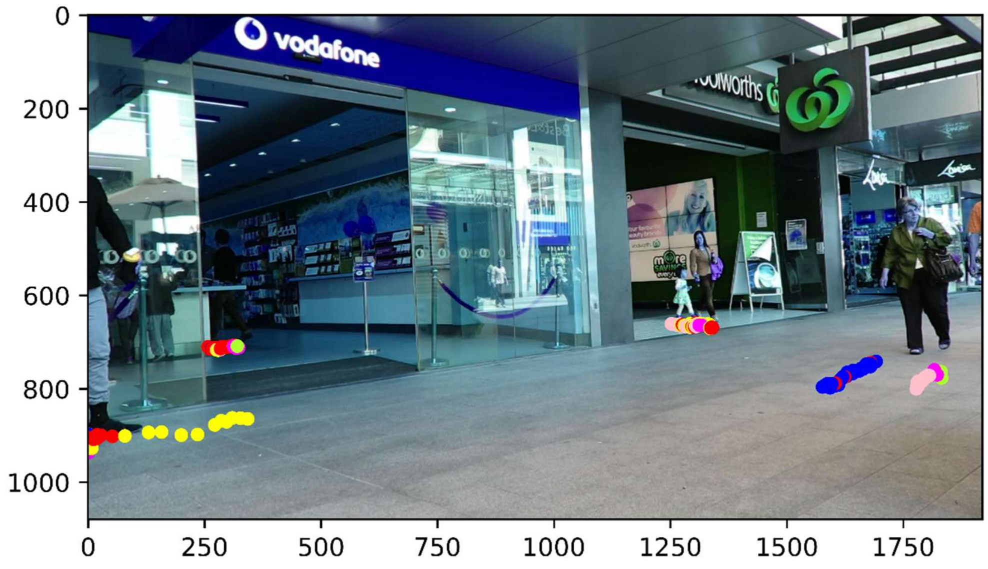


Figure 9: Image result of Tracking

4.4.1 Ablation Study

Table 3 presents the ablation study of the Bi-GRU + ILSTM model. The Bi-GRU + ILSTM approach is compared with the model without feature extraction, model with conventional sematic and with conventional filtering. The Bi-GRU + ILSTM model's error incidence is assessed using the following metrics: MSE, MAE, MALE, RMSE, MPL, ME, FDE. The Bi-GRU + ILSTM model's MSE occurrence rate is 0.019, meaning that errors larger than 0.02 are present in the models with regular filter, regular sematic feature and model without feature extraction. The Bi-GRU + ILSTM model achieved a RMSE value of 0.139, indicating that it performs better than the model using conventional techniques. The model without feature extraction obtained a MAE value of 0.135. Improved Local Ternary Pattern (ILTP) based features, improved semantic features are retrieved in the Bi-GRU + ILSTM model, which enables the model to lower the prediction error of the human trajectory and accurately track the human trajectory. The MALE of the conventional filtering model and the model with conventional sematic is approximately 0.017, whereas the MALE of the Bi-GRU + ILSTM model is less than 0.009. The FDE value of the Bi-GRU + ILSTM model is 4.50, but the errors of the model with other approaches are 5% higher than that of the Bi-GRU + ILSTM model. According to the ablation study, the Bi-GRU + ILSTM model outperforms pre-existing approaches for predicting and tracking human trajectories.

Table 3: Ablation study of Bi-GRU + ILSTM approach

	Proposed without feature extraction	Proposed with conventional sematic	Proposed with conventional filter	BI-GRU + ILSTM
MSE	0.027452	0.028828	0.029045	0.019414
MAE	0.135586	0.138276	0.139603	0.080516

MALE	0.016169	0.017055	0.017386	0.009646
RMSE	0.165687	0.169789	0.170427	0.139333
MPL	0.067793	0.069138	0.069801	0.040258
ME	0.541736	0.544192	0.542119	0.539815
FDE	9.74343	9.988387	10.02235	4.508007

4.5 Statistical Analysis

The Bi-GRU + ILSTM model and conventional techniques are evaluated in terms of statistical analysis on the basis of error occurrence. Statistical analysis of the Bi-GRU + ILSTM and traditional techniques is shown in Table 4. Mean, Median, Standard deviation, minimum and maximum level of error occurrence are evaluated from the Bi-GRU + ILSTM and traditional models. Comparing the average error occurrence of traditional methods and Bi-GRU + ILSTM model, proposed model has the lowest error occurrence possibility as the mean value of Bi-GRU + ILSTM is 0.018 and other traditional techniques error occurrence is above 0.02. The minimum error occurrence for Bi-GRU + ILSTM model is 0.018 and maximum is 0.019, where the traditional techniques ranges from 0.019 to 0.029. Standard deviation of the Bi-GRU + ILSTM model is 0.0003, which is lowest of all other techniques. Thus, from the statistical analysis proves that the Bi-GRU + ILSTM model performs well for predicting and tracking of human trajectory.

Table 4: Statistical Analysis of Bi-GRU + ILSTM approach to previous methods

	LSTM	RNN	KNN	RF	CNN	DEEPMAX-OUT	SVM	BI-GRU + ILSTM
Mean	0.027539	0.021742	0.02464	0.028328	0.026484	0.027406	0.026945	0.018923
Median	0.027557	0.021686	0.024603	0.028738	0.026744	0.027814	0.027279	0.018881
Std	0.001453	0.001782	0.001169	0.001518	0.001141	0.001292	0.001205	0.000337
Min	0.025481	0.019289	0.023297	0.02601	0.024654	0.025332	0.024993	0.018517
Max	0.02956	0.024307	0.026058	0.029825	0.027795	0.028664	0.02823	0.019414

5.0 Conclusion

The paper enhances the existing efficacy of RNN methods for sequential task prediction by providing a hybrid deep learning model that may be utilized to learn fundamental human movement patterns and forecast future trajectories. The future position and tracking method of the human trajectory prediction in this case involves three steps: Preprocessing, feature extraction and future location prediction. Preprocessing enable the incoming video from the dataset to be transformed into frames or images, which were then filtered employing an improved wiener filtering procedure to improve the image quality. The preprocessed image was then used to extract features based on Texton, Resnet, VGG 16, improved semantic features, and improved LTP features. Next, the model's last phase—future location prediction—proposes a hybrid deep learning architecture that combines an improved LSTM with a Bi-GRU model. Consequently, the Yolo V7 technique is employed for tracking purposes and the proposed hybrid algorithm was trained to predict future location. In terms of error measurements, experiments conducted on a publicly accessible dataset show that this technique performs better than other existing technologies.

References:

[1] P. Kothari and A. Alahi, "Safety-Compliant Generative Adversarial Networks for Human Trajectory Forecasting", IEEE Transactions on Intelligent Transportation Systems, vol. 24, no. 4, pp. 4251-4261, April 2023, doi: 10.1109/TITS.2022.3233906.

[2] Zhiquan He, Hao Sun, Wenming Cao & Henry Z. He, "Multi-level context-driven interaction modeling for human future trajectory prediction", Neural Computing and Applications (2022) Volume 34, pp: 20101–20115, doi : 10.1007/s00521-022-07562-1

[3] Yanyan Fang, Zhiyu Jin, Zhenhua Cui, Qiaowen Yang, Tianyi Xie & Bo Hu, "Modeling human–human interaction with attention-based high-order GCN for trajectory prediction", The Visual Computer volume 38, pages 2257–2269 (2022), doi : 10.1007/s00371-021-02109-2

[4] Pei Lv, Hui Wei, Tianxin Gu, Yuzhen Zhang, Xiaoheng Jiang, Bing Zhou & Mingliang Xu, "Trajectory distributions: A new description of movement for trajectory prediction", Computational Visual Media volume: 8, PP: 213–224 (2022), doi : 10.1007/s41095-021-0236-6

[5] Ronny Hug, Stefan Becker, Wolfgang Hübner and Michael Arens, "Quantifying the Complexity of Standard Benchmarking Datasets for Long-Term Human Trajectory Prediction", IEEE Access, vol. 9, pp. 77693-77704, 2021, doi: 10.1109/ACCESS.2021.3082904.

[6] Akif Hacinecipoglu, E. Ilhan Konukseven & A. Bugra Koku, "Multiple human trajectory prediction and cooperative navigation modeling in crowded scenes", Intelligent Service Robotics volume 13, pages 479–493 (2020) doi : 10.1007/s11370-020-00333-8

[7] Pedro A. Peña & Ubbo Visser, "ITP: Inverse Trajectory Planning for Human Pose Prediction", KI - Künstliche Intelligenz volume 34, pages209–225 (2020), doi : 10.1007/s13218-020-00658-7

[8] Quan T. Ngo, Doi Thi Lan, Seokhoon Yoon, Woo-Sung Jung, Taehyun Yoon and Daeseung Yoo, "Companion Mobility to Assist in Future Human Location Prediction," in IEEE Access, vol. 10, pp. 68111-68125, 2022, doi: 10.1109/ACCESS.2022.3186319.

[9] Brandon Victor, Aiden Nibali, Zhen He and David L. Carey, "Enhancing trajectory prediction using sparse outputs: application to team sports", Neural Computing and Applications volume 33, pages11951–11962 (2021), doi : 10.1007/s00521-021-05888-w

[10] Hao Li, Haiyan Kang, "Research on User Behavior Prediction and Profiling Method Based on Trajectory Information", Automatic Control and Computer Sciences, Volume 54, pages: 456–465 (2020), doi : 10.3103/S0146411620050065

[11] Q. Li, Z. Zhang, Y. You, Y. Mu and C. Feng, "Data Driven Models for Human Motion Prediction in Human-Robot Collaboration", in IEEE Access, vol. 8, pp. 227690-227702, 2020, doi: 10.1109/ACCESS.2020.3045994.

[12] Hee-Seung Moon and Jiwon Seo, "Sample-Efficient Training of Robotic Guide Using Human Path Prediction Network," in IEEE Access, vol. 10, pp. 104996-105007, 2022, doi: 10.1109/ACCESS.2022.3210932.

[13] Junyao Guo, Sihai Zhang, Jinkang Zhu and Rui Ni, "Measuring the Gap Between the Maximum Predictability and Prediction Accuracy of Human Mobility", IEEE Access, vol. 8, pp. 131859-131869, 2020, doi: 10.1109/ACCESS.2020.3009868.

[14] Zhongshuai Wang, Yuan Yuan, Liang Chang, Xiyan Sun and Xiaonan Luo, "A Graph-Based Visual Query Method for Massive Human Trajectory Data", IEEE Access, vol. 7, pp. 160879-160888, 2019, doi: 10.1109/ACCESS.2019.2948304.

[15] J. Zhong, H. Sun, W. Cao and Z. He, "Pedestrian Motion Trajectory Prediction With Stereo-Based 3D Deep Pose Estimation and Trajectory Learning," in IEEE Access, vol. 8, pp. 23480-23486, 2020, doi: 10.1109/ACCESS.2020.2969994.

[16] Yue Yu, Yepeng Yao, Zhewei Liu, Zhenlin An, Biyu Chen, Liang Chen, Ruizhi Chen, "A Bi-LSTM approach for modelling movement uncertainty of crowdsourced human trajectories under complex urban environments". International Journal of Applied Earth Observation and Geoinformation, Vol. 122, August 2023, 103412, https://doi.org/10.1016/j.jag.2023.103412

[17] X. Zhao, Y. Chen, J. Guo and D. Zhao, "A spatial-temporal attention model for human trajectory prediction," in IEEE/CAA Journal of Automatica Sinica, vol. 7, no. 4, pp. 965-974, July 2020, doi: 10.1109/JAS.2020.1003228.

[18] C. Ding et al., "Continuous Human Motion Recognition With a Dynamic Range-Doppler Trajectory Method Based on FMCW Radar," in IEEE Transactions on Geoscience and Remote Sensing, vol. 57, no. 9, pp. 6821-6831, Sept. 2019, doi: 10.1109/TGRS.2019.2908758.

[19] Ki-In Na, Ue-Hwan Kim, Jong-Hwan Kim, "SPU-BERT: Faster human multi-trajectory prediction from socio-physical understanding of BERT", Knowledge-Based Systems, Volume 274, 15 August 2023, 110637, doi : 10.1016/j.knosys.2023.110637

[20] Fan, Z., Song, X., Chen, Q. et al. Trajectory fingerprint: one-shot human trajectory identification using Siamese network. CCF Trans. Pervasive Comp. Interact. 2, 113–125 (2020). https://doi.org/10.1007/s42486-020-00034-2

[21] Qianhui Men and Hubert P.H. Shum, "PyTorch-based implementation of label-aware graph representation for multi-class trajectory prediction", Software Impacts, Volume 11, February 2022, 100201, doi : 10.1016/j.simpa.2021.100201

[22] Xingchen Zhang, Panagiotis Angeloudis and Yiannis Demiris, "Dual-branch spatio-temporal graph neural networks for pedestrian trajectory prediction", Pattern Recognition, Volume 142, October 2023, 109633, doi : 10.1016/j.patcog.2023.109633

[23] Han Bao, Xun Zhou, Cara Hamann and Steven Spears. "Understanding children's cycling route selection through spatial trajectory data mining", Transportation Research Interdisciplinary Perspectives, Volume 20, July 2023, 100855, doi : 10.1016/j.trip.2023.100855

[24] Alessia Bertugli, Simone Calderara, Pasquale Coscia, Lamberto Ballan and Rita Cucchiara, "AC-VRNN: Attentive Conditional-VRNN for multi-future trajectory prediction", Computer Vision and Image Understanding, Volume 210, September 2021, 103245, doi : 10.1016/j.cviu.2021.103245

[25] Qinghua Li, Lei Zhang; Mengyao Zhang, Yuanshuai Du, Kaiyue Liu and Chao Feng, "Robust Human Upper-Limbs Trajectory Prediction Based on Gaussian Mixture Prediction," in IEEE Access, vol. 11, pp. 8172-8184, 2023, doi: 10.1109/ACCESS.2023.3239009.

[26] Dapeng Zhao and Jean Oh, "Noticing Motion Patterns: A Temporal CNN With a Novel Convolution Operator for Human Trajectory Prediction", IEEE ROBOTICS AND AUTOMATION LETTERS, VOL. 6, NO. 2, APRIL 2021, doi : 10.1109/LRA.2020.3047771

[27] Dongho Choi, Janghyuk Yim, Minjin Baek and Sangsun Lee, "Machine Learning-Based Vehicle Trajectory Prediction Using V2V Communications and On-Board Sensors", Electronics 2021, 10, 420, doi : 10.3390/electronics10040420

[28] Kalaivani K and Asnath Victy Phamila Y, "Modifed Wiener Filter for Restoring Landsat Images in Remote Sensing Applications", Pertanika Journal of Science and Technology 26 (3), pp:1005 - 1018, 2018

[29] Xu Qun and Zhou Jian, "Improved SNR Estimation Algorithm", 2017 International Conference on Computer Systems, Electronics and Control, 2017

[30] Hao Chen, Zhaofeng Cen, Chuanchuan Wang, Shun Lan and Xiaotong Li, "Image Restoration via Improved Wiener Filter Applied to Optical Sparse Aperture Systems", Optik, Volume 147, October 2017, Pages 350-359, doi : 10.1016/j.ijleo.2017.08.102

[31] Xiu Liu and Chris Aldrich, "Deep Learning Approaches to Image Texture Analysis in Material Processing", Metals 2022, 12, 355, doi: 10.3390/met12020355

[32] Wen Xin Cheng, P.N. Suganthan and Rakesh Katuwal, "Time series classification using diversified Ensemble Deep Random Vector Functional Link and Resnet features", Applied Soft Computing, volume 112, (2021), 107826, doi : 10.1016/j.asoc.2021.107826

[33] D. Albashish, R. Al-Sayyed, A. Abdullah, M. H. Ryalat and N. Ahmad Almansour, "Deep CNN Model based on VGG16 for Breast Cancer Classification", 2021 International Conference on Information Technology (ICIT), Amman, Jordan, 2021, pp. 805-810, doi: 10.1109/ICIT52682.2021.9491631.

[34] Zaifeng Shi, Hui Li, Qingjie Cao, Huizheng Ren & Boyu Fan, "An Image Mosaic Method Based on Convolutional Neural Network Semantic Features Extraction", Journal of Signal Processing Systems, 12 September 2019, doi : 10.1007/s11265-019-01477-2

[35] A. Ferreyra-Ramirez, C. Aviles-Cruz, E. Rodriguez-Martinez(B), J. Villegas-Cortez, and A. Zu˜niga-Lopez, " An Improved Convolutional Neural Network Architecture for Image Classification", doi: 10.1007/978-3-030-21077-9_9

[36] Chigozie Enyinna Nwankpa, Winifred Ijomah, Anthony Gachagan, and Stephen Marshall", Activation Functions: Comparison of Trends in Practice and Research for Deep Learning", Machine Learning, 2018, doi :10.48550/arXiv.1811.03378

[37] Wankou Yang, Zhenyu Wang and Baochang Zhang, "Face recognition using adaptive local ternary patterns method", Neurocomputing, Volume 213, 12 November 2016, Pages 183-190, doi : 10.1016/j.neucom.2015.11.134

[38] Skander Hamdi, Mourad Oussalah, Abdelouahab Moussaoui, Mohamed Saidi, "Attention-based hybrid CNN-LSTM and spectral data augmentation for COVID-19 diagnosis from cough sound", Journal of Intelligent Information Systems (2022) 59:367–389, doi : 10.1007/s10844-022-00707-7

[39] XINYU LIU, YONGJUN WANG, XISHUO WANG, HUI XU, CHAO LI AND XIANGJUN XIN, "Bi-directional gated recurrent unit neural network based nonlinear equalizer for coherent optical communication system", Optics Express, Vol. 29, No. 4, 15 February 2021, pp: 5923-5933, doi : 10.1364/OE.416672

[40] Kai Zhao, Lulu Zhao, Yanan Zhao and Hanbing Deng, "Study on Lightweight Model of Maize Seedling Object Detection Based on YOLOv7", Applied science 2023, 13, 7731, doi : 10.3390/app13137731

[41] https://motchallenge.net/data/MOT17Det/

Nomenclature

Abbreviation	Description
AI	Artificial Intelligence
Bi-GRU	Bi-Directional Gated Recurrent Unit
BN	Batch Normalization
CBS	Conv-BN-Silu
CNN	Convolutional Neural Network
Conv	Convolutional
DCNN	Deep Convolutional Neural Network
DL	Deep Learning
ELAN	Extended Latent Attention Network
GAT	Graph Attention Network
GCN	Graph Convolutional Network
GRU	Gated Recurrent Unit
ILTP	Improved Local Ternary Pattern
ILSTM	Improved LSTM
ITP-IRL	Inverse Trajectory Planning – Inverse Reinforcement Learning Algorithm
LBP	Local Binary Pattern
LSTM	Long Short-Term Memory
LTP	Local Ternary Pattern
LVQ	Learning Vector Quantization
MCDIM	Multi-Level Context-Driven Interaction Modeling
ML	Machine Learning
MP	Max- Pooling
NN	Neural Network

PIE	**Person Id Embedding**
POI	**Person Of Interest**
PSNR	**Peal Signal To Noise Ratio**
ReLU	**Rectified Linear Unit**
ResNet	**Residual Network**
RF	**Random Forest**
RNN	**Recurrent Neural Network**
SC	**Spatial Closeness**
SGAN	**Social Generative Adversarial Networks**
SiLU	**Sigmoid Linear Unit**
SNR	**Signal To Noise Ratio**
Social-PEC	**Social Pattern Extraction Convolution**
TL	**Transfer Learning**
VGG	**Visual Graphics Group**