Predicting Human Movement in Crowds

A Dissertation
Presented in Partial Fulfillment of
the Requirements for the Degree of

Doctor of Philosophy

with a Major in

Computer Science

in the

College of Graduate Studies

University of Idaho

by

Abdullah Alajlan

Approved by:

Major Professor
Terence Soule, Ph.D.

Committee
Frederick Sheldon, Ph.D.
Marshall Ma, Ph.D.
Yacine Chakhchoukh, Ph.D.

Department Administrator
Terence Soule, Ph.D.

May 2022

Abstract

The prediction of human movement when people gather in crowds for reasons has become very important for public safety and the protection of property. From the early 1990s different techniques have been studied to predict the next steps of individuals in crowds and the field of study has increased in rapidly as a result. Our research has developed along three lines of inquiry. First, we developed the use of a combination of genetic algorithms and neural networks (GA-NN) to predict individuals' future steps in crowded areas. We applied a method, using a cone of vision of individuals to specify the location of the nearest people, in order to train the neural networks to accurately predict the decisions the individual agents would make based on their nearest neighbors. We demonstrated that using this combination of genetic algorithms and neural networks is effective at predicting movement in crowds. We also demonstrated that different physical layouts of areas and the difference in crowd types give different results when compared across experiments. Our crowd types included a structured crowd area and tan unstructured crowd area. We also introduced a new metric, the cumulative distance error (CDE) that is very effective in measuring prediction accuracy and can be used to improve experimentation in the field of human movement prediction.

Second, we tested the use of the long short-term memory (LSTM) using the similar methods that were used for the cone of vision calculations. It is known that recurrent neural networks (RNNs) and its sub-type LSTMs make use of memory and the past for training; we took the directions of the agent's recent past steps, and the directions of other pedestrians in the field of view and predicted a sequence of future directions by taking the average speed as a constant speed in the future. We used three different layout styles, including structured crowds, unstructured crowds, and merging paths. We also used two different behaviors: the flocking model (FM) and the social force model (SFM). We compared our results with the LSTM method with the method described above that used the combination of genetic algorithms and neural networks.

The results show that the LSTM outperforms the combination of genetic algorithms and neural networks in both simulations, and for every dataset.

In the third line of inquiry, we expanded our work to include more scenarios, more types of crowds, and a longer distance for prediction. For example, intersections, waypoints, and more complicated unstructured crowded areas have been included in this line of inquiry. Additionally, we extended the predictions from five time-steps up to fifty time-steps in certain situations. The results show that we should differentiate between crowd types in predictions because the results depend on the crowd types, and layout styles.

Acknowledgements

All glorification be to God, who has given me the ability to complete this dissertation. This work would not have been done without the assistance and support from many students and professors in this field. I would like to thank every person individually. Special appreciation goes to my professors, without whom I would not have been able to finish my dissertation, and especially to my major advisor, Professor Terence Soule, whose knowledge of the subject matter and patience steered me through this research. Special thanks also goes to Professor Rick Sheldon, whose support and suggestions allowed my research to go beyond my expectations. Thanks goes to the rest of my committee, Professor Marshall (Xiaogang) Ma, and Professor Yacine Chakhchoukh, for their supports, advice, and encouragements.

My sincere thanks go to all those in the Department of Computer Science at the University of Idaho, who have provided a superb environment and such a pleasant working place.

Moreover, I am thankful to the Saudi Arabian Cultural Mission and Technical and Vocational Training Corporation for funding my scholarship. I would also like to give a separate thanks to the Ministry of Hajj; without their kind support and help, this achievement would not have been possible.

Dedication

I dedicate this doctoral dissertation to my parents, my wife, my children, my entire extended family, and the people who have supported me during the challenges of this scientific study.

Contents

List of Tables

List of Figures

CHAPTER 1

Introduction

The long short-term memory (LSTM) is a special kind of the popular recurrent neural network (RNN). An RNN accounts for the disadvantages that the ordinary neural networks have. Because people normally begin to think about a problem or an issue based on something that already exists, the RNN algorithm was developed from the idea of remembering the past when deciding what actions to take in the future. An RNN uses internal states to handle a series of past values of a neural network's variables as inputs. Gradient exploding and gradient vanishing are the problems for the RNN function. These problems were discovered by computer scientists Sepp Hochreiter in 1991, and by Yoshua Bengio in 1994. The information in traditional neural networks goes from the input neurons to the outputs, and in backpropogation training, it is transmitted back to rescale the weights after the calculating the error. In a RNN the situation is different, because the information from the previous timesteps is input for the next timestep, the error can be figured out at each time-step. The error is calculated as the difference between the outputs and the ground truth. At this point, while back propagating to rescale the weights at each time-step, the outputs need to be multiplied by the recurrent learning weight, and the same thing is done at each time-step; the result will diminish very quickly when you multiply these outcomes by a number much less than one. Contrasting, if the outcomes are multiplied by a number greater than one, the results will explode, and an exploding gradients problem will occur. To find a solution for vanishing and exploding gradients, Sepp Hochreiter and Jürgen Schmidhuber suggested the use of LSTMs.

In the field of genetic algorithms and neural networks, the early papers in this area were by Montana *et al*. (1989) and Miller *et al*. (1989), who published their papers in 1989. The idea was to train the neural networks by using genetic algorithms.

The social force model (SFM) for pedestrian movement was suggested by Helbing and Molnar (1995); the idea was to describe pedestrians' movements and how they subordinate to social forces. The three pillars for the social force model can be summarized as follows: acceleration to the target, keeping distance between the pedestrian and constant-dynamic obstacles, and the effect of catchy places.

The flocking model (FM) is another model for describing the behavior of multiple agents in motion. This model was developed by Reynolds (1987). The idea behind the flocking model was that the agents move together as a group and try to avoid any collisions.

The field of prediction has grown with the use of machine learning approaches. Handwriting, text-to-speech, vehicle motion, and pedestrians' trajectories are just a few examples where promising results have been obtained by using machine learning. Predicting humans' movements has become a substantial matter for public safety, and a more paramount issue for the authorities and organizers who deal specifically with crowded areas. The existence of robots among people is an additional case that has raised the importance of predicting humans' movements. Predicting humans' motion in highly dense crowds, such as at the Hajj in Saudi Arabia, or kumbh Mela in India, is beneficial for organizers, for example, to position barriers to manage crowd motion or to avoid barriers to smooth movement.

Tracing people in very dense crowds is a complicated mission. Beginning in 2030, Saudi Arabian authorities plan to require every person who participates in the Hajj to have a watch or wristband tracker Alarabiya (2018). This action will help obtain the actual locations of individuals in such a dense crowd.

In this research, we concentrated on the prediction of individuals within crowds. This subject is important to us because dense crowds can lead to a risk of deaths at huge events, as happened and was reported by the Ministry of Hajj in Saudi Arabia wikipedia (2019). The styles of motion in our simulations were inspired from the real-world crowds. Xue *et al*. (2017) mentioned how the layout of places can play a significant role in the predictions, so we created many designs of areas and more than one behavior of walking agents. The difference in planning areas shows how the prediction is different from condition to condition. We varied our planning or scenarios to include intersection, structured crowded area, more than one scenario for unstructured crowded areas, a roadway that converged into one lane, and waypoints where people stop for a while to do something and then continue on their way.

Crowd scenarios come in many styles, but the main categories in this field are structured crowds, and unstructured crowds. In structured crowded areas people usually go in one direction and they have one goal. In unstructured crowded areas, people move in more than one direction and have many goals.

1.1 PRELIMINARY RESEARCH BACKGROUND

In this section, we concentrate on some problems that have been mentioned in the past literature for pedestrian prediction. Xue *et al*. (2017) researched how the area design could have an impact on pedestrian prediction. Shirazi and Morris (2015) mentioned the influence of making a decision when the agent is alone or with a group, and in proximity to cars. Ridel *et al*. (2018) talked about some problems in existing datasets, such as pre-directed agents, and the limitation of the number of collected datasets. Schmidt and Faerber (2009) explored the sudden stops of individuals, and the variations of speed based on age and gender. A longer section of literature review and background can be found in Chapter 2.

1.2 RESEARCH MOTIVATION AND OBJECTIVE

Saudi Arabia, represented by the Ministry of Hajj, intends to require every person (Hajji) who is practicing the Hajj to wear a watch or wristband starting in 2030, which in turn means that every Hajji will be tracked by locations during the days of the Hajj. The news about the watch and wristband motivated us to work on crowd prediction, since tracking people is a difficult task in dense crowds. Also, a lack of the application of machine learning methods that deal with longer times of prediction stimulated me to work in this field to predict longer times than existing experiments; most prediction research has focused on predicting a pedestrian's movements only a few seconds ahead. Longer predictions can increase the objectives that benefit prediction, such as planning to set up barriers or avoiding them. Many crowd disasters, resulting from disorganization, happened during the Hajj between 1992 and 2015, and left a large number of people dead. These accidents happened not from bad intentions, but only because of disorganization and chaos, and this encouraged me us to dig into this field of crowd management. In this research, we try to answer these four questions:

- Can we predict human movements in a crowd using NNs trained with GAs?
- Do NNs and GAs make better predictions than the LSTM for human movements in a structured crowd?
- Do NNs and GAs make better predictions than LSTM for human movements in an unstructured crowd?
- Should we differentiate between crowd types in predictions because the results of the structured crowd and the unstructured crowd are different?

1.3 SIGNIFICANCE AND CONTRIBUTION

Obtaining data has become more important in many problem domains, because improvements in technology has resulted in the use of data in almost every aspect of our lives. One of these domains is using location data for people walking in crowds, which was difficult in the past, but will be easier in the future.

Simulating crowd movements and forecasting their motion provide a method to improve the design of spaces and the standards for safety. Designing environments for crowds by applying existing models that have specific rules for behaviors is problematic. Incorporating the cone of vision of pedestrians could possibly be the most important means for deciphering crowd motion, especially for structured crowded areas. Our work has proved that the combination of neural networks and genetic algorithms, using the nearest people in the cone of vision as the influence on decision-making, is effective in predicting pedestrians' movements. Another contribution of our research is that we used the LSTM to study the history for each agent in his cone of vision. We have also added a new measurement tool to this field, the cumulative distance error (CDE); this tool provides a new metric of accuracy in percentages regardless of metric lengths and makes it easier to compare longer term prediction techniques.

1.4 SCOPE OF THE RESEARCH

The research was constrained to the following:

• It only dealt with people's environment and excluded other obstacles, such as cars;
• It focused on crowded areas;
• It used two models of behaviors for its experiments, the Social Forces Model and the Flocking Model; and,
• It did not concentrate on normal or uncrowded areas.

1.5 AUTHOR'S RELATED PUBLICATIONS

1. Alajlan, A., Edris, A., Heckendorn, R. B. and Soule, T., "Using Neural Networks and Algorithms for Predicting Human Movement in Crowds," In Advances in Artificial Intelligence and Applied Cognitive Computing, pp. 353-368, Springer, 2021.

2. Alajlan, A., Edris, A., Sheldon, F. and Soule, T., "Machine Learning for Dense Crowd Direction Prediction Using Long Short-Term Memory," In 2020 International Conference on Computational Science and Computational Intelligence, pp. 686-689, IEEE, 2020, December.

3. Alajlan, A., Edris, A., and Soule, T., "Predicting Human Movements Using Machine Learning". The 23rd International Conference on Artificial Intelligence, ICAI'21: July 26-29, 2021, USA.

4. Edris, A., Alajlan, A., Sheldon, F., Soule, T. and Heckendorn, R., "An Alert System: Using Fuzzy Logic for Controlling Crowd Movement by Detecting Critical

Density Spots," In 2020 International Conference on Computational Science and Computational Intelligence, pp. 633-636, IEEE, 2020, December.

5. Edris, A., Alajlan, A. and Soule, T., "A Contribution to Crowd Control by Detecting Critica Density Spots" 2021 International Conference on Computational Science and Computational Intelligence, ICAI'21: July 26-29, 2021, USA.

1.6 DISSERTATION ORGANIZATION

The dissertation is organized as follows: Chapter 2 presents the literature review of the research; Chapter 3 presents how the neural network and the genetic algorithm (NN-GA) can be effective in pedestrians' trajectory prediction; Chapter 4 introduces a use of long short-term memory (LSTM) in the cone of vision method; Chapter 5 presents a comparison of results between NN-GA and LSTM in more than one scenario and more than one model; and, Chapter 6 presents the results of the comparison between the NN-GA and the LSTM in a variety of scenarios and with more than one behavior. The last chapter of this dissertation, chapter 7, includes a summary of implementation specifics, future works for predicting human movements in dense crowds, and our conclusions and future work.

CHAPTER 2

Background and Related Work[1]

2.1 SUMMARY

The growth of the population in the world has created crowds that require attentive handling. Crowded areas always create difficulties for authorities and organizers regarding management. Crowd management needs to be approached with great accuracy. Predicting crowd motion is one of the pillars in crowd management; it permits a semi-real visualization for organizers of how the crowd moves. The importance of prediction of individuals' next locations could also appear in numerous sections in our life, for example mobile robots amidst pedestrians. Hence many studies and models of human movement have been published in order to comprehend people's behaviors while walking. Machine learning is one of the tools that has been employed in this field to improve the predictions of humans' movements. Specifically, the Recurrent Neural Network (RNN) and the sub-type of Long Short-Term Memory (LSTM) have recently caught researchers' attention for the prediction of human trajectories, because of their success of dealing with sequential data in other fields. In this paper, we review the large number of research papers that use LSTMs in their studies to predict pedestrians' trajectories.

2.2 INTRODUCTION

Human action is usually for some purpose. An example of using purposeful common sense would be moving away from a person or an object to avoid a collision or moving towards a goal. Crowd management requires a contribution from more than one sector to help a crowd to move smoothly. These sectors can include health concerns (e.g. allowing pedestrians to reach water or emergency medical personnel to reach

[1]This chapter was submitted in an international journal .

people in need), planning, and management. The future existence of robots among crowds will require greater accuracy in predicting human movement. Health matters have been one of the most concerning issues for authorities and the World Health Organization (WHO), and after COVID-19, the need for more organized crowds has become even more important. Hajj and Kumbh Mela are two of the greatest gatherings in the world that deal with a very large number of people in small geographic areas. The Hajj is a ritual in the Islamic religion that every Muslim must perform at least once in his or her lifetime. In Hinduism, Kumbh Mela is considered as an enormously important religious celebration that takes place four times over the course of twelve years at four sacred rivers. For the Hajj Saudi Arabia is planning to give every pilgrim a tracking bracelet by 2030, which is a promising technology that will track pilgrims in the most crowded areas. Most datasets struggle with the limitation that they are confined to a specific number of people. These types of technologies will accelerate the success of obtaining more datasets from one of the most crowded locations in the world. Crowd areas and the scenarios within crowds can be categorized into more than one type: the structured crowd, in which people move in one direction and aim at one target; the unstructured crowd, in which people move in more than one direction and aim at many targets Rodriguez *et al*. (2009); merged paths, in which pedestrians in structured crowded areas come from more than one path, and merge into one route; and, intersections, in which pedestrians scramble into intersections. Additional scenarios exist, such as stopping points (water fountains, restrooms, pilgrimage points, etc.) in the structured/unstructured crowds.

With the increase of applications that used location-based services (LBS), the prediction for next locations became important and much more data will become available Sabarish *et al*. (2015). This feature makes it easy for algorithms to play a pivotal function in the future. Rudenko *et al*. (2020) raised three substantial questions with regards to the data for crowd prediction. The first question was about the quality of the metrics that have been used to measure the accuracy of prediction. The second question was about whether all approaches to prediction reach similar levels of accuracy. The third question was whether the problem of movement prediction has been resolved or not. Bighashdel and Dubbelman (2019) considered the employment of numerous mechanisms in crowd prediction due to the intricacy of agent behaviors. They pointed out that the effectiveness for algorithms is based on their performance in the metrics that have been used, with different metrics leading to different apparent effectiveness. Similarly Rudenko *et al*. (2020): asked are existing metrics capable of producing the desired results? Are there extra metrics that should be used? so that we can distinguish between algorithms, and not only for metrics that have been used to date.

In the report of the World Health Organization. (2017, December), various factors were considered as a hazards for crowds, primarily including inappropriate preparation for medical care and incomplete crowd control. However, the risk factors for large crowds can be change from year to year; for example, insistence on social distancing started after COVID-19. To the best of our knowledge, there is currently no study in the field of pedestrian prediction that assumes the factors of COVID-19. Chapter 7, Section 7.4 presents some preliminary results for social distancing. Before the application of machine learning to the prediction of human movements, the question was: do existing approaches have the ability to capture all the complexities in human movements Bighashdel and Dubbelman (2019)? Dangerous situations in structured/unstructured crowded areas could be predicted by some methods, and these dangerous situations could be avoided by human observation and prediction of the crowd condition. For example, planning and designing barriers for a crowd in motion should be studied before they are set up. Various modeled motions and algorithms have been applied in this field. Physics-based method, movements based on the target, and the pattern mode approach are three of the most popular modeling approaches for human movement Rudenko *et al*. (2020).

Artificial intelligence approaches seek to build a system that can comprehend and predict people's behaviors and intentions Kong and Fu (2018). Most of the LSTM based methods use the past positions of agents and additional information, such as the environments, or the pedestrians' targets to try to predict movement. For our work, we summarize the most recent studies in this domain, and make some comparisons between approaches.

In this paper, we have excluded the studies that deal with heterogeneous environments, such as environments including pedestrians and vehicles, and we have focused on the research with environments that treat only pedestrians, or at most robots among pedestrians. Moreover, our survey concentrates on studies that have been published in the last four years.

2.3 PREDICTING HUMAN MOVEMENTS

2.3.1 *Trajectory Prediction*

Predicting the trajectory of pedestrians is essential in diverse applications, including robot navigation and autonomous driving. However, predicting trajectories reliably requires an understanding of human social behaviors, which are difficult to express using hand-crafted rules. At the same time, although Long Short Term Memory (LSTM) networks have shown great potential in learning and predicting social be- haviors, they often neglect the current intention of neighbors in a crowd scenario.Accordingly, Zhang et al. (2019) proposed a data-driven state refinement module for LSTM (SR-LSTM) that leverages neighbors' current intentions and refines current states of all participants in the crowd, both iteratively and jointly, through a messaging mechanism.The researchers also introduced a social-aware information selection mechanism to support the extraction of the social effect of neighbors. Experiments done using this approach showed excellent outcomes. The proposed algorithm, SR- LSTM has various advantages and disadvantages. Perhaps the main advantage is that it considers the current neighbor states for timely inference. By considering the current states of neighbors, the accuracy of the prediction of the next movement is enhanced. Another advantage is that the algorithm adaptively selects useful information from neighbors based on their locations and motions. Furthermore, this algorithm offers a large range of parameters, thus eliminating the need for fine adjustments. On the downside, the proposed algorithm has several weaknesses, including the failure to remove vanishing gradients completely and the need for extensive resources and time when training it. Additionally, SR-LSTMs might not be effective in crowd management settings where developers need to remember information for extended periods. Overall, the study recommends the utilization of SR-LSTM when predicting trajectory. Future studies should attempt to apply this algorithm in practical settings, such as autonomous driving, to establish its effectiveness and efficiency. The LSTM architectures have emerged as important approaches for path prediction. Es- essentially, anticipating trajectories is essential in computer vision, autonomous systems, and robotics. Hasan et al. (2018) proposed the use of vislets, which are short sequences of the pose estimates of the head, to increase the accuracy of trajectory prediction. They then integrated these vislets into MX-LSTM, which is a novel framework that can jointly forecast positions and the orientations of heads by leveraging vislets and tracklets (or short track). MX-LSTM captures the interplay between tracklets and vislets. It also forecasts future head poses, which is essential as it improves long-term trajectory prediction, Hasan et al. (2018). By combining attention-based social pooling with head pose estimates, the algorithm exhibits state-of-the-art forecasting capabilities when tested on different datasets. One of the major advantages of the algorithm proposed in this study is that it exhibits exemplary performance in situations when pedestrians slow down. This situation as been difficult to model in other approaches. Another advantage is that this algorithm considers head pose estimates, hence significantly increasing the accuracy of trajectory prediction. The authors emphasize that head poses influence trajectories, and their inclusion in the algorithm enhances trajectory prediction. The main weakness in this paper relates to the requirement of extensive resources during training. Despite this drawback, the authors stress the consideration of head poses in trajectory prediction. Future studies should explore the utilization of head poses in other algorithms and situations concerning trajectory prediction to improve accuracy.

Predicting human trajectory continues to be a major challenge to computer vision systems. As a result, Manh and Alaghband (2018) developed a forecasting system utilizing information obtained from the scene and the trajectories of human movements in static crowded scenes. The approach adopted encompassed utilizingscene-LSTM to capture scene information and pedestrian-movement LSTM to capture movement trajectories. The researchers superimposed a two-level grid structure and explored common trajectories happening within each grid cell. They then trained two coupled

LSTM networks (comprising a pedestrian-movement LSTM and its cor- responding scene-LSTM) to predict future movements. The authors demonstrated that using common path information enhanced the accuracy of the prediction. They also designed a scene data filter to enable the selection of the relevant types of information relative to the state of the target. Based on the experiments done, the proposed approach minimizes displacement errors significantly as compared to current LSTM-based methods. A key advantage of the algorithm is that it considers scene information and demonstrates how this information can be utilized to enhance the prediction of movement trajectories. Another advantage is that the method reduces displacement errors, which are characteristic of LSTM-based methods. However, the algorithm requires a lot of time and resources to train and utilize because it is based on the LSTM. Nevertheless, the study demonstrates how scene information, combined with pedestrian movement, can improve trajectory prediction. Future studies ought to explore how to integrate the social model into the scene model to improve the accuracy of the prediction. It would also be important to examine the interactions of humans with other moving or static objects and how these interactions influence trajectories

Crowded scenes of human beings are characterized by three critical elements that inhibit the accurate forecasting of human behavior: interpersonal relationships, social acceptability, and multi-modality. Modeling the trajectory of human beings is imperative for applications such as social robots and autonomous driving. To address these challenges, Gupta et al. (2018) developed a trajectory prediction system that combined sequence prediction and generative adversarial networks. The result of this combination was a recurrent sequence-to-sequence model for aggregating information, observing the histories of motions, and forecasting future behavior. The authors trained the model against a recurrent discriminator to predict possible social features. They also utilized a novel variety loss to enable diverse predictions. According to the experiments done, this model outperformed existing solutions. One of the key advantages of the algorithm is that it has a variety-loss capability that enables the generative network to cover different possible paths. Another advantage is that the algorithm can learn a global pooling vector. A final advantage relates to the exemplary results of the algorithm after evaluations were completed. However, the algorithm fails to consider static or other moving objects within a crowded space. Nevertheless, the study suggests the combination of generative adversarial networks and sequence prediction for modeling human trajectories. Future studies should attempt to consider other moving objects and static elements within the algorithm.

Xu et al. (2018) have designed a deep-learning framework, Crowd Interaction Deep Neural Network (CIDNN) to predict future steps for pedestrians by taking into account the effect of people nearby. They believe the future positions for pedestrians depend on three things: first, the movement data, such as velocity and acceleration, for the targeted pedestrian; second, the movement data for all pedestrians in the vicinity; and third, the distance between this specific pedestrian and other pedestrians. The CIDNN is built based on three factors: the movement's history encoder that uses the LSTM networks; positions for others based on their spatial locations; and, the future coordinates for pedestrians. They used six datasets in their experimental work, including New York Grand Central (GC), ETH, UCY, the CUHK Crowd Dataset, and the subway station dataset. The approach has shown promising results in comparison with some methods that have been used before.

Social robots and other autonomous systems are dependent on the capability to accurately model human motion. However, the process of modeling trajectory is challenging as people move based on their intention and it is almost impossible to know the destination of the pedestrian. Accordingly, in this article, the authors presented an interpretable and end-to-end model for predicting human trajectory Dendorfer et al. (2020). The model adopted an intuitive two-stage process: estimating the goal of the pedestrian and developing a routing module. The role of the module is to estimate the possible trajectories that can be followed to reach the estimated goal. The model considers the visual scene context and pedestrian dynamics to estimate the posterior-over-possible goals. Thereafter, it predicts trajectories using a recurrent neural network. The overall approach adopted in this study was comprised of three steps: developing a novel architecture for estimating future goal positions, training the network using the Gumbel Softmax trick through the stochastic process, and evaluating the approach using various public benchmarks and qualitative measures. The algorithm utilized has various advantages. To start with, the algorithm is goal-conditioned in that it predicts the final position first and then generates the appropriate trajectory based on the position. The utilization of the generative adversarial network (GAN) was also important as GANs often generate data that is like the original position, hence helping with the trajectory determination. Furthermore, GANs are detailed and can interpret different versions of the data. As a result, their utilization helps to determine potential trajectories easily. More importantly, the algorithm attained state-of-the- art results on various datasets. It also generated multi modal, feasible, and diverse trajectories. On the downside, a key disadvantage of the algorithm is that it is difficult to train.

The algorithm relies upon the loss function, which is difficult to optimize. Indeed, another disadvantage is that this algorithm cannot be applied to problems involving speech and text. Future studies should explore the possibility of improving forecasting by integrating different factors into the algorithm. For instance, weights could be assigned to neighbors based on distance. Additionally, the level of comfort between pedestrians ought to be considered. Human trajectory prediction is an important area of research due to the vast applications it could enable. In this study, the authors developed an approach to enhance the accuracy of human trajectory forecasting by using visualizations of crowded spaces Singhal and Indu (2020). The idea was to optimize accuracy by learning the entire pipeline and computing values for numerical stability. The researchers combined semantic and social components to exploit the local awareness of the surrounding space. Accordingly, the algorithm that was developed combined the social-LSTM and social elements and the scene-aware LSTM model and integrated more features to encode interactions in the space to enhance accuracy Singhal and Indu (2020). In other words, the approach involved extracting social conventions from observed trajectories and then augmenting them with semantic information derived from the neighborhood. The algorithm that was developed had various advantages. To start with, the LSTM model that was utilized allows pedestrians to share their hidden representations. The social pooling mechanism included in the LSTM merges hidden states in immediate neighborhoods to make each trajectory aware of its neighborhood. Another important advantage is that the algorithm considers other factors that influence pedestrian dynamics, such as zebra crossings and sidewalks. Therefore, by using a combination of social elements and scene-aware LSTM, the authors were able to consider human dynamics, neighborhood semantics, and past observations. Perhaps the most important advantage of the algorithm is that it produced better results than other state-of-the-art methods. More importantly, it included an error-calculation system, which significantly reduced errors, hence enhancing the accuracy of the models. The algorithm also has drawbacks: specifically, the reliance on the LSTMs means that the time required to train the algorithm is lengthy; and additional memory is needed and the LSTMs are highly sensitive to random weight initialization.

The study recommends the combination of social and semantic elements to exploit the local awareness of the surrounding space to increase accuracy. The integration of additional features was shown as being helpful in improving trajectory prediction. Concerning the direction of future research, the authors highlighted the need to explore different datasets with more nuanced dynamics, such as considering the motion of elements such as cyclists and vehicles to help refine trajectory prediction. Furthermore, the researchers recommended an in-depth examination of the impact on trajectory forecasting of semantic segmentation of areas in a neighborhood. Trajectory predictions are beneficial in smart cities as they can support autonomous driving and urban scene monitoring. Accordingly, the researchers in this study focused on the modeling of single trajectories using recurrent neural networks. The main contribution of the study to the literature is the examination of the effect of the choice of the output representation on the network performance. Although the study does not present a new trajectory prediction system, it assesses how the current state- of-the-art systems encode their output. The researchers showed that the performance of an LSTM-based neural network improved when estimate residuals, which are relative displacements or deviations to linear regression, were utilized Ek-Hobak et al. (2020). Accordingly, the authors concluded that residuals ought to be used instead of absolute positions if better prediction outcomes are to be realized. The algorithm that was utilized has various advantages. The first one is that it combined the idea of neural networks with residual output representations to improve trajectory prediction. The normal approach involves absolute positioning, which is ineffective Ek-Hobak et al.

(2020). Another advantage is based on the utilization of the LSTM method since the LSTM algorithm does not require fine adjustments. On the downside, however, the researchers did not try to propose a new algorithm for predicting trajectories; they instead focused on showing how the performance of deep recurrent neural networks can be improved by using residual positioning. Nevertheless, the overarching recommendation is that neural network-based trajectory forecasting systems ought to utilize residual output representations to predict trajectories more accurately. Therefore, future work ought to focus on the use of residual output representations to enhance accuracy in other systems for predicting trajectories.

More importantly, future research also ought to assess how the idea of residual output representations can improve predictions in situations where the scene includes moving objects, such as cars and bicycles.The orientation of the head can give an idea regarding the trajectory of a person. Accordingly, the researchers in this study postulated that the head pose and the human trajectory could be jointly modeled. The authors introduced the MiXing LSTM (MX-LSTM), which encodes the relationship between peoples' dynamics and the movement of the head Hasan et al. (2019). The utilization of the MX-LSTM enables the mixing of tracklet and vislet streams in the LSTM hidden-state recursion Hasan et al. (2019). The algorithm presented in this study has the major advantage of integrating head poses into trajectory prediction. From a social and observational perspective, head poses provide insights concerning the potential direction that one intends to take. Similarly, the researchers encoded head poses into vislets to improve motion prediction. Another advantage of the algorithm is that it allows unconstrained optimization, which enables the inclusion of other variables, such as the possibility of a person belonging to a social group within the scene. Furthermore, the algorithm performs exceedingly well when people slow down and look around before changing direction. However, the algorithm makes use of LSTMs, which take a longer time and more resources to train. Our recommendation is that trajectory prediction systems ought to consider head poses to enhance the accuracy of forecasting. Current approaches largely focus on the position of the pedestrian and the location of neighbors. However, such approaches fail to consider other aspects that could improve accuracy, such as head poses. To integrate head poses into the forecasting process, the MX-LSTM model ought to be utilized. Future work should attempt to examine the proposed algorithm using actual head poses as because this study made use of estimated ones.

Doing so would help determine whether the system can be applied to practical settings. Within smart buildings, predicting the trajectory of occupants can be useful to enhance space utilization, crowd management, operations, security, comfort, and evacuation. In this study, the researchers presented and compared two trajectory methods. The models were implemented and compared at the same time and location in a reliable and non-intrusive manner. The idea was to ensure that the conditions were similar enough to limit confounding variables. The test space that was utilized was a multi-utility area in a public building with installed three-dimensional cameras to capture spatial location coordinates from a bird's-eye view Das et al. (2020). Based on the information collected, the accuracy of both models for predicting occupant trajectories was compared. According to the findings, the gated recurrent unit (GRU) algorithm was found to be more accurate than the LSTM in predicting trajectories Das et al. (2020). Essentially, the GRU had a lower mean square error (MSE) and mean absolute error (MAE) than the LSTM, highlighting its high fidelity. This study was unique as it compared two models rather than presenting one novel algorithm. To start with, the LSTM eliminates the need for fine adjustments due to the utilization of a large range of parameters. However, it has a complex architecture, suffers from the exploding gradient problem, and makes use of time-consuming computations. Additionally, as the study found, it is less accurate as compared to the GRU. Likewise, the major advantage of the GRU is that it is more accurate than the LSTM, as shown in the experiments. In addition, it requires fewer training parameters and, therefore, needs less memory and exhibits faster processing speeds. However, for large datasets, the accuracy of the GRU might be lower than that of the LSTM. The overall recommendation of this study is that the GRU is better than the LSTM in predicting occupant trajectories in a smart building. Indeed, the GRU and the LSTM have emerged as state-of-the-art methods for capturing long- and short-term dependencies among variables, which makes them suitable for tracking applications. In addition to possessing the capacity to capture interactions between occupants in a scene, both models consider rich semantic information that characterizes the typical occupant space. To further research in this area, future research should attempt to enhance the applicability of the GRU in practical environments by considering other factors that affect human motion within buildings.

Prediction of human trajectory movements is essential for autonomous and self-aware robots. Current approaches for predicting human trajectory focus on the effect of neighbors on a person's movement; they do not consider the destination of the person. However, destination plays a vital role in route planning and the general trajectory of a person. Xue et al. (2017) proposes a two-stage prediction method that aims at coming up with multiple paths. The paths lead to the same destination, but each possibility is different. The method proposed by Xue et al. (2017) is referred to as bi-prediction. Before trajectory forecasting, this method used bi-directional long short-term memory to classify paths into smaller route subgroups. The field of artificial intelligence has made significant progress over the years. However, there are some challenges in the automatic analysis and prediction of human trajectories. Therefore, the training of machines to foresee human trajectories is an essential concept for socially aware robots and tracking systems. Methods of trajectory prediction can be divided into model-based and long short-term memory. However, model-based methods fail to accurately predict trajectories in crowded scenes because they rely on energy functions and specific pedestrian settings rather than learning from trajectory datasets. However, trajectory prediction based on LSTM architecture has attracted much attention owing to the success of the LSTM in the sequential data processing. An advantage of this approach is that it utilizes both the past and the future contexts in data sequences. Pedestrians choose their routes based on the intended destination in a real-world application. Trajectory prediction will be more accurate if pedestrian destinations can be learned through trajectory data. Generating multiple pedestrian trajectories would be more beneficial for anomaly detection to depict complex multiple pedestrian movement patterns. A disadvantage of this approach is that trajectory prediction in complex clustered scenes with multiple exits and entry points is difficult. This is because human movement is influenced by several factors not modeled into the bi-directional algorithm. Thus, the pedestrian may opt for an entirely different unconventional route despite having multiple paths to a destination. Therefore, it is not possible to accurately predict pedestrian movement. For a recommended future work, a two-stage trajectory methodology maybe introduced. LSTM architecture classifies potential destinations, and thus prediction accuracy is much higher. This prediction method can yield multiple prediction trajectories with different possibilities. Some trajectories can have lower probabilities and can be labeled as abnormal trajectories. Future work includes automatically detecting entry and exit points in crowded scenes and considering factors influencing human movement

2.3.2 *Movement Patterns and Scenario*

The complex movement of crowds and the complex environments that crowds often occupy makes it challenging to forecast the path of pedestrians. At the same time, forecasting the movement of people in crowded areas is important for many computerized vision-based applications, such as smart video surveillance. Current LSTM-based methods for predicting pedestrian paths are based on the availability of rich context data, including information about the background scenes, exits and entrances, and static obstacles. The requirement of rich context information makes approaches inappropriate in most applications and limits their generalization. Additionally, the inclusion of contextual information increases computational overload.

Accordingly, Xue et al. (2019) proposed a joint location-velocity attention LSTM method for predicting trajectories. The algorithm was developed based on the idea of enhancing the LSTM network and training the attention mechanism. The resultant model was used to learn to combine the location and velocity of information optimally in the prediction process. The algorithm was evaluated on several publicly available databases and exhibited better results as compared to other prediction approaches. More importantly, the algorithm showed extensive generalizability. The location-velocity attention LSTM algorithm has various advantages. The main benefit is that it does not need additional contextual information. This aspect is particularly important as it enhances the generalizability of the algorithm. Another advantage is that it exhibits better performance than other state-of-the-art prediction models on different datasets. However, since this algorithm is based on the LSTM, it requires a lot of resources and time during training. Although the researchers recommend this algorithm for predicting pedestrian trajectories, they recommend further improvements to enhance prediction accuracy. Accordingly, future studies should focus on utilizing temporal attention mechanisms to enhance prediction accuracy. Additionally, future studies ought to assess the algorithm in diverse tasks to improve the process of tracking people in a high-density crowd.

Path prediction has also received attention due to its potential utilization in various applications beyond crowd prediction, particularly movement in areas with diverse objects or movement types, e.g. pedestrians, bicyclists, skateboarders, etc. The conventional approach to prediction considers diverse factors to not only forecast paths but also eliminate collisions. Scene semantics have emerged as a technique to promote reliability in prediction in numerous applications. However, these techniques consider all objects to belong to the same class, but classes of objects differ in terms of aspects such as distance, area, and speed. It is imperative to consider the path corresponding to each class of objects to enhance accuracy. Therefore, Minoura et al. (2019) proposed a model that views the target type as an attribute and considers information regarding the environment to predict paths based on the individual categories.

The features integrated into the LSTM algorithm include the attribute, past trajectory of the object, and the semantics of the environment. Doing so enabled the researchers to forecast paths for each target. Based on the experimental results, the method produced highly accurate predictions. The method proposed has several advantages. The most essential advantage is that it considers the fact that different categories of objects exhibit different behaviors in terms of movement. As such, the prediction's accuracy is likely to improve if category-specific factors are considered. Another advantage is that the algorithm considers the semantics of the environment. Furthermore, the proposed method can be applied to different targets using a unified framework. On the downside, the method failed to achieve good prediction results in cases where the objects were bicycles and skateboards. Part of the reason for this is that there are mini- mal traffic rules governing the utilization of these methods of transportation. Still, the study recommends considering the semantic environment and object attributes when predicting path trajectories. Future studies should encompass enhancing training for cases of rare attribute targets such as bicycles.

The interaction between vehicles and pedestrians should be considered in autonomous driving as it influences safety. As such, predicting pedestrian trajectories is essential. However, the complex movement patterns of pedestrians in the presence of vehicles make the process of predicting trajectories difficult. Still, recent research has identified LSTM as a promising approach to trajectory prediction by perceiving the problem as an issue of sequence learning. Cheng et al. (2018) conducted a study in which they enhanced the effectiveness of the LSTM by incorporating a social grid. The proposed Social-Grid LSTM utilizes the LSTM cell structure by incorporating a social pooling operation to enable an influence relationship among neighboring pedestrians. The proposed method was evaluated using two public datasets and compared to two baseline methods. According to the findings, the proposed method outperforms the baseline approaches. A key advantage of this algorithm is that it integrates the human-to-human model. The role of this model is to enable the consideration of social aspects related to trajectories. Another advantage is that it adopts a two-dimensional Grid LSTM model, which differs from the conventional LSTM structure in terms of the parameter transfer mechanism from layer to layer. However, this algorithm requires extensive resources in training and it is sensitive to random weight initializations. Nevertheless, the authors recommend the inclusion of the social grid in trajectory prediction as human-to-human interactions have a profound impact on people's movements. Future studies should attempt to integrate attention mechanisms into the algorithm. Moreover, it is imperative to examine the performance of the proposed method in practical autonomous driving applications.

Trajectory prediction, while important, continues to be challenging as it has a multi-modal issue. Multi-modal means that there is the possibility of predicting more than one specific path in open scenes. Additionally, it entails social interactions that can influence movement decisions and the presence of structures can inhibit movement in certain directions. To address these problems, Huynh and Alaghband (2019) developed a novel forecasting system that combines pedestrian-LSTM with scene-LSTM to predict pedestrian trajectories in static crowd scenes. The inclusion of the scene-LSTM aimed to enable the collection of information relating to the commonly utilized paths, thus enhancing the accuracy of forecasting in local areas. The approach also included integrating scene data filters to choose the most essential scene information in local areas and combine it with pedestrian-LSTM to enhance the accuracy of prediction. According to the experimental results, the proposed approach outperformed similar works and exhibited better accuracy in diverse scenes. The main advantage of the algorithm is that it considers the typical human movements in localities within a given scene. This aspect is critical as human beings are likely to adhere to common movement patterns in each area. Another advantage is that the algorithm exhibited better results than similar approaches in different scene settings. However, its main disadvantage is the failure to recognize the importance of the social aspects influencing trajectories. Still, the authors recommend the integration of common human movements when predicting trajectories in an area. Future studies should focus on integrating social information into the scene model to improve prediction. Additionally, future studies ought to consider the interaction between humans and other objects and the impact of the interactions on trajectories.

Applications such as robot navigation and autonomous driving depend on the accurate estimation of pedestrian trajectories for the prevention of collisions. However, modeling pedestrian trajectories in the presence of autonomous vehicles is challenging because it is influenced by the movement of other pedestrians, and by the impact of the autonomous vehicles, as well as by static objects within the scene. To advance research in this area, Haddad et al. (2019) presented a new spatio-temporal graph-based LSTM to forecast the trajectory of people in this type of heterogeneous crowded environments. The proposed approach considers the interaction between people and dynamic elements (such as autonomous vehicles and robots) as well as static objects. Experiments done showed that the approach performed better than state-of-the-art methods utilized in predicting human trajectories. More importantly, the approach minimized the average and final displacement errors with respect to other methods. The main advantage of the algorithm is that it operates on the global and local contexts around the pedestrian during the prediction process. The local aspect is key as people typically make movement decisions regarding static objects when they are closer to them. Another advantage is that the algorithm attains significant qualitative and quantitative improvements over other methods. However, since the algorithm is based on LSTM, extensive resources and time are needed to train and utilize it. For future studies, attempts should be made to apply this algorithm to practical problems.

As noted previously early research in the area of social robot navigation focused on modeling the movement patterns of human beings. However, the approaches taken failed to recognize the subtle and complex interactions among humans in crowded spaces, thus making it difficult to define the path of the robot. Because of the need to consider these complex interactions, Vemula et al. (2018) developed a novel trajectory model called social attention, which considers the relative significance of each pedestrian in a crowd, regardless of proximity. During the evaluation process, the researchers examined the performance of this model against other state-of-the-art approaches using two crowd datasets and realized positive outcomes. A key advantage of their algorithm is that it considers the relative influence of each person in a crowd on the behavior of other pedestrians. Essentially, everyone, regardless of their proximity, influences human trajectories, but the extent of the influence differs. Another advantage is that the algorithm outperforms state-of-the-art methods on two publicly-available datasets in terms of prediction errors. However, a major weakness is that the algorithm fails to consider static objects. Nevertheless, the study recommends incorporating the relative influence of individuals on the movement trajectories of others. Future studies should extend the model by considering static obstacles within the scene. Additionally, future research ought to attempt to examine the model in practical settings by placing a robot, powered by this model, in a human crowd. In scenarios where social robots and autonomous vehicles occupy the same space as human beings, the need to forecast long-term future paths becomes essential. Shi et al. (2021) explored long-term path forecasting problems in crowds. The idea was to generate future sequence trajectories by utilizing short observations. The authors observed that current methods rely on modeling social interactions and forecasting multi-modal features, which is a daunting task as machines cannot consider social interactions and the uncertainty of features simultaneously. As a result, the authors presented a model that jointly considers different interacting motion sequences and forecasts future multi-modal socially acceptable distributions. The evaluation of this model demonstrates that it can predict socially acceptable distributions of future paths effectively in complicated environments. The major advantage of this algorithm is that it predicts socially acceptable trajectories, which remains a challenging area in the research. It does so by adopting a new aggregation mechanism that selectively incorporates the latent states of concurrent movements in a crowded environment by using a messaging capability. It also includes a loss function to enable the generation of socially possible future distribution. Another advantage is that the model exhibits conformity and coherence to social norms. However, the algorithm did not consider all possible future nodes of an interacting group. Additionally, other moving objects, such as cars and bicycles, were not considered. Still, the authors recommend the integration of socially acceptable distributions when predicting future paths. Future studies should focus on expanding the model to enable it to predict future trajectories in interacting groups and consider different static and moving objects.

Intelligence surveillance includes classifying objects, tracking, describing behavior, and detecting motion. Detecting and understanding motion is a critical area as it helps in predicting future trajectories. To help in developing a system to predict human trajectories, Peng et al. (2021b) proposed the social-relationship attention LSTM algorithm (SRA-LSTM). To begin with, the researchers created a social-relationship decoder to collect information concerning the social relationship between pairs of pedestrians. Next, they adopted the social-relationship feature and latent movements to obtain the social relationship of pedestrian pairs. The method was compared with similar methods using public datasets and better performance was noted. One of the key advantages of this algorithm is that it models social relationships using the temporal correlation of relative positions, which produced exemplary results. Another advantage is that the algorithm produced better outcomes compared to similar state-of-the-art approaches. In addition, the utilization of the social-relationship attention model enabled the simultaneous consideration of social relationships and the potential movement interactions on decisions relating to movement. On the downside, the algorithm failed to accurately predict future trajectories in some scenarios, including when a person moves towards the wall and stops. Still, the paper recommends the utilization of the SRA-LSTM framework to predict human trajectories. Concerning future studies, it is imperative for scholars to explore how scene-specific information can be integrated into the model to improve the accuracy of predictions. The human-scene component was missing in this study, and thus there is the need to examine it in the future. Furthermore, the proposed method should be examined when utilized in robot navigation systems.

The utilization of deep learning methods in forecasting human trajectories has gained prominence in recent years due to the potential applications it promises. However, current deep-learning LSTM methods often ignore scene layouts and rely exclusively on neighborhood influence in predicting trajectories. This aspect weakens the accuracy of the trajectory prediction. Accordingly, Xue et al. (2018) developed a hierarchical LSTM-based network, which considers scene layouts, as well as the neighborhood influence of pedestrians. The algorithm utilized three LSTMs to capture information relating to the scene, social setting, and pedestrians, respectively. The authors selected a neighborhood with a circular shape rather than a rectangular one.

According to the experimental results, this approach produces better outcomes than other state-of-the-art techniques. The utilization of the circular-shaped neighborhood also enhanced the accuracy of forecasting. The advantages of this algorithm included the consideration of the scene layout and the adoption of a circular neighborhood. The idea of combining three LSTMs, as well as using scene information, contributed to the high-quality prediction outcomes observed during evaluation. However, the algorithm lacked a temporal attention mechanism, which means that other factors influencing trajectories were not considered. Additionally, it did not consider the influence of neighbors based on the differences in distances between them. Overall, the study recommends the use of the SS-LSTM prediction model because it offers an effective way of integrating scene information in trajectory prediction. Future studies ought to improve the model by assigning influence weights to neighbors based on the distance between them and incorporating a temporal attention mechanism to collect and aggregate other information necessary for predicting trajectories. Moreover, an additional network should be integrated into the model to learn other factors influencing trajectories, such as comfortable distances

Deka et al. (2018) focused on the area around each pedestrian, and the pedestrian's positions., They, therefore, fed this information to the Structural Recurrent Neural Network using LSTM. Focusing on the positions and the surrounding environment, without the velocity, gave a superior result. Two datasets were used: the ETH dataset, and the ATC dataset.Peng et al. (2021a) suggested a Spatio-temporal Interaction-aware Recursive Net- work (STIRNet) to forecast various acceptable trajectories of people in the scene. They used a graph attention network to shape the spatial interactions, which is joined with the encoding as inputs to the LSTM to capture movement advantage. The LSTM is part of their model architecture to predict human trajectories. Two benchmarks were used in this experiment, ETH, and UCY. Rozenberg et al. (2021) used asymmetrical bidirectional recurrent neural networksto encode pedestrians' paths. The methodology is based on three things: input embedding; asymmetrical bidirectional recurrent neural network architecture, and the decoder. The input contains the positions' coordinates, velocity, and trajectory embeddings. The asymmetrical bidirectional RRNs have the backward and forward hidden states, where the future information can be used in the forward pass. The decoder obtains the predicted locations for each pedestrian. They used the Trajnet++ benchmark that has various datasets, such as ETH, UCY, WildTrack, L-CAS, and CFF.

The Long Short-Term Memory (LSTM) algorithm is utilized extensively to predict pedestrian trajectories. Accordingly, the researchers in this study deployed the Grid-LSTM, which functions over multi-dimensional feature inputs, to model pedestrian trajectories. Specifically, they proposed a Graph-to-Kernel LSTM (G2K LSTM) that converts a spatial-temporal graph into the kernel to approximate correlations between pedestrians Haddad and Lam (2020). The correlations suggest that relationships among people as individuals tend to maintain close distances when they feel comfortable or are evading collisions. In the research, the authors utilized this Grid-LSTM as an encoder to learn about future neighborhoods and their impacts on pedestrian movement. Based on the analyses done, the authors established that their model outperformed other state-of-the-art methods. Exemplary performance was also noted across different datasets. A major advantage of the proposed algorithm is that it estimates the neighborhood given the static context and a combination of social cues.

The authors proposed an adaptive neighborhood based on spatial constraints and visual locus Haddad and Lam (2020). Another key advantage is that generalization is possible because the algorithm does not rely on a fixed assumption relating to specific scenes. The fact that scenes are not defined strictly means that the algorithm can be utilized in diverse environments. Furthermore, the authors demonstrated that this algorithm performed better than other state-of-the-art methods. On the downside, the algorithm requires a lot of time to make predictions. For instance, it required about four seconds to predict trajectories for a scene comprised of only twelve pedestrians. Still, the overall recommendation is that the G2K LSTM is accurate when predicting trajectories because it leverages adaptive neighborhoods. Concerning future research, the authors highlight the need to generate more realistic perceptions of the neighborhood's importance. Additionally, there is a need to minimize network-component complexity to reduce the time required to process data in forecasting trajectories

Stationary and mobile elements influence the behavior and mannerisms of a human being in a crowded setting. Bartoli et al. (2018) propose a context-aware human trajectory prediction model that considers human-to-human interaction and the interactions of humans with their surroundings. This model will be context-aware, thus enabling it to predict human trajectories in various settings, such as malls and museums. Usually, every pedestrian has an objective they wish to achieve. This objective may be to reach a particular destination, reach another person, or avoid a specific obstacle. Therefore, they can adjust their trajectory accordingly to achieve their objective. Human navigation is a complex process to be understood by robots. Some past works in this field have predicted targeted human paths by interpreting how humans interact with the agents in the scene. However, these models are not very accurate because they may not have prior knowledge of the interaction between humans and static elements in the scene. These static elements include sidewalks, trees, and staircases. The context-aware model assumes that human movement is based on the interactions with static and dynamic elements within the environment. This work builds upon the social LTSM model and considers human interactions with others and the space surrounding them. An advantage of this model is that it considers the environment in which a person is moving to get to an accurate trajectory prediction. Static and dynamic elements are first identified. The static elements are manually defined, including exit and entry points. A disadvantage of this model is that each agent in the space contributes equally to trajectory prediction. For a recommended future work, a model that also considers environmental conditions would be useful because the weather also influences human trajectory.

Amirian et al. (2019) propose the use of the crowd simulation method to imitate the behavior of pedestrians in a given setting. First, the learning dataset will consist of the observed trajectories, and then, using generative adversarial networks, the patterns will be analyzed, and new trajectories will be generated. This approach involves developing a crowd while still focusing on real-time interactions among various agents. The simulated crowds exhibit characteristics similar to real crowds, but elements can be added to the virtual environment for a richer scenario. Simulation of crowds involves applying artificial intelligence, computer graphics, and psychology, the latter because humans are considered social beings. To model human behavior, crowd simulation requires real-world input. Crowd behavior is observed from this input, and the same behavior is replicated in the virtual crowd. A limitation of this system is that the simulated crowd cannot exhibit any behavior that is not part of the input. However, the proposed system overcomes this limitation by enabling the simulation of crowds while at the same time giving the simulated crowd freedom to alter trajectory when necessary. In summary, this approach can generate new trajectories based on the learned input trajectories. These trajectories are then incorporated into the crowd's simulations while allowing interactions among the agents. To generate trajectories, a planner environment is used to model agents. A trajectory is then defined as a path an agent moves in T seconds. The trajectory also involves consideration of the agent's speed. The generative adversarial network (GAN) consists of a generator and a discriminator. The generator generates new samples while the discriminator decides whether a sample is authentic or not. The advantage of this model is that elements can be added to the virtual scene to make the virtual location rich in terms of static and dynamic agents. This enables the model to come up with more potential trajectories. However, a disadvantage of this model is that training the model is a complex process.

It requires real-time input data to generate trajectories, which the model will later use to create more trajectories. Another disadvantage is that the model does not perform well in environments where agents must act independently. These limitations will be looked into to improve the model in future work.

Recently, an increase in robot interactions with humans has been noted. In airports, for instance, there is a need for robots to help humans make the best decision on what route to follow to avoid crowds. In universities, there can be a need for robots to help humans access the different facilities in the institution. The increase in robots potentially creates collisions with humans and can affect safety in general Hamandi et al. (2019). This paper explores and explains deep model for target-driven imitation (deepMoTIon), which is an algorithm that trains robots to mimic human navigation while in crowds. The advantages of the algorithm is that in all the tests, it reaches the intended destination. The training set is twisted in different angles to make the learning challenging to assure there is a well-trained model in the end. The algorithm has disadvantages as well; it takes a lot of time to learn, which increases the resources that are used in training. The robot is interested in the reward of getting to the target instead of concentrating on the learning. Human beings are inconsistent in their movement, and in some cases, they collide. This can make robots learn the inconsistency, and their collision with humans could be fatal. Light detection and ranging (LiDAR) is the method that was used to learn human movement and train the robots. To increase the accuracy and the natural state of the experiment, an improvement in the algorithm can be achieved if real images are used. This will help the algorithm outperform the social force model; the SFM is the only algorithm that outperforms the deepMoTIon in terms of the number of collisions. Penalizing the robot heavily for collisions will assist the robot to concentrate on learning rather than reaching the target.

Xue et al. (2017) suggested a use of bidirectional LSTM with a method called bi- prediction. This method takes two phases to process its operation. First, it divides the scene into regions, and then it predicts the regions that the pedestrian may head toward. Second, it chooses a trained LSTM to predict a path to all possible destinations. Xue et al. (2017) think the division of regions is an important process in the prediction of pedestrians' trajectories. The advantages of their method are the use of the bi-directional LSTM for the first time to predict people's paths, and the suggestion of more than one prediction for each possible region that the agent may move to.Alajlan et al. (2021) suggested using the cone of vision for pedestrians that had been used before Alajlan et al. (2020) in the combination of neural networks and genetics algorithms (NN-GA) to specify the nearest people in the field of view for each agent. The tracking of these agents' several previous locations in the cone of vision were used as inputs for the LSTM-Directions. They compared the results of LSTM-Directions with the results of NN-GA using many scenarios, and more than one behavior of the moving agents.

2.3.3 *Factors Inhibiting Accurate Forecasting*

Predicting the future trajectories of pedestrians in crowded places is valuable because it can help in advancing autonomous driving and robot navigation. However, the creation of effective prediction models remains challenging due to the dynamic nature of human interactions. Additionally, human motion is characterized by intrinsic multi-modality. Accordingly, Shi et al. (2020) proposed a spatio-temporal model to capture the multi-modality of the motion patterns and aggregate information obtained from socially interacting agents. The researchers also introduced a coordinate transformation to represent the relative motion between people and to enable the integration of more factors to represent the typical interaction of people. The authors conducted extensive experiments on the algorithm and established that it can predict diverse scenarios and that it exhibits excellent performance. One of the main advantages of the proposed algorithm is that it considers spatial social awareness and the temporal movements of the agents when predicting trajectories. The consideration of the social awareness and temporal transitions helps to enhance the performance of the algorithm and increase its applicability to different scenarios. The use of the coordinate transformation was also important as it ensured that essential information, such as speed, position, and direction, were considered. However, the algorithm has the disadvantage of failing to consider past information collected over a long period. Additionally, the model requires extensive resources and time during training.

Still, the paper recommends this model due to its exemplary performance. Future studies should attempt to apply this algorithm to different situations to examine its generalizability.

The utilization of deep learning methods to predict the behavior of pedestrians has gained popularity in recent years. Although positive outcomes have been attained, the use of deep neural networks with the LSTM-Direction is a one-dimensional vector, which means that the data of the model has to be a one-dimensional vector. Accordingly, spatial information concerning the pedestrian is destroyed. Therefore, Chen et al. (2020) proposed the utilization of multi-channel tensors to represent information relating to pedestrians. Additionally, the proposed method encompassed representing spatio-temporal pedestrian interactions using convolution tensor operations Chen et al. (2020). The result of this exercise was the creation of an end-to-end fully convolutional LSTM model for encoding and decoding. The model was then trained and tested based on current LSTM-based methods using publicly-available datasets with appropriate video sequences. Based on the findings, the proposed method offers a realistic trajectory-forecasting approach for diverse applications. It also minimizes the displacement offset error. The algorithm presented by Chen et al. (2020) has several advantages. It addresses the problem of pedestrian spatial information destroyed in existing deep neural networks with LSTM approaches. Another advantage is that it minimizes prediction errors, thus improving accuracy. Furthermore, the experiments that were done showed that this algorithm performed better than existing methods. However, the algorithm did not consider the interaction between scene and social models. Considering the behavior of other objects, whether static or not, is imperative to improving the prediction of pedestrian trajectories. The study recommends the utilization of multi-channel tensor data to improve prediction accuracy. Future studies should examine the interaction between humans and other objects and how this interaction influences path trajectories. Moreover, the proposed algorithm should be utilized in practical environments to resolve issues concerning computer vision.

Crowd management also encompasses developing systems that can count the number of people in a building and discover the places where the occupants are located. According to Qolomany et al. (2017), smart buildings today utilize sensor technology and control algorithms to establish the number of occupants in a building. However, this approach is ineffective due to the limitations associated with the technology utilized. Consequently, the authors proposed the use of time series, rather than sensors, to forecast occupants in a location at a given time. The adopted approach makes use of Wi-Fi datasets and trains LSTM time series models and auto-regression integrated moving average (ARIMA) models Qolomany et al. (2017). When applied to a smart-building case scenario, the proposed approach enabled the prediction of the number of occupants at different time intervals and access point levels. Concerning the LSTM, the researchers created the models in two ways. The first method encompassed developing a different model for each time scale. By contrast, the second approach entailed creating a combined model for three timescales. The combined approach was more effective because it reduced the computational resources needed while exhibiting good performance. Additionally, the utilization of the LSTM showed a lower error rate when compared to ARIMA models. One of the major advantages of the algorithms is that they exhibit better performance than the conventional sensor technology in counting people. A key aspect of the models is that they leverage Wi-Fi networks, which are characteristic of the typical building today. Furthermore, the LSTM combined approach was advantageous as it reduced the number of neurons needed, hence improving efficiency. However, the LSTM approach has several weaknesses, including the need for more training data and additional time required to train and run it. Overall, the researchers recommend the application of the combined LSTM method to the problem of counting people within a building. Future studies ought to explore how the combined LSTM method can be improved and applied in practical settings.

Forecasting human trajectory is one important goal in computer vision. Objects such as socially-aware robots must possess the capability to map and anticipate the movement of people to avoid collisions and enhance their safety. The need to forecast human trajectories is particularly imperative in urban scenarios such as shopping malls and streets due to the presence of large, dense crowds in these areas. Consequently, Lisotto et al. (2019) proposed an LSTM-based model that considered three key aspects: the semantics of the environment, human interactions, and past observations. The proposed model consisted of several pooling mechanisms for integrating the three elements, which also defined multiple tensors: semantic, social, and navigation tensors. The model was evaluated in unstructured environments where intentional and unintentional factors influenced paths. Based on the findings, the proposed model exhibited significant performance accuracy in predicting the human paths. The algorithm has several advantages. One of the main ones is that it considers social interactions and context information when modeling paths, which increases the accuracy of prediction. In this study, the researchers included previous information about the scene as a navigation map, which included information about areas frequently crossed. It also considered scene context by utilizing semantic segmentation, hence minimizing the potential directions of motion to the possible paths. Another advantage is that the algorithm performed better than state-of-the-art methods that do not consider information relating to the context. Furthermore, the proposed algorithm minimizes errors and predicts movement even when the scene does not have other people. The main disadvantage of the study is that it fails to evaluate the algorithm using publicly available datasets. Such datasets are complex; they comprise different categories of moving objects and, therefore, provide a better standard for assessing the performance of algorithms. Another weakness is that the algorithm is based on the LSTM, which requires a lot of time and resources to train and utilize. As a recommendation, this study emphasizes the consideration of context information and social interactions when creating models for predicting human trajectories. Future studies should focus on examining the algorithm against publicly available datasets.

Social interactions, multimodal behavior, and scene context are some of the factors affecting the accuracy of trajectory prediction. Creating an algorithm that addresses these challenges is central to the development of effective trajectory prediction systems. Although current research has addressed these elements to a certain degree, the aspect of multimodality is under-examined. This being the case, Kosaraju et al. (2019) presented a graph-based generative adversarial network that predicts trajectories in a better way by modeling pedestrian social interactions in each scene. In addition, the algorithm included a recurrent architecture for encoding and decoding, which was trained to predict the paths of human beings based on the available features.

Furthermore, the researchers formed a reversible transformation between a scene and its noise to address the multimodality of trajectory prediction. The experiments performed showed that the framework outperforms other methods based on the baselines utilized. The main advantage of the algorithm is that it can generate multiple trajectories for every person involved. Additionally, it can generate the trajectories for each pedestrian in the crowd in a multimodal way. Furthermore, the algorithm can predict human trajectories more realistically and outperforms other state-of-the-art mechanisms. Considering the disadvantages, the algorithm did not take into account other objects within the typical scene. Nevertheless, the study suggests using flexible graph-attention networks to improve trajectory prediction. Future studies should attempt to examine the algorithm in practical environments.

Pfeiffer et al. (2018) suggested a model that combines the three important factors in pedestrian prediction: pedestrian velocity; pedestrians' static obstacles in the scene; and information about all pedestrians in and around the targeted pedestrian. Based on LSTMs, they could predict person-to-person interactions, that the pedestrian could avoid a static obstacle, and they introduced a new technique, the angular pedestrian grid (APG) to deal with dynamic obstacles. Yang and Peters (2019b) focused on the group's issues. One of the shortcomings in the field of social robotics is assuming that the members of the collection are constant during generating reasonable paths for robots. To overcome this limitation, Yang and Peters (2019b) suggested a new method by using LSTM-based Generative Adversarial Network (GAN) with group interactivities that combines the agent's position and his/her head orientation in the collection in order to generate reasonable paths for approaching a group. The datasets used were synthetic. App-LSTM is a refurbished LSTM to produce reasonable paths for approaching a group, and was suggested for use by Yang and Peters (2019a). By taking into consideration positions and orientation, and using a synthetic dataset, they designed a group interaction module (GIM). The module's current state of an agent that is approaching is repeatedly expurgated to obtain an accurate concentration.

The computation of human trajectory prediction requires the consideration of both temporal and spatial interactions. However, based on this study, most of today's proposed methods fail to consider temporal correlations, especially associations between pedestrians in each scene. Accordingly, the authors proposed STGAT (spatial-temporal graph attention network) to predict pedestrian trajectories along with a sequence-to-sequence methodology Huang et al. (2019). The algorithm also includes an LSTM to encode temporal interaction correlations. The authors evaluated this sys- tem and established that it performed well against other methods using two publicly available datasets. They also emphasized that the system produced more “socially”possible pedestrian trajectories. The proposed algorithm has the main advantage of considering temporal correlations among pedestrians. Indeed, these correlations provide insight concerning the potential movement of the person. Furthermore, the algorithm exhibited better performance than other state-of-the-art methods. Finally, the algorithm produced “socially-acceptable” predictions of future movement and can be applied to different scenes. On the other hand, the algorithm is based on the LSTM and, therefore, requires more time and resources to train. In terms of recommendations, the authors emphasized the utilization of the STGAT for predicting future human movement in crowded spaces. The main strength of this method is that it considers temporal aspects that improve the accuracy of prediction. Concerning future work, it is imperative to explore how the algorithm can be improved to enhance its efficiency. Additionally, future work ought to consider moving objects within the scene.

This study proposed an approach for perceiving pedestrians as clusters to predict their dynamics by using a neural network. By regarding people as clusters, the authors were able to model a system using cluster features rather than agent features Yokojima and Sakai. Essentially, the maximum number of individual agents is limited by the size of the neural network. The authors also regarded the training model at a cognitive level to distinguish phenomena learned at the current level of cognition from others that require additional surrounding details. The study demonstrated this methodology by using a neural network and leveraging the analogy of cognitive modes. The major advantage of this algorithm is that it perceived pedestrians as clusters rather than agents. It also made use of cognitive modes as human beings tend to move from one mode to another. Additionally, the researchers attempted to reduce the computational intensity of the LSTM used for encoding, which can enhance efficiency. However, the algorithm was disadvantageous mainly because the authors failed to examine it empirically. As a result, it was difficult to determine whether the algorithm performed well as compared to other methods. Overall, the study recommends the use of abstracted features and the conceptualization of crowds as clusters rather than agents to improve pedestrian learning.Future work ought to examine how this algorithm, which is based on cognitive modeling, can work on publicly available crowd datasets.

Humans are complex and, in most cases, are not predictable. The unpredictability is that they can change their minds anytime, making it impossible to predict the actual location at a given time based on the history of their movements. When humans change directions suddenly, whether due to a change of plans or to avoid an obstacle,the training model can be confused. To solve the problem, pedestrian positions are measured related to the coordinates of the movement history Choi et al. (2019). The information of the pedestrian movement is needed as a feature vector. The long short-term memory network (LSTM) does the encoding. The LSTM performs time- series processing efficiently since it is a recurrent neural network algorithm. Multi- layer perception (MLP) makes it possible to have different layers to operate. The pedestrians, the neighbors, the velocity, and the obstacles are considered to provide a larger view for the whole approach. The trajectory prediction will be more accurate because we linearly combine all past motion vectors. In conjunction with the New York Grand Central dataset, PyTorch was used for training and testing. The data was divided into flames; 10 flames were used, five used as input, and the next five as the output (prediction). The advantages of using this algorithm are that the data used is actual data and validates the results. Combining the LSTM and MLP makes the algorithm more accurate as it enjoys the advantages of the two algorithms. The disadvantage of the algorithm is taking into consideration the prediction of human trajectory considering the unpredictability of humans. Plans for the future involve the model's being trained using human movement without trying to predict their movement; instead, they will learn how to make their movements and avoid obstacles by altering their velocity or direction.

A detailed table summarizing all papers, which includes overall approaches, the objectives of each method, and the metrics used in each technique, can be found in the appendix.

2.4 CONCLUSION

This chapter is a literature review of about using LSTM to predict pedestrian's next locations. Many criteria have been applied to this survey to concentrate on a specific path in this field. Heterogeneous environments have been excluded from this survey, and we focused on the environments that deal only with pedestrian surroundings. Also, for this review, the papers that were published before 2017 were eliminated. These papers form many categories. Table 1 includes a summary of the title of the paper, method, objective, and the metrics that were used in the paper. There is a speedy evolution in the field of using machine learning, and deep machine learning for the estimation of pedestrians' future steps. The information about datasets is still less than desired despite the everyday presence of crowds worldwide We believe the tools of collecting the data of agents in the most crowded areas have improved, but there is still a gap between the hope and the on the ground reality regarding datasets.

CHAPTER 3

Using Neural Networks and Genetic Algorithms for Predicting Human Movement in Crowds[2]

3.1 SUMMARY

Safety is an important issue at large gatherings of people such as at religious gatherings or sporting events. Therefore, it is important to control crowds, and identify in advance when dangerous situations may arise. Simulations play an important role in predicting how people in crowds will react to each other and their environment. Simulations often rely on *a priori* models of human behavior to predict crowd behavior. We combine, Genetic Algorithms and Neural Networks to learn how people behave in crowds to avoid assumptions used in *a priori* models. We examine learning in two important regimes, structured crowds where individuals are moving in a specified direction, as in the Islamic Hajj or Hindu Kumbh Mela; and unstructured crowds, such as town squares and train stations. In this preliminary work we are most concerned with questions of trainability. In order to provide sufficient data and control qualitative features of our crowd data we begin use generated data based on elaborations of wildlife flocking models in NetLogo. We compared performance on structured and unstructured crowds by predicting a series of next locations. The results showed we are able to predict crowd motion, but error rates for individuals grow as time passes; however, the structured crowd gave more reliable results than the unstructured crowd.

3.2 INTRODUCTION

Simulations of crowd behavior can be used to improve the design of public spaces for the movement of people at large events, such as religious gatherings, athletic

[2]This chapter was published in the 22nd International Conference on Artificial Intelligence. Springer.

events, or concerts. This, in turn, can be used to help prevent crushing and trampling disasters and help guarantee public safety. Many simulations rely on assumptions about how crowds move, leaving a possible predictive gap between reality and the simulation. Our approach is to use machine learning to learn how individuals in a crowd move and provide a simulation more directly tied to an unbiased observation of crowd behavior. We examine learning in two common and important regimes, **structured crowds** where individuals are moving in a specified direction, as in the Islamic Hajj or Hindu Kumbh Mela; and **unstructured crowds**, such as town squares, shopping malls, and train stations.

Neural Networks (NNs) have a track record of success in problem areas such as speech recognition, language translation, and image/video classifications Chan *et al*. (2016); Bahdanau *et al*. (2014); Shao *et al*. (2017). Genetic Algorithms (GAs) have been shown to work well at training Neural Networks (NNs) Wang and Wang (2017). In this paper, we use GAs to train NNs to predict the movement of people in a crowd. The object of this paper is to predict the next location of person in a crowd by taking into account their nearest neighbor within their cones of vision.

In the next section we show the background organized into three subsections. This is followed by a section on how our model works. The fourth section presents the results, and the discussion. Finally, the conclusions, and future works are presented.

3.3 BACKGROUND

Predicting individual trajectories in crowded areas for use in crowd management simulations is an evolving research topic.

Alahi *et al*. (2016) show that they were able to predict future trajectories by using an individual's past positions and a social Long Short Term Memory (LSTM) model trained on observed human movement. To discover the motion of each pedestrian and how that person interacts with nearby neighbors Xu *et al*. (2018) used a Crowd Interaction Deep Neural Network (CIDNN) to predict displacement frames for each pedestrian. The DESIRE encoder and decoder Lee *et al*. (2017) predicts future locations of objects in dynamic scenes. Walking-step size prediction, using Genetic Algorithms to optimize a Neural Network model, predicts the step size by collecting data from different sensors that have been applied to the pedestrians Wang and Wang (2017).

Based on observing the mobility behavior for a person over a period of time, using the Mobility Markov Chain (MMC) model can lead to a prediction for the next location Gambs *et al*. (2012). The use of this model Karasev *et al*. (2016) improves long-term prediction by modeling pedestrian behavior, as a jump-Markov process. Using multi-layer architecture (IaKNN) Interaction aware Kalman Neural Networks for forecasting the motion of surrounding dynamic obstacles can solve high-density traffic issues Ju *et al*. (2019).

Different approaches have been applied in Yang *et al*. (2006); for example, NN and GA have predicted the number of occurrences of dwelling fires in the United Kingdom. Plans to apply deep Neural Network Rehder *et al*. (2018) have been proposed to predict a pedestrian's trajectory using Goal-Direction Plan, and the learning patterns motion behavior will be operated with Fully Convolutional Network (FCN).

3.3.1 *Structured/Unstructured Crowds*

Analyzing a crowd's behavior can be used to improve the designs for the movement of people at large events, such as religious gatherings, concerts, or sporting events, in order to prevent improve public safety Johansson *et al*. (2008); Krausz and Bauckhage (2012). As noted in Rodriguez *et al*. (2009), there are two types of crowds: the **unstructured crowds**, in which people move in a variety of directions as in Figure 5.1(a); and, the **structured crowds**, in which people tend to move in a specific direction toward a target as in Figure7.1(b). Large events or festivals, such as Züri Fäscht in Zürich, Switzerland, deployed an app for crowd management Blanke *et al*. (2014) over a period of three days. An app with a similar purpose has also been proposed by Yamin *et al*. (2016) for crowds at the Hajj (an Islamic ritual).

A proposed RFID-based Hajj management system Al-Hashedi *et al*. (2013) would include data sharing, network communication, and Mohammad and Ades (2009); Yamin *et al*. (2008) other wireless technologies. To improve the research on a crowd's motion when there is a great density of people, Yamin (2008); Nasser *et al*. (2017) suggest a framework for Hajj management. Schubert and Suzic (2007) describes the development of a decision-support system for crowd control that uses a GA with simulation to discover control strategies. Distributing Combining GPS and Bluetooth Low Energy (BLE) tags among groups of people and using smart phones Jamil *et al*. (2015) Jamil et al. hope to capture large-group dynamics for large-scale events. Pellegrini *et al*. (2009) has developed a crowd simulation to model dynamic social behavior that has been trained from birds-eye video records of high traffic areas. Koshak and Fouda Koshak and Fouda (2008) have used GPS and added GIS to capture and analyze pedestrian movement. Schubert *et al*. (2008) have described a decision-support system by storing sample situations and then used GAs to run trials in order to find a successful system to control crowds. In this section, we compared more than one crowd style by emphasizing structured crowded areas and unstructured crowded areas for the purpose of identifying the types of crowd motion.

3.3.2 *Evolution Models NN & GA*

Lately, NNs have had a high impact with accomplishments in many areas. Chan *et al*. (2016) demonstrates NNs ability to learn LAS (Listen, Attend, and Spell) by duplicating an audio sequence signal into a word sequence. Regarding translations, Bahdanau *et al*. (2014) used NNs to predict relevant translatable words. Also, NNs are valuable for classified tasks for videos and image processing Shao *et al*. (2017). Wang and Wang (2017) has shown that the NN model developed by GA produces better results for prediction. Gupta and Sexton (1999) has compared the genetic algorithm with backpropagation for neural network training, and has demonstrated that GA is superior to backpropagation. The methodology in our paper is based on the combination of the Neural Networks and genetic algorithms.

3.3.3 *Simulating Crowd Interactions*

In our model, the collected dataset that applied unwritten rules in the crowd, such as avoiding collision, depended on the Netlogo model as an important machine for simulating people in different approaches when seeking crowd behavior motion. Agent-based models have been an attractive research tool for people seeking crowd

(a) An unstructured crowd.Image from Ozturk *et al*. (2010).

(b) A structured crowd. Image is a screen capture from videos in 2019 from Ministry of Hajj, Kingdom of Saudi Arabia, https://www.haj.gov.sa/en

FIGURE 3.1: Examples of two major regimes of nominal crowd flow.

motion outcomes to evaluate structured public areas and closed spaces. The hope is to show the effectiveness of simulation in crowd management.

For instance, Pluchino *et al*. (2013) presents a simulation using a NetLogo model, for pedestrian motion at the Castello Ursino Museum in Catania, Italy. The simulation is used to evaluate the capacity of the museum and the safety of visitors in cases of an alarm. Camillen *et al*. (2009) compares their evacuation approach with different evacuation approaches looking for an optimal solution. Their results show the efficiency of their evacuation plan by uncovering hard forecasts in emergency results. Based on experts on animal-migration, Hargrove et al. Hargrove and Westervelt (2012) simulated the efficacy of a PATH (Pathway Analysis Through Habitat) by using NetLogo to study animals moving outside their territory through a connected but fragmented landscape. It is clear that NetLogo as a very useful agent based modeling tool for research and teaching Tisue and Wilensky (2004).

3.4 OUR MODEL

In a crowd, people usually take into account their nearby neighbors in order to avoid a collision. We model a person's vision by their **field of view** or we say **cone of vision**. In our 2D world a cone of vision is the region in front of a traveling person subtended by an angle on either side of vector of travel. See Figure 3.2. Only people visible to the person can act as a influence on the direction the person is proceeding. For example, a person will take an alternative route or will stop, due to the influence of the people who walk into his/her cone of vision.

We created a model for the prediction of pedestrian movement by locating nearby people in an individual's cone of vision. The location of the nearest people are fed into our NN and a predicted direction is returned.

Because this is a complex system, the farther into the future we attempt to predict the more inaccuracies will build and the model strays from actually tracking the location of individuals Pellegrini *et al*. (2009). We will use a NN given the list of three nearest people in the cone of vision sorted by distance. The NN will be trained by a GA.

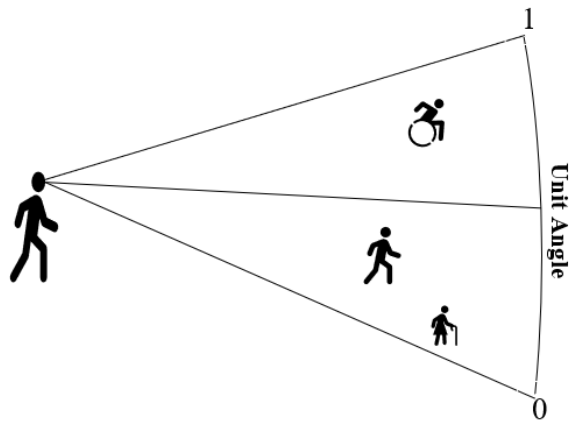


FIGURE 3.2: In our model, the NN focuses on the nearest agents in the individual's cone of vision to predict his/her next position. The position of the nearest three agents are represented as a distance and angle. The angle is scaled between 0 at the right side of the cone of vision and 1 at the left side. The scaled angle we will call the unit angle. Distances are absolute.

3.4.1 *Hand Collected Dataset*

Our goal is devise a method of predicting where individuals will move in a crowd given observations of crowds. This will require lots of data in form of (x, y) pairs and timings to train on. We decided that a low cost solution is to create a separate data generating simulation for our initial development.

We use something similar to a Social Force Model (SFM). An SFM is based on three factors: 1) the acceleration of an individual to a desired speed; 2) the individuals maintaining a specific distance from obstacles and other individuals; and, 3) the impact of attraction, as in Helbing et al. Krausz and Bauckhage (2012). Our data generating crowd model uses a NetLogo model. The model is derived from the flocking model that comes with NetLogo, but with some additions for variation in speed, collision avoidance, field of view, boundary and initialization conditions, and for structured crowds and common direction of flow. Both structured and unstructured crowds were simulated as seen in Figure 3.3.

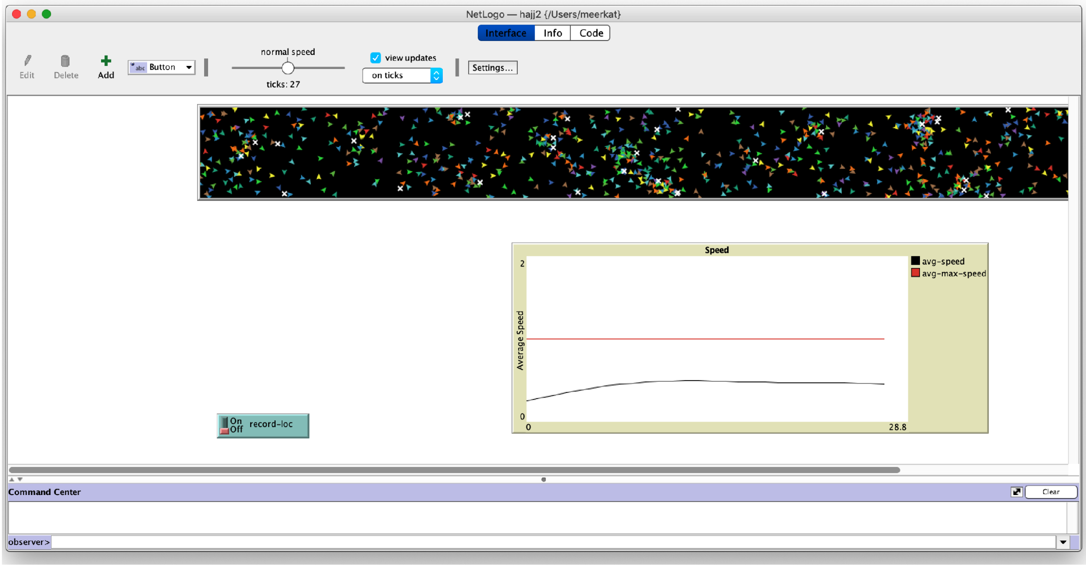
3.4.2 *Combining Neural Networks with GAs*

Crowds may behave differently depending on the cultural composition, event, or environment. In order to more accurately model crowds it is important to learn from observation rather than apply a one size fits all solution. This motivated us to design a method that can read and learn from data. Perceiving the patterns of data (such as nearby neighbors' positions) to predict the motion of a crowd is the main feature of the algorithm. The combination of NN and GA has produced excellent results Wang and Wang (2017). This motivated us to develop an algorithm using NN and GA. In a crowd people pay attention to the people in front of them and "sort" them visually as nearest neighbors, which becomes a major factor when making a decision to change direction, even if only a slight change. This logic encouraged us to apply it in designing the inputs to the NN.

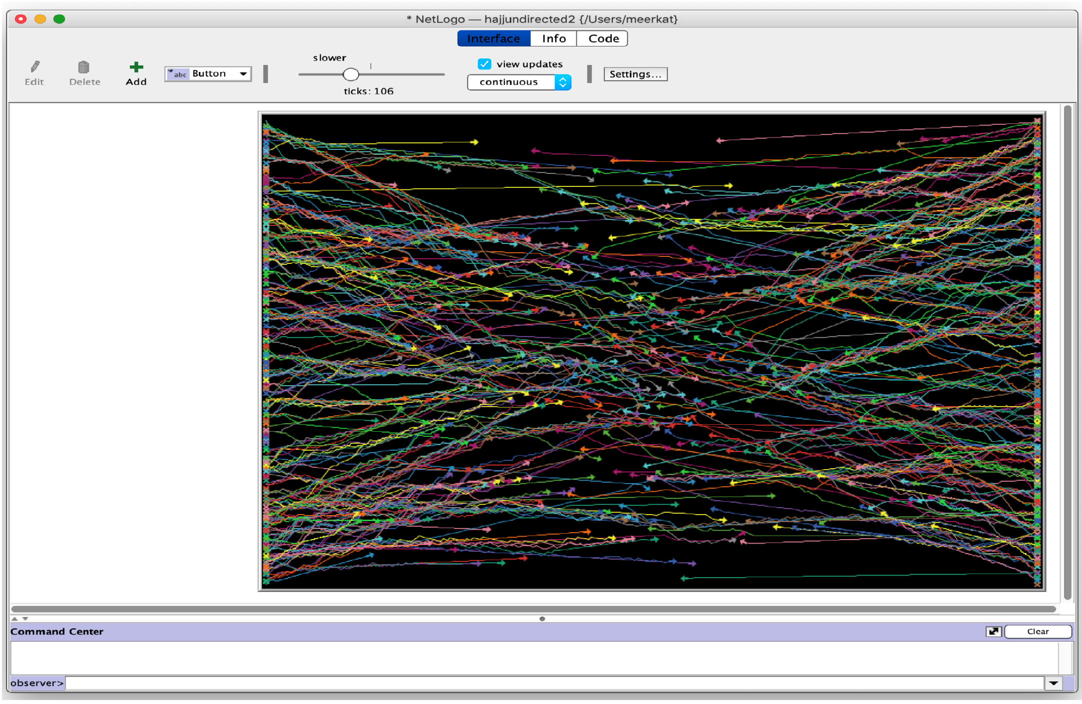
After calculating the distance between points, the neural network takes the three nearest neighbors' positions in the cone of vision as inputs. For example, if we specify that 60 degrees is the cone of vision for the individual, the distance is calculated as in Equation 7.3

$$nearest3(c) = \operatorname*{argmin\,3}_{i \in Cone_c(Pop)} \text{distance}(c,\ i) \tag{3.1}$$

In this Equation, c is the current pedestrian for whom we are looking for his/her nearest neighbors. i is from the set of individuals in the cone of vision of c denoted $Cone_c(Pop)$. The cone of vision is determined by the direction of motion of c. We assume a cone of vision is 60 degrees. argmin 3 gives the list of three smallest values, in this case, distances. Finally, in our unit angle, every individual from those three nearest neighbors obtains a number between 0 and 1 based on his/her location in the unit angle as in Figure 3.2.



(a) Structured crowd simulation in which people tend to move from left to right.



(b) Unstructured crowd simulation in which people on the sides move at random initial top speeds to a paired random location on the opposite side. This forces the left and right to negotiate passage through the middle area by adjusting their direction and speed.

FIGURE 3.3: NetLogo simulations of structured and unstructured crowds.

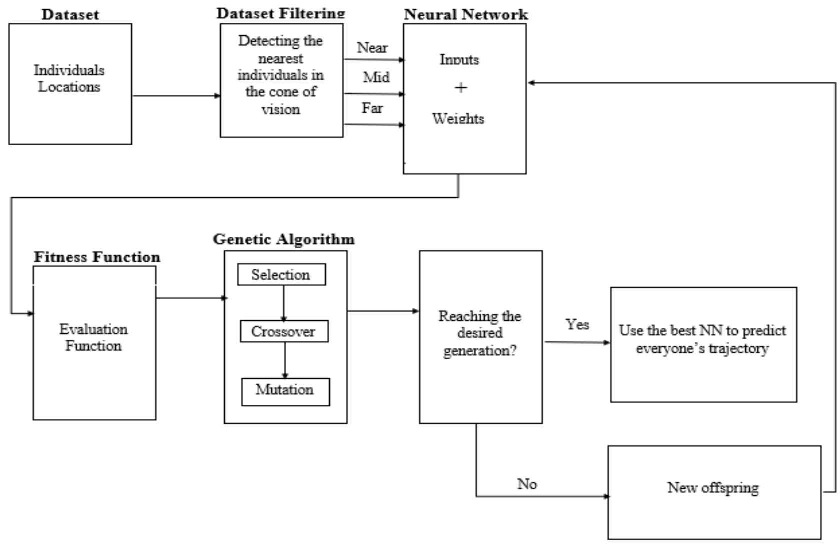


FIGURE 3.4: This shows a sketch of the workflow for training the NN. Training data based on NetLogo simulations is input to the NN which produces trajectory predictions for the fitness function. The fitness function is used in the GA to converge on better weights for the NN.

3.4.3 *The Use of the Genetic Algorithm*

A Genetic Algorithm (GA) are an optimization algorithm inspired by biological evolution Eiben *et al*. (2003). A GA has several key components. A population of potential solutions. Each individual in our population is a possible set of weights for the NN. Imperfect copies of the individuals from the population will be made using cloning and mixing using mutation and crossover operators. Selective pressure will be applied to force an enrichment in the population with sets of weights which are selected for, that is, have higher fitness. In our case fitness will be determined by the success of the NN in predicting where individuals move given what they see (More below). If selection is controlled by the fitness function so that individuals with higher fitness function values are selected, then we have an optimization algorithm using a scheme very similar to that envisioned by Charles Darwin (1964). Each generation of the genetic algorithm uses tournament selection to choose the worthy individuals among the population as parents and uses them to produce offspring for the next generation using mutation and crossover. For diversity in every generation of the

Table 3.1: Table of GA parameters

Parameter	Value
Type of GA	steady state
Pop size	100
Mutation Rate	0.12 per weight
Mutation	Add random $N(0,1)$
Crossover probability	100%
Crossover Type	1pt
Mating Selection	Tournament size 10
Stopping Criteria	1000 generations

population, 12% is the percentage of mutation applied. To optimize the quality of NN weights, each individual corresponds to an evaluation, which is the distance between the predicted location and the actual location. The fitness distinguishes the NNs with high/low scores based on their outputs. It calculates the difference between the predicted position and the actual position for every individual's next location. Table 3.1 describes the Genetic Algorithm parameters used in this paper.

$$fitness(NN) = \sum_{i=0}^{n-1} |\alpha_i - \pi_i| \tag{3.2}$$

In Equation 3.2 $fitness(NN)$represents the fitness for the weights of the NN. The variable i indexes through all the training data of position and three nearest neighbors. α_i is the actual angle of the next move of training case i. π_i is the angle predicted by the NN. Angles are in unit angles. The less the difference between the angles, the better the neural network. That is, the GA is minimizing the sum of simple errors of the trajectories.

In summary, the GA begins by initializing with a random population of neural networks. The GA uses the fitness based the nearest neighbors in the cone of vision for each agent. This is used to train the weights for the neural network using GA. The fitness function scores each neural network based on the difference between the predicted trajectory and the actual trajectory in terms of a unit angle in the cone of vision. The neural network with the smallest error will be the model's neural network

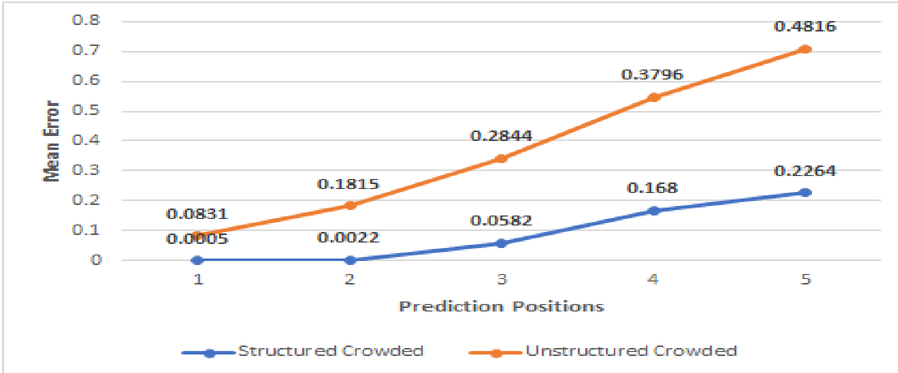


FIGURE 3.5: The Error in the average distance between the predicted location and the actual location.

to predict an individuals trajectory. The step size will be assumed to be the same as the last step size. Figure 3.4 explains the workflow of the training.

3.5 RESULTS

In this experiment, we produced training data using NetLogo in two scenarios representing the two crowd types: structured and unstructured. We used NetLogo since it is an agent-based model that is well known in research Pluchino *et al.* (2013); Camillen *et al.* (2009); Hargrove and Westervelt (2012); Tisue and Wilensky (2004). To model structured crowds, we observed surveillance cameras that were deployed on Hajj 2019 to monitor the behavior of people. A model was then built based on flocking/herding in which agents move within the limits of their own speed to move together and yet not collide or get too close and proceed toward a goal. The model is parameterized to emphasize the distributions of maximum speeds and how close they can get before they feel the urge to separate. There is a cohesion factor as well in that people in a crowd tend to move with others much like a herd. Unstructured crowds were modeled similarly but individuals were initially positioned on the left and right of the arena (see Figure 3.3(b)). They then proceed at different initial maximum speeds to cross to paired target points on the other side. This way the two sides must negotiate to slide between the opposing moving people. Decisions to avoid collision with others is the

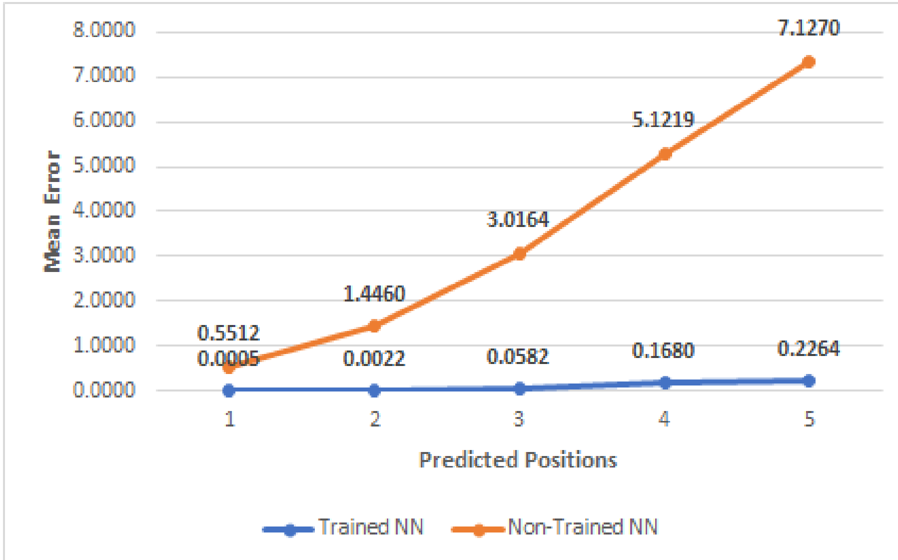


FIGURE 3.6: The mean error between the predicted location and the actual location for an structured crowd. Both trained and untrained neural networks are shown for comparison.

most important feature of the NetLogo models. This is done by the agents deciding to change their speed and direction.

3.5.1 *Trained vs Control Neural Network*

To answer the question are we able to learn how to predict the trajectory we compared the output of the NN we trained to the a control. As a control, we chose a random NN without training. We predict the trained NN should actually be able to move more like real people in a crowd.

Figure 3.6 and 3.8 show the results for both experiments, trained and control, with the same test for structured and unstructured crowds respectively. The graph shows the mean Euclidean distance between the predicted position and the actual position assuming the step size for all predicted steps as the step size for last step before starting the prediction.

The graphs support the idea that a GA can be used to train a NN to model people's behavior in a crowd.

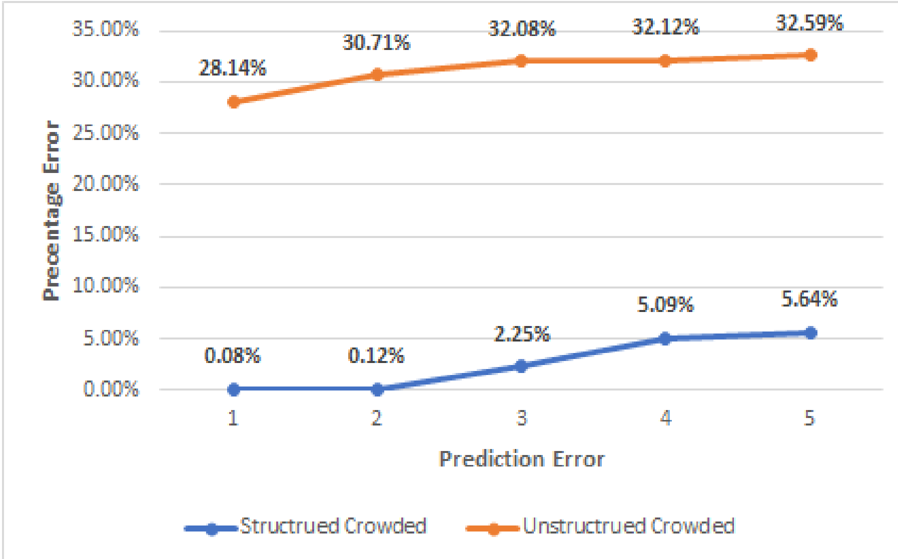


FIGURE 3.7: The chart shows Cumulative Distance Error (CDE) for all the next prediction positions. Since we have trained the NN for the first position, the result shows how the error increases as time passes. That is, error accumulates between the predictions the model is giving and the actual directions that individuals turn.

3.5.2 *Time Series Use of NN*

Our next question is whether the NNs we are generating can be repeatedly applied in a simulation to predict crowd movements farther into the future. In our experiment, we have applied two metrics, the cumulative distance error (CDE) and the mean in two different types of crowds.

The cumulative distance error (CDE) for position 1 is the average distance between the current position and the predicted position, divide by the average distance between the current position and the previous position. It then multiplies that result by 100 to obtain the percentage error rate for the first predicted position. For the CDE of the second predicted position we calculate the distance between the true position and the predicted position, except that it divides by the average distance between the current position and the position from which we started the prediction. We proceed like this for the remaining predicted positions in our experiment. The results for CDE are displayed in Figure 5.6, which shows how the error rate grows as time passes.

Figure 3.5 displays the mean for each distance between the predicted position and the true position. The mean calculates how the error in the average distance between the predicted position and the true position increases as time passes.

Both graphs indicate a divergence between predicted and real locations. This suggests a simulation based on our learned NN will quickly diverge from reality. While this may at first appear as a simple negative result, the system we are modeling is a complex system with classic problem that it small errors will accumulate exponentially. We would expect these graphs. But since this is our initial research we believe these measures and others may help to greatly improve our ability to predict and stave off the inevitable divergence.

3.5.3 *Discussion*

One of the innovations of our work is the use of a sorted list of nearest neighbors in the cone of vision. We believe this mimics cognitive input to the individuals. Even in structured crowded areas, Rodriguez *et al.* (2009) people walk around each other as they walk in the same direction to the same goal. However, we realize that the unstructured crowded area is assembled differently, where people randomly cross each other's paths, and still humans tend to watch out for his nearest neighbors and make a decision about his/her next position or direction.

For the comparison between the structured and unstructured crowds in our experiment, the results include the cumulative distance error, the mean. The structured crowd results have smaller errors of prediction than unstructured crowds. We believe this is because the behavior is much more predictable in structured crowds than unstructured crowds.

CONCLUSIONS

Simulating crowd motion and predicting the individuals' movements can be used to design crowd areas and to improve safety criteria. The need for these kinds of simulations inspired us to design a simulation that shows the reaction of people in an environment to seek a predictable crowd behavior in order to provide safe

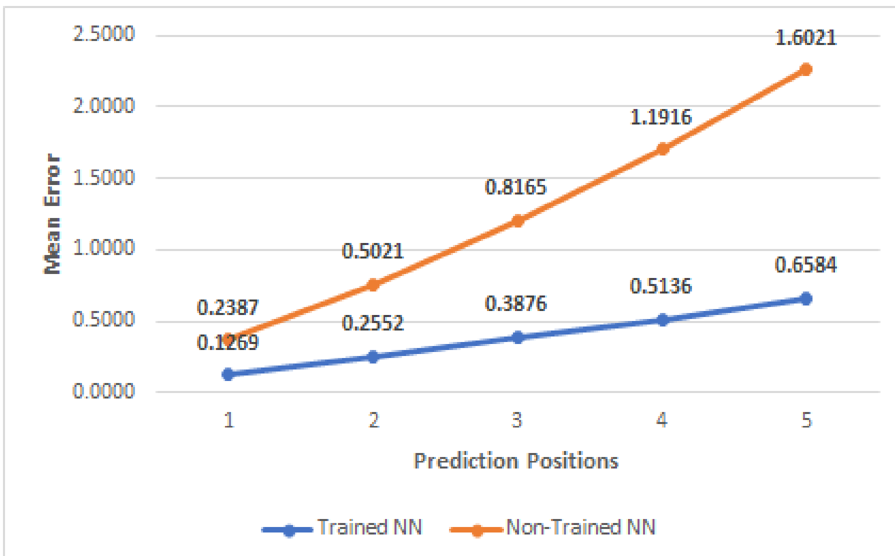


FIGURE 3.8: The mean error between the predicted location and the actual location for an unstructured crowd. Both trained and untrained neural networks are shown for comparison.

and pleasant experiences for participants. We have shown that the combination of neural networks and genetic algorithms can be effective for predicting movement in a crowd. We used the first predicted position for individuals to train the neural networks, and the genetic algorithm to obtain the best NN. One of our innovations in our NN design is use of a sorted list of nearest neighbor locations in the individual's cone of vision. We believe this method shows promise as a technique for learning the movement of people in a crowd. The results showed a more reliable outcome for a structured crowds than an unstructured crowds. Our future work, will involve diversifying tests, improving the evolutionary algorithm, and measures of crowd movement. Additionally, the agents of the model will not be limited to pedestrians, but include other agents, such as cars, bicycles, etc.

CHAPTER 4

Machine Learning for Dense Crowd Direction Prediction Using Long Short-Term Memory[3]

4.1 SUMMARY

The safety of a dense crowd is one of the most important matters for an event's organizers. Therefore, management of the crowd, and noticing any serious issues in advance becomes important. Developing a crowd simulation by using a social force model simulates the behavior of crowds in reality. The prediction of individual agents' behavior in the simulation and how the agents interact with each other can improve the safety of dense crowds. Depending on the success of Recurrent Neural Network(RNN) handling of sequential data, we propose a model that is based on Long Short Term Memory (LSTM) to predict individual agents' next locations. Our proposed approach will be tested two different densities of crowds, structured crowds, and unstructured crowds. In structured crowds, people generally move in one direction and head to the same destination, such as at the Islamic Hajj. In unstructured crowds people move in many different directions and head for different destinations, such as in public town squares.

Crowds, Structured Crowded Area, Unstructured Crowded Area, Machine Learning, Recurrent Neural Network, and Long Short-Term Memory, "Short Paper" Symposium on Artificial Intelligence (CSCI-ISAI).

4.2 INTRODUCTION

People usually follow rules that are taken for granted when they walk in crowds. For instance, in a dense crowd, individuals move to their next locations while avoiding people or obstacles in front of them. Understanding these rules leads us to avoid

[3]This chapter was published on 2020 in the International Conference on Computational Science and Computational Intelligence .

(a) Example of an unstructured crowd.Image from Ozturk et al. Ozturk *et al*. (2010).

(b) Example of structured crowd. Image is a screen capture from videos in 2019 from Ministry of Hajj, Kingdom of Saudi Arabia, https://www.haj.gov.sa/en

FIGURE 4.1: Examples of two forms of crowds.

dangerous situations and maintains the safety and stability of crowds. Rodriguez *et al*. (2009) has shown that there is one type of crowd motion in a structured crowd, such as at the Hindu Kumbh, and another type of crowd motion in an unstructured crowd, such as in subway stations. Most prediction simulations have been done without differentiation between the types of crowds. Our proposed approach is to employ machine learning to understand how people behave in the two different types of crowds, structured, and unstructured.

Recurrent Neural Networks (RNN) and especially Long Short-Term Memory (LSTM) networks have become a very popular method to understand the sequential nature of their inputs. LSTM has shown promising results in problems with sequential data, such as individuals' trajectories, vehicle motion, handwriting, and speech. Alahi *et al*. (2016) presented how to connect LSTM networks for every trajectory in relation to each other, which in turn, allows every LSTM network to share information with close networks. Rodriguez *et al*. (2009) has done experiments that depend on two types of crowds (structured and unstructured crowds) based on the closest people in the cone of vision. Prediction of an individuals' movements could be improved if we take into consideration the differences in crowd behavior in each type of crowd. This will make it easier for machine learning to accurately learn each kind of movement. Focusing on the crowd type to acquire data is a form of classification. Our proposed model is an extension of Alajlan *et al*. (2020), which use a cone of vision to specify the direction of motion based on the closest people; furthermore, employs the LSTM networks technique to monitor the previous cone of vision direction for every individual in order to understand each individual's behavior. Additionally, the crowd results will be divided into two categories (structured, and unstructured crowds).

We present related work in the following section. This will be followed by the Methodology and Datasets sections. Finally, we will present the conclusion of our proposed model.

4.3 BACKGROUND

Based on the past locations, Alahi *et al*. (2016) used the LSTM model to train their model to predict the humans' trajectories. They let the LSTM network join and share information with LSTM networks in its range. Shi *et al*. (2019) suggested LSTM networks that use encoding and decoding, which in turn, encode movements and interactions for a long sequence of predictions. Gupta *et al*. (2018) suggested Generative Adversarial Networks that use encoding and decoding structure to predict future paths and avoid the existence of more than one prediction. Manh and Alaghband (2018) shows two models of Scene-LSTM that can predict human motion; it presented how the information from the scene is important by feeding it to the cells, which in turn, uses only the useful data to forecast next movements. Xue *et al*. (2017) uses the Bidirectional LSTM to class people's destinations, which in turn, improves prediction precision.

Necessary crowd safety, such as at religious gatherings, concerts, or sporting events, can be improved by analyzing crowds behavior and improving the designs of crowd movement at large events Krausz and Bauckhage (2012); Johansson *et al*. (2008). Crowd behavior and movement have been defined as two types: structured crowds, where people are heading in specific directions; and unstructured crowds, where people's directions cross each other Rodriguez *et al*. (2009). Yamin proposed an app for crowd management Yamin *et al*. (2016) at the Hajj (an Islamic ritual), where most of the crowd formed as a structured crowd. In Switzerland, there is a festival that takes place at Züri Fäscht in Zürich Blanke *et al*. (2014), deployed an app for crowd management with sign points for each booth that must be collected by the visitors. The RFID-based Hajj management system proposed by Al-Hashedi *et al*. (2013) to manage crowds used data sharing. In addition, Mohammad and Ades (2009) managed the crowd by using RFID with network communications. Since the Hajj is the largest religious gathering of people and behaves as a structured crowds, Yamin *et al*. (2016) proposed integrating social media and mobile apps for Hajj management. A framework has been suggested for Hajj management by Yamin (2008) to improve crowd motion research. He proposed a framework for monitoring hajjies(people who practice the Hajj ritual) upon their arrival at the airport, which is the starting point of their ritual participation. Additionally, Nasser *et al*. (2017) proposed a crowd monitoring and management framework for the Hajj gathering.

4.4 METHODOLOGY

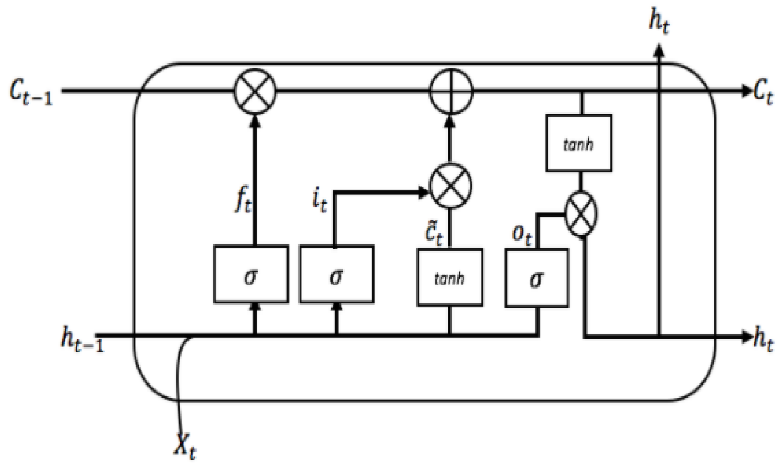
Dense crowds usually form particular patterns depending upon the crowd type. According to Rodriguez *et al*. (2009), one direction and one goal is the pattern formed by a structured crowded, and different directions and different goals is the pattern of an unstructured crowded. Figure 7.1(b) and Figure 5.1(a) show examples of how structured and unstructured crowds. In a structured crowd, people usually maintain their direction in a particular pattern but adapt the path they use. By contrast, in an unstructured crowd, people create more than one pattern, but usually maintain their direction in a specific manner. The idea of an individual's "cone of vision" was applied in Alajlan *et al*. (2020), and emphasizes determining the three closest people (near, mid, and far), then letting machine learning learn how these closest people in the cone of vision might impact the decision of an individual's choice of path. Figure 4.3 shows how Alajlan *et al*. (2020) uses the "Unit Angle," which numbers the neighbors in the cone of vision. In contrast, LSTM has been used to predict people's trajectories based on past positions, such as in Alahi *et al*. (2016).

4.4.1 *Overview on the LSTM*

To enhance the performance of RNN to alleviate the difficulty of learning a long data sequence, LSTM has been suggested to fix the problem of vanishing and exploding gradients. Regarding vanishing means, the gradient tends to be smaller and smaller when we return to the earlier layers. By contrast, when we consider the exploding means, the gradient tends to get larger and larger when we go back to the earlier layers. In general, that may mean there is no problem with a small number of hidden layers, but it may cause an unstable situation when we deal with a large number of hidden layers. Some of the LSTM advantages are remembering data for a long time, and predicting more precisely sequential information based on previous data. By using three gates (forget gate, input gate, and output gate), the idea of an LSTM network can be formulated. Figure 4.2(a) illustrates how an LSTM network functions in the LSTM modules, and figure 4.2(b) illustrates how the LSTM network can be detailed in the equations Varsamopoulos *et al*. (2018). f represents the forget gate, i is the input gate, and o is the output gate. W represents weight matrix of input; U represents weight matrix and recurrent link between the past and running hidden layer. h is the hidden state of the past time-step. $\hat{C}_t$ is the candidate for the cell state that is calculated based on the previous hidden state and X_t (the input of the current time-step), while C_t represents the cell state at the current time-step.

4.4.2 *problem statement*

Our goal is the accurate prediction of the people locations from a sequence of data based on people's walking behaviors. The challenges always depend on the accuracy of the prediction for pedestrian trajectories in compare between the actual location with the predicted location. Our proposed model is based on two main factors: 1) the direction that the individual decides for his/her path results from the sequence of previous steps; and 2) the velocity which can estimate the final predicted location over time is based on the average of the previous speed. In our proposed model, we picture the behavior as a drawn pattern. By taking the notion of the field of view, we will draw the sequence of steps based on the Unit Angle, shown in Figure 4.3, for each individual. In other words, every person will have a sequence of data, each datum represents a value between 0 and 1 that denotes the direction of his/her cone of vision, which in turn produces a value for each step in the sequence. The velocity for every individual is an important matter in our prediction model in order to calculate distances more accurately. The prediction for speed in Alajlan *et al*. (2020) was based on the last-step velocity. In contrast, our proposed model uses the average speed for every person that is calculated from a sequence of data. To process all these data, we will use LSTM networks to handle the sequence of data for pedestrians. LSTM networks have the proven ability for predicting sequential data. We employ the concept of the Unit angle discussed above, to get the directions of past trajectories, and then we feed the sequential results to the LSTM networks. The angle of direction



(a) LSTM network example, which f represents the forget gate, i represents the input gate, and o represents the output gate.

$$f_t = \sigma(W_f * X_t + U_f * h_{t-1})$$

$$i_t = \sigma(W_i * X_t + U_i * h_{t-1})$$

$$o_t = \sigma(W_o * X_t + U_o * h_{t-1})$$

$$\hat{C}_t = tanh(W_g * X_t + U_g * h_{t-1})$$

$$C_t = \sigma(f_t * C_{t-1} + i_t * \hat{C}_t)$$

$$\hat{C}_t = tanh(C_t) * o_t$$

(b) LSTM network equations

FIGURE 4.2: LSTM network example and equations that describe how LSTM network functions in the LSTM modules.

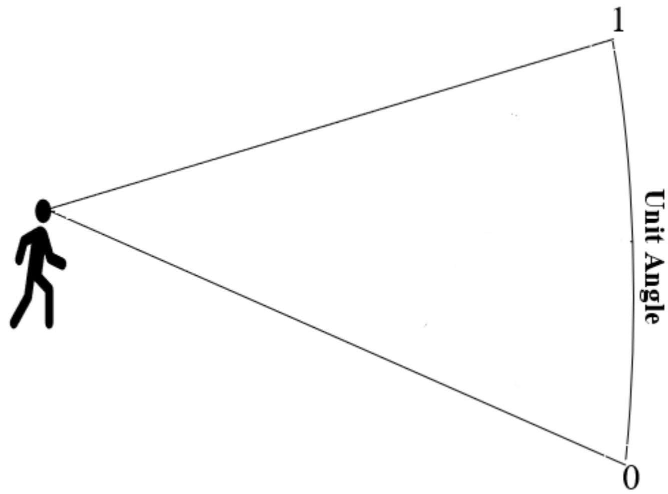


FIGURE 4.3: our model focuses on the individual's previous direction in his/her cone of vision to predict his/her next positions. The patterns of his/her last directions are represented as a distance and angle. The angle will be scaled between 0 and 1.

in the cone of vision will be calculated as $\theta = Atan2(y2 - y1, x2 - x1)$, and the equation for distance is $Distance = sqrt((x2 - x1) * (x2 - x1) + (y2 - y1) * (y2 - y1))$.

4.4.3 *Datasets*

We extended our work in Alajlan *et al.* (2020), which has two categories of datasets. The datasets have been acquired from the Netlogo simulation machine. One of the dataset represents structured crowded areas, in which people walk in the same way and go toward the same destination. The other dataset represents unstructured crowded areas, in which people go to several destinations in more than one way. Both of the datasets applied something similar to the social force model to simulate the behavior of people in a dense crowd. These social force rules include people keeping a fair distance between themselves and other people, between themselves and obstacles, acceleration of pedestrians to the desired speed, and the effect of the surrounding attractions. The datasets in [2] deal with the closest people in the cone of vision for each step that is fed into the neural networks. In short, the closest people's positions will be the inputs for the neural networks, which, in turn, allows for the prediction of the next direction. By contrast, using the same datasets, we will

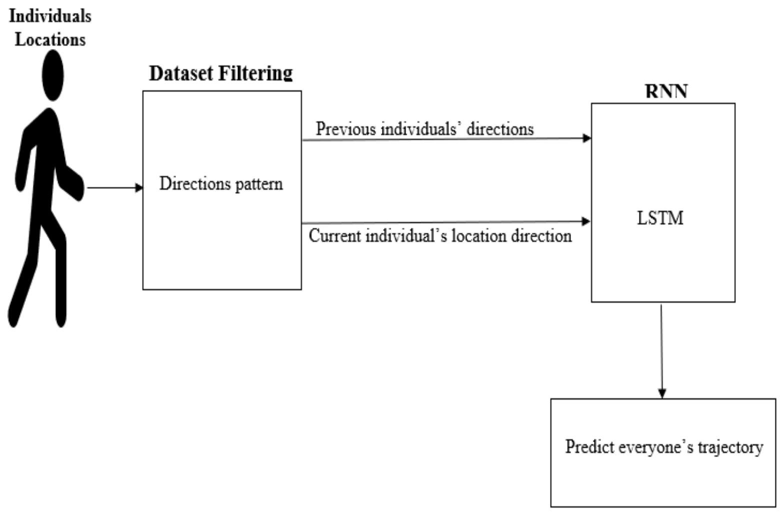


FIGURE 4.4: The workflow presents how to generate the Individuals locations data to be fed for the LSTM networks to predict the next individuals locations.

use past directions for every individual to obtain data about the patterns of his/her movements that , will predict next trajectories.

4.5 CONCLUSION

In this paper, we present a novel model based on an LSTM network for the prediction of pedestrian trajectories. Our proposed model depends on two factors, the last directions' values for each individuals, and the average speed for each individual. In other words, the model learns from last values of directions and predicts the future trajectories by LSTM networks. Additionally, the model calculates the average speed for each individual from the last steps' velocity, then specifies the future locations of individuals. The experiment in the proposed model was applied to two kinds of datasets based on crowd type: structured crowded areas and unstructured crowded areas.

CHAPTER 5

Predicting Human Movements Using Machine Learning [4]

5.1 Abstract

Assurance of safety in a crowd, such as the density of mass gatherings in religious rituals, or sports events, represents an important matter for authorities. Chaos and crowds create a big challenge for movement prediction. Even though, knowing the pedestrians' next positions is difficult regarding their behavior and their intentions, it is useful for ensuring their security. In this paper, we predict the future positions of pedestrians using the Recurrent Neural Network (RNN) and, specifically, its extension, Long Short-Term Memory (LSTM). We use the past directions of individuals predict their next directions, which leads to their future positions. We compared our proposed LSTM method, based on previous directions (LSTM-direction), with the neural networks and genetic algorithms (NN-GA) method. We used several datasets that are based on various scenarios and several simulation types, to evaluate both methods. Our results show that the LSTM based on the sequence of past directions and its features is superior to NN-GA.

5.2 Introduction

Avoiding dynamic objects such as other pedestrians, or static obstacles such as a wall is an individual's sensible method to continue moving to his/her destination. Keeping crowds stable and avoiding any dangerous situations are always the authorities' goal. Predicting a human's next positions is important, for instance, to avoid collisions when robots exist in a crowd. Next-step prediction is also useful to assure the crowd's safety and ensure smooth motion in dense crowds, such as at the Islamic Hajj in Saudi Arabia or the Kumbh Mela in India. Another significant reason for the prediction of

[4]This chapter was published in the 2022 International Conference on Computational Science and Computational Intelligence .

pedestrians' next steps arises when, for instance, designing temporary barriers in densely crowded areas, or if two or more paths merge into one route.

Numerous studies have suggested methods to resolve crowd density difficulties. Resolving crowdedness issues, such as protecting the crowd from chaos, or letting a crowd move faster, starts from analyzing different aspects of pedestrian behavior. For instance, dissolving any crowd density is an appropriate approach, in order to avoid collisions, and predicting the crowd's future positions in order to help authorities in organizations provide superior resolutions to what exist now. Crowds can be divided into two categories: structured crowds where agents walk in one direction with a common goal; and, unstructured crowds where agents walk in various directions with differing goals. It should be stated that there are other crowd types that combine these two categories.

Simulations are of great importance in studying a large number of agents in an environment who have various characteristics and behaviors in different scenarios. We applied two simulation methods: the social force model (SFM) that was suggested by Helbing and Molnar (1995); and the flocking model (FM) that was developed by Reynolds (1987), but with some additions that make the agent act more like pedestrians.

Attempts to understand pedestrians' behaviors are the keys to predicting their future positions. Plenty of research using various methods search for accuracy in the prediction of humans' movements. Our proposed method is to use machine learning, and specifically Recurrent Neural Networks (RNNs) to study the behaviors of individuals in the environment of crowd density.

RNNs and particularly LSTMs have shown efficacy in predicting sequential data. Alahi *et al*. (2016) was the first study that used LSTM networks in the prediction of humans' movements; based on past positions, they connected the LSTM networks for trajectories, and the networks then share information with nearby networks.

Our work is presented in five sections. The background of our work is illustrated in the Section 2; it is followed by the details about our model in Section 3. Results and discussions are detailed in Section 4. The last section of our work, Section 5, includes our conclusions and future works.

5.3 BACKGROUND

5.3.1 *Humans' Movements Predictions*

Various approaches utilize LSTMs to handle trajectories that are based on X-Y coordinate systems; these are able to discover sequential behavioral patterns in the observed motion trajectories. Alahi *et al*. (2016) was the first study that included LSTMs in the prediction of pedestrian trajectories. They based their work on previous positions to predict future trajectories by introducing a social pooling layer. They connected a pedestrian LSTM network with nearby LSTMs networks. Unlike taking into account only the nearby humans, Bartoli *et al*. (2018) extended the social-LSTM to consider all the objects in the scene that could have an impact on the pedestrian. By taking into account other agents' behaviors, Xu *et al*. (2018) suggested using LSTMs to shape the movement of all persons and then, based on the spatial locations of people, they scaled the movement features to predict the displacement. By combining the RRT-Reach algorithm and mixtures of Gaussian processes, Aoude *et al*. (2011) showed how independent position patterns can be utilized to forecast an agent's motion to avert collisions and improve the detection system. Shi *et al*. (2019), proposed LSTM to be used for encoding and decoding motions and behaviors for a long sequence of forecasts. Manh and Alaghband (2018) illustrates two Scene-LSTM models that can forecast people's movement; they explain how the data from the environment is significant by using it as inputs to the cells, which then utilize just the relevant information to anticipate future motions. Xu *et al*. (2018) used LSTM to gather movements' information, then modeled a crowd interaction deep neural network (CIDNN) to forecast the displacement of pedestrians. Gambs *et al*. (2012) developed a model called n-MMC that is an extension of Mobility Markov Chain (MMC), which in turn predicts the next location based on visited positions. To upgrade the LSTM capability, Quan *et al*. (2021) proposed a holistic LSTM; they added additional memory cells, which include a speed cell, an intention cell, and a correlation cell. Furthermore, they suggested an intention gate to assist with dealing with complicated movements' information for trajectory prediction. Based on the inverse reinforcement learning method, Henry *et al*. (2010) proposed an approach to mimic humans in crowd areas;

they taught a mobile robot to move through the crowded area safely, and to take the shortest path when the crowd was not dense.

5.3.2 *Crowd Management*

One of the big examples of structured and crowded paths is in Mecca, Saudi Arabia, during the Hajj. Many researchers have suggested a number of technologies for crowd management for this big event. We believe obtaining instant data from pilgrims' locations is the first step for predicting collective future trajectories. For pilgrims' identification, tracking, monitoring and planning, Mohamed *et al*. (2019); Mohandes *et al*. (2011); Al-Hashedi *et al*. (2014); Naser *et al*. (2010); Binsalleeh *et al*. (2009); Alsubhy *et al*. (2020); Mitchell *et al*. (2013) proposed using RFID technology. We started our experiments based on two simulations and many scenarios, and, as noted in Rodriguez *et al*. (2009), we categorized crowds in two types: the structured crowded environment, which has one direction and one goal as in Figure 7.1(b); and, the unstructured crowded environment, which has various directions and various goals as in Figure 5.1(a). To improve Hajj crowd management, Ahmad *et al*. (2014); Rahman *et al*. (2017); Mohamed *et al*. (2013); Lakhdari *et al*. (2020) suggested utilizing mobile phone applications and GPS for pilgrims' tracking and monitoring. Additionally, Alshalani *et al*. (2020) proposed an application that is based on GPS to organize pilgrims during Hajj; this includes locating the pilgrim and his/her responsible manager during Hajj, which provides a notification to both the pilgrim and his/her manager if the pilgrim is away from the group, etc. Yamin (2019) took the Hajj in Saudi Arabia and the Kumbha Mela in India as case studies for using wireless and mobile devices to facilitate crowd management in order to reduce the health risks between pilgrims. To help crowd management at the Hajj, Mitchell *et al*. (2013) proposed using RFID and mobile integration to determine pilgrims' locations and track them. To manage the density of the crowd effectively, Al-Kodmany (2013) proposed three pillars: an examining designs, preparing realistic plans, and identifying any difficulty.

Crowd Simulation Interactivities — In our simulations, two simulation types were implemented: the social force model suggested by Helbing and Molnar

(1995) and the flocking model. The social force model is a popular simulation that can mimic collective behavior. Pelechano and Malkawi (2008) described how the social force model represents a situation of people's panic in simulations and how people in a panic mimic what others do. The flocking model uses rules for cohesion in a flock and for avoiding collisions. Mehran *et al*. (2009) suggested a method to detect unusual actions in the environment of crowdedness. Dewi *et al*. (2011) conducted an experiment that utilized the flocking algorithm to simulate collective behavior in order to reach a specific aim. The Netlogo programming environment is an important program for simulating people in various conditions, and following different rules. Pluchino *et al*. (2013) used the Netlogo model to mimic peoples' movement at the Castello Ursino Museum in Catania, Italy. Camillen *et al*. (2009) utilized Netlogo for simulating people who visit a museum and it applied an evacuation plan for various situations, such as for different group sizes.

5.4 OUR MODEL

In a crowd, individuals are usually aware of the people in front of them, by which, they adapt their speed and the direction of their next position in order to be in the appropriate place and avoid collisions. This adaptation creates a sequence of movements that draws a pattern for successive steps. We studied the behavior of the past successive directions for each individual in order to predict the future location and direction for all of the agents.

Because this is based on people's sequences of steps, and requires the knowledge of their past sequential directions, we used LSTMs. Unlike previous LSTMs methods that are based on past positions, we used the cone-of-vision concept Alajlan *et al*. (2020) to consider the past directions inside what we called Unit Angle for each person in the scene. The observed direction paths inside the field of view were our inputs for the machine learning system.

(a) A picture of an unstructured crowded area.Image from Xu et al. Xu *et al*. (2019).

(b) A structured crowd. The image is a screen capture from Ministry of Hajj videos for Hajj 2019, https://www.haj.gov.sa/en

Figure 5.1: Examples of two major systems of crowd flow.

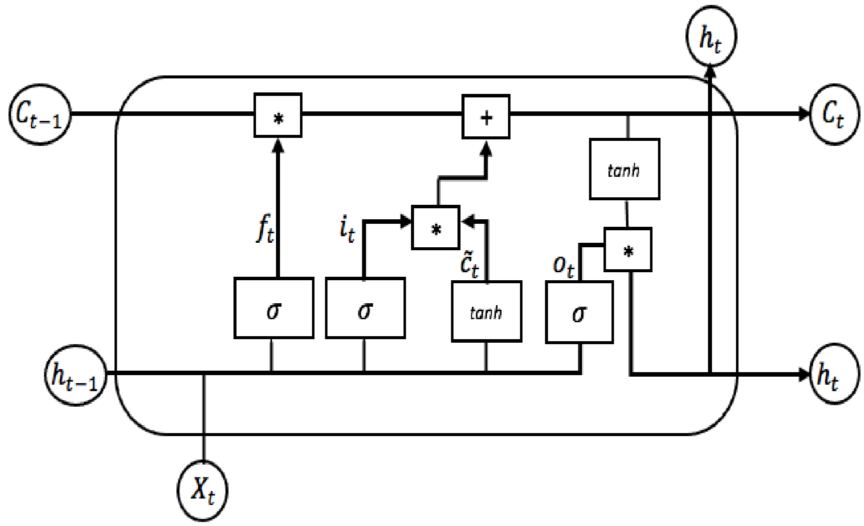


Figure 5.2: An example of an LSTM network that is represented by three gates (input gate, forget gate, and output gate). More details can be found in figure 4.2.

5.4.1 *Overview of LSTM*

LSTMs are an extension of Recurrent Neural Networks (RNNs). RNNs work with sequential data to generate or predict future data. Vanishing and exploding gradients are the difficulties faced by RNNs, which remain a hurdle for training RNNs. Vanishing gradients occur when gradients head toward very small values whenever they shift to the earlier layers. On the other hand, exploding gradients occur when the gradients head to very large values whenever they shift to the earlier layers. To resolve this problem, Hochreiter and Schmidhuber (1997) suggested the long short-term memory, which has the advantage of remembering or recalling data from long sequences. Three gates are employed by an LSTM to form its model: forget gate, input gate, and output gate. Dealing with a small number of hidden layers may not affect the values, but a large number of hidden layers may lead to an unstable condition. In Figure 5.2, we illustrate one LSTM network.

LSTMs have plenty of successes in predicting pedestrians' next positions. These studies motivated us to improve the idea of using the past sequence of directions inside the cone of vision to study the movement patterns of individuals, especially in dense structured crowds.

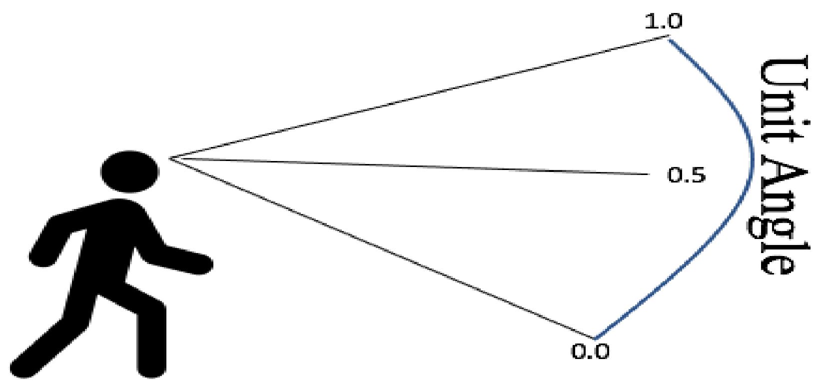


FIGURE 5.3: our model concentrates on the individual's past direction and obstacles in his/her cone of vision to forecast next locations. The patterns of people's directions are represented as a distance and angle. The angle is scaled between 0 and 1.

5.4.2 *Problem Statement*

Our aim is to accurately foresee individuals' locations from a series of data dependent upon their walking patterns. The difficulties are often dependent on the precision of the human trajectory forecast when comparing the true position to the expected location. Our proposed model is based on two major assumptions: 1) a person's trajectory is determined by their prior steps; and 2) the speed used to predict the final predicted position over time is dependent upon the prior velocities. The action is depicted as a drawn pattern in our model. We drew a series of steps that are dependent on the field of view, as shown in Figure 5.3, and for each person, we used the concept of the cone of vision that is called "Unit Angle".

5.4.3 *Simulations and Datasets*

Regarding Ridel *et al*. (2018), studies in this field have created various datasets specifically in urban scenarios, and some have compared their results with one or two other algorithms or approaches. A major problem is the datasets themselves; either the pedestrians are pre-instructed, and he/she follows what the data collector wants, or in realistic scenarios, when the person is not an actor, most of the pedestrians, when crossing a street, for instance, negotiate by eye contacts and eye contact is not easily modeled. In our proposed approach, we focused on the predictions for crowded areas, and used the cone of vision as an analogy to eye contact.

In this paper, we expanded our work from Alajlan *et al*. (2020) to include more than one type of simulation and multiple scenarios. One of the popular pedestrians' simulations is the Social Force Model that was suggested by Helbing and Molnar (1995); the second simulation in our experiment is the flocking model that was developed by Reynolds (1987). Similar scenarios are applied in these two types of simulations to track the difference in predictions. Our scenarios were inspired from real life and some of them are specifically from the pedestrians' paths in the Islamic ritual Hajj, which pilgrims follow in Mecca. The first scenario is of a structured crowded area with high density and low density. The second scenario is of an unstructured crowded area with high density and low density. The third scenario is of two structured crowded paths that merge and head toward one destination. As with the other scenarios, the merged paths were tested with the high density and low density.

The Social Force Model (SFM) is based on three factors: 1) agents accelerate to the desired speed; 2) agents avoid other pedestrians and obstacles; and, 3) agents are attracted to a goal. SFM is one of the most widespread methods for characterizing the movement of individuals in crowds. We represent the SFM in equation 5.1 as:

$$\mathrm{SF}\,i = \mathrm{desired}_i + \sum_{j \neq i} \mathrm{social}_{ij} + \sum \mathrm{obstacle}_{iw} \tag{5.1}$$

where $desired_i$ is the desired force of direction and velocity, $social_{ij}$ is the social force that wards the pedestrian off from other pedestrian, and $obstacle_{iw}$ is force of an obstacle that wards off the pedestrian from obstacles. We based our second simulation model on the **Flocking Model** that comes with Netlogo, but with several changes. Netlogo is a multi agent modeling platform that may be programmed to imitate natural and social processes. The applied properties in our simulation include: the maximum speed, acceleration to the desired speed, avoidance of other pedestrians and obstacles, cone of vision, and cohesion. The model uses several

FIGURE 5.4: Pedestrian paths merging in Mecca, Saudi Arabia. The image is from Google Earth.

parameters: separation, alignment, cohesion, and cone of vision. Byrisetty (2013) summarizes the job of the first three parameters, separation, alignment, and cohesion. In separation, the agent aims to move towards the destination while avoiding other agents; in alignment, the agent aims towards the average heading of nearby agents; and in cohesion, the agent moves toward the average position of other agents. The cone of vision specifies the angle of vision that the agent can see while moving.

The datasets were collected from Netlogo for four properties: ID, Time, X-coordinate, and Y-coordinate; these are recorded for each agent at each time step. The scenarios are, a structured crowd, an unstructured crowd, and merging paths as in Figure 5.4, and are applied in both SFM and FM simulations.

5.4.4 *Methodology*

In Alajlan *et al*. (2020), we identified the locations of the nearest three persons in a cone of vision and predicted the location the individual chose based on these people in front of him/her. Equation 7.3 clarifies how the model works in Alajlan *et al*. (2020). In this equation, c represents the current pedestrian, and the model looks for the nearest three persons i in the field of view of c; they are then sorted for the closest.

$$nearest3(c) = \underset{i \in Cone_c(Pop)}{\operatorname{argmin} 3} \ \operatorname{distance}(c,\ i) \tag{5.2}$$

The locations of the nearest persons are the data that were fed to the neural networks. By using genetic algorithms and neural networks, we obtained the best neural network to predict next steps. We then compared the low densities of the structured and unstructured crowded areas.

Additionally, we extended our work to try another technique with another strategy that uses the same concept. This model is based on the LSTMs algorithm, and it has two facets: 1) using the field of view to figure out the directions for many steps in the past for each individual; and, 2) calculating the average speed while each agent was taking actual steps in his/her desired directions. Our idea was inspired by employing the cone of vision that was used in Alajlan *et al*. (2020), not to determine the nearest people for each pedestrian, but to draw a sequence of pedestrians' aimed directions in their Unit Angle that they decided on in many previous steps. Figure 5.5 shows how the model draws one step for one individual to illustrate the idea. The sequence of directions are numbers between 0 and 1. We specified the angle for each individual based on his/her direction. For example, if the pedestrian moved one step in angle 0, and if we specify the field of view to be 100, then the cone of vision for his/her next step is between -50 and 50. The angle -50 will be represented as 0 and angle 50 will represent the number 1, and every angle of direction in between these two angles will be a number between 0 and 1. We trained our data on LSTM networks as follows:

(1)reshaping the inputs data, which are the numbers of directions in the Unit Angle for four steps; (2)feeding the LSTMs with these steps of directions for each individual; (3)training the model for the whole dataset in order to predict the fifth step and optimizing the difference between the actual directions and the predicted directions using the mean square error loss function; (4)calculating the average speed for each individual while obtaining the previous directions; and, (5)testing the model by predicting the next four steps for each individual.

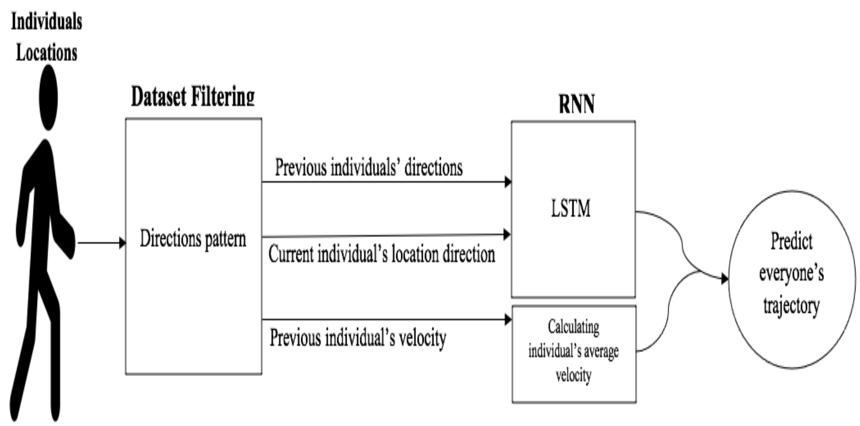


Figure 5.5: Our model concentrates on drawing a pattern of each individual's past directions in his/her cone of vision. Based on the cone of vision angle, several steps will be scaled between 0 and 1, and average velocity will be calculated.

5.4.5 *How We Tested Our Model*

The result of the predicted fifth step was fed to the LSTMs to predict the sixth step, which means that the second, third, fourth, and predicted fifth step will be the new inputs into the LSTMs to predict the sixth step. At the same time, we calculated the average speed in order to figure out the distance for the next step, which in this instance is the sixth step. The testing process continued to establish the rest of the predicted steps.

5.5 RESULTS

The LSTMs directions experiment was achieved using the open-source library Keras Chollet *et al*. (2015). We ran our experiments using two simulations, SFM and FM. The scenarios were inspired from real world situations, including the structured crowded area, the unstructured crowded area, and the merging paths area and these were used to examine the effectiveness of our approach. In the LSTMs experiment, we observed four time-steps in order to predict the next five-time-steps; each time-step in the sequence was observed and input to the machine learning algorithm. During our observations, the average velocity for each individual was computed. In the neural networks and genetic algorithms method, we used the same approach as in Alajlan *et al*. (2020), in which the inputs to the machine learning were the values of the closest three persons in the agent's field of view. Some improvements were applied to the second method to assist in the evolution of its results, such as considering the average speed of the steps' distance instead of constant speed. We compared our new method that uses the LSTMs based on the direction of the individual's cone of vision with neural networks and genetic algorithms that were used in Alajlan *et al*. (2020). In table 5.1, we present our prediction methods error based on three metrics. For the first measurement, we illustrated the error of prediction using the average displacement error, as in Pellegrini *et al*. (2009), which calculates the mean Euclidean distance between predicted points and the ground truth, through all time-steps.

$$ADE(\hat{P}, P) = \frac{1}{T}\sum_{i=1}^{T}\sqrt{(X_i - \hat{X}_i)^2 + (Y_i - \hat{Y}_i)^2} \tag{5.3}$$

For the second measurement, we used the final displacement error, which calculates the mean Euclidean distance between the predicted final step and the true final position, as in Alahi *et al*. (2016).

$$FDE(\hat{P}, P) = \sqrt{(X_F - \hat{X}_F)^2 + (Y_F - \hat{Y}_F)^2} \tag{5.4}$$

For the third measurement, we reported the cumulative distance error (CDE), as in Alajlan *et al*. (2020). For the CDE, we calculated the average distance between the predicted final point and the true final position, divided by the average distance between the step from which we had started our prediction and the true final step; finally, we multiplied the result by 100 to give the error as a percent of the total movement.

$$CDE(\hat{P}, P) = \frac{1}{N}\sum_{i=1}^{N}\frac{\sqrt{(X_F - \hat{X}_F)^2 + (Y_F - \hat{Y}_F)^2}}{\sqrt{(X_F - \hat{X}_S)^2 + (Y_F - \hat{Y}_S)^2}} \times 100 \tag{5.5}$$

In the results, we included the average for each method in its simulation type over all scenarios.

5.5.1 *LSTM-Direction vs. NN-GA*

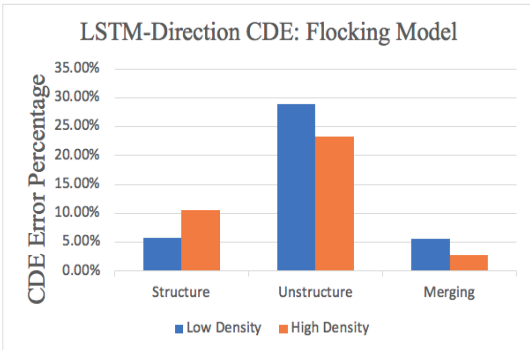
To determine whether our new method, LSTMs-based direction (LSTM-direction) that predicts trajectories, is superior to the neural networks and genetic algorithms (NN-GA) approach in Alajlan *et al*. (2020), we compared these two algorithms. The LSTMs-direction algorithm, uses past directions inside the cone of vision, while the (NN-GA) algorithm uses the closest three persons in the individual's field of view. Second, we used two movement simulations, and multiple scenarios with different densities of crowds (low density-high density) assure that both approaches were stringently tested. The results in Table 5.1 show that the new technique that uses (LSTMs-direction) outperforms the (NN-GA) in both simulations, and for every dataset.

5.5.2 *Low Density vs. High Density Scenarios*

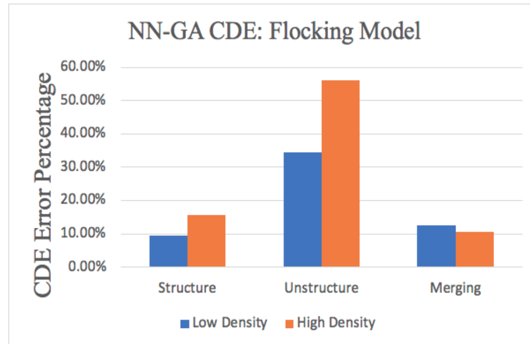
We present our observations about the variation between low density and high density in our results in Figure 5.6. In the SFM, the structured crowded areas/unstructured crowded areas showed superior results in low density for both algorithms, while the merging paths scenario had a lower error in the high density scenario. On the other hand, the FM structured crowded area with low density created less fewer errors than high density, while the merging paths areas high density outperformed low density. In the unstructured crowded areas of the FM, each algorithm acted differently; the high-density was superior to the low density in the LSTM-direction situation; and, the low density produced minimal error in the NN-GA situation.

5.5.3 *Average and Final Errors*

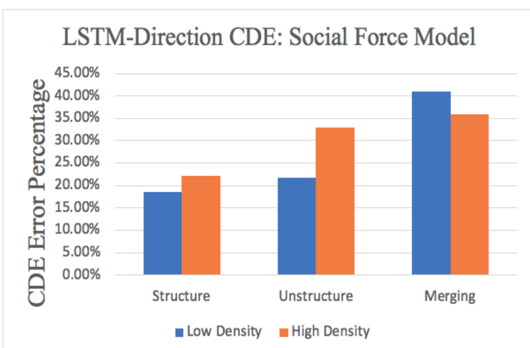
The innovation of our new method is to employ the strategy that we previously used with a (NN-GA) based on the closest people in the cone of vision, but to draw a pattern of the past individuals' directions in their field of view. Additionally, we tested the approach with multiple behaviors and scenarios. We observed the pilgrims' behavior on some watchtowers' cameras from the Hajj 2019 to design our flocking model. In the social force model, we simulated the model that was suggested by Helbing and Molnar (1995) to imitate another behavior for pedestrians. We implemented a diversity of simulations, crowd densities, and scenarios. Table 5.1 presents the resulting difference between our approaches for all three error metrics. Every scenario has two densities of pedestrians. Each density has been tested on two datasets, one belonging to the flocking model, and one belonging to the social force model. Each dataset was used to test two prediction approaches, the (LSTMs-direction) and the (NN-GA).



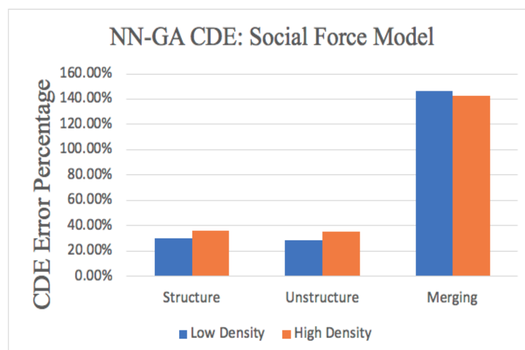
(a) The cumulative distance error in LSTM-direction for a flocking model simulation between a low-density crowd and a high-density crowd.



(b) The cumulative distance error in NN-GA for a flocking model simulation between a low-density crowd and a high-density crowd.



(c) The cumulative distance error in LSTM-direction for the social force model simulation between a low-density crowd and a high-density crowd.



(d) The cumulative distance error in NN-GA for a social force model simulation between a low-density crowd and a high-density crowd.

FIGURE 5.6: Low density-high density Cumulative Distance Error (CDE). Note the larger scale for the NN-GA model

TABLE 5.1: The results for the two prediction approaches in different simulation types and different scenarios. We report the performance for each algorithm using three metrics. The first four rows represent the average displacement error, the next four rows represent the final displacement error, and the last four rows show the cumulative distance error. Regarding simulations, algorithms, and scenarios, the second column represents the simulation type, the third column is the kind of algorithm, and the rest show various crowd scenarios. The LSTM-direction-based prediction outperforms NN-GA-based prediction in all cases

			Structured		Unstructured		Merging		Avg.
			Low D.	High D.	Low D.	High D.	Low D.	High D.	
Avg. disp. error	SFM	LSTM-direction	0.398	0.439	0.486	0.687	0.371	0.313	0.449
		NNs& GAs	0.660	0.752	0.630	0.736	1.468	1.39	0.939
	FM	LSTM-direction	0.064	0.089	0.215	0.197	0.034	0.016	0.102
		NNs& GAs	0.123	0.136	0.344	0.692	0.072	0.056	0.237
Final. disp. error	SFM	LSTM-direction	0.696	0.786	0.830	1.172	0.678	0.606	0.794
		NNs& GAs	1.124	1.278	1.092	1.264	2.446	2.373	1.596
	FM	LSTM-direction	0.152	0.158	0.466	0.391	0.062	0.02	0.208
		NNs& GAs	0.244	0.273	0.580	0.879	0.139	0.109	0.370
CDE	SFM	LSTM-direction	18.5%	22.16%	21.76%	32.85%	41.02%	35.98%	28.71%
		NNs& GAs	29.82%	35.80%	28.39%	35.27%	146.8%	142.33%	69.73%
	FM	LSTM-direction	5.7%	10.6%	28.9%	23.3%	5.5%	2.8%	12.8%
		NNs& GAs	9.37%	15.55%	34.35%	56.2%	12.41%	10.62%	23.08%

5.6 CONCLUSION

Our proposed new method, called (LSTM-direction) uses the individuals' directions in their cones of visions to track the patterns for pedestrians several steps, to predict the future trajectory based on the past patterns and past average speed. We compared our new technique with the strategy that has been used in [3], that we call (NN-GA), which used the cone of vision to determine the closest three persons in the field of view as inputs to the prediction algorithm. The results show that (LSTM-direction) is superior to (NN-GA) in every dataset on which we experimented. Our datasets collected data from two simulation types, two different densities of pedestrians, and three scenarios from the real world. Our approach is focused on the pedestrians' field of view, because usually people in structured crowded zones, such as in the Hajj, are confined by their cone of vision.

The necessity for improving crowd management criteria motivated us to focus our research on a the fertile field of crowd management, which includes humans' movements prediction. In this we research we concentrated on the behavior of structured crowded areas, such as the holy places in Mecca. Saudi Arabian authorities recently created a vital means for how to improve the system of crowd management. However, there are two substantial matters that will be serious issues for the Hajj in the future: the increased number of pilgrims from three million to five million; and, digitizing the plan for the Hajj to include tracking pilgrims. The plan of obtaining immediate data from pilgrims' locations encouraged us to try to help and to be a part of the improved plan. Our future work will concentrates on predicting humans' movements for long trajectories, which we believe is doable. Moreover, this could assist the organizers' awareness ahead of time of any disaster that might occur in the pedestrians' path; for example, in the case of designing a barriers among the pilgrims.

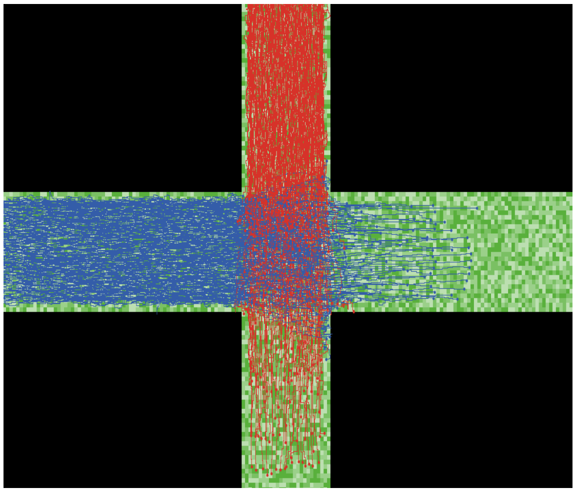
CHAPTER 6
Many Scenarios in Predicting Human Movement in Crowds

6.1 SUMMARY

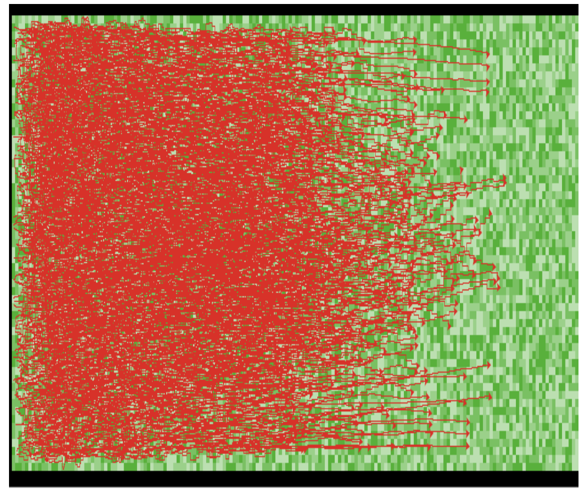
To continue testing our methods and comparing the results, we tested the LSTM-Direction, and the NN-GA techniques on multiple scenarios that correspond to commonly observed scenarios in crowd control situations. These are: an intersection, a structured crowded area, an unstructured crowded area (part 1), an unstructured crowded area (part 2), merging paths, and a waypoints path in the crowd behavior of the Social Force Model(SFM). Additionally, to test the limits of the algorithms we lengthened the period of time for both observations and predictions.

6.2 INTRODUCTION

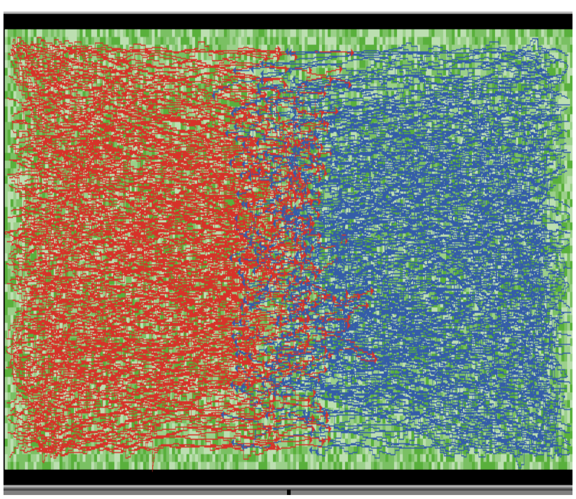
All of these scenarios are common in daily life. The Hajj, which is one of the largest gatherings of people in the world, has all of these scenarios during the Hajj journey. The structured crowded area, which is one of the popular paths for people to move from one ritual area to another, was simulated and is illustrated in Figure 6.1(b). The unstructured crowded area (part 1) scenario, which was the deadliest disaster in the history of the Hajj, was simulated and is illustrated in Figure 6.1(c). Intersections, which are common in Mina ritual places, were simulated and are illustrated in Figure 6.1(a). The unstructured crowded area (part 2), which is common during the night life at the Hajj, was simulated and is illustrated in Figure 6.1(d). Merging paths, also common at the Hajj, was simulated and is illustrated in Figure 6.1(e). The waypoints path, another path that is common during the Hajj, was simulated and is illustrated in Figure 6.1(f)



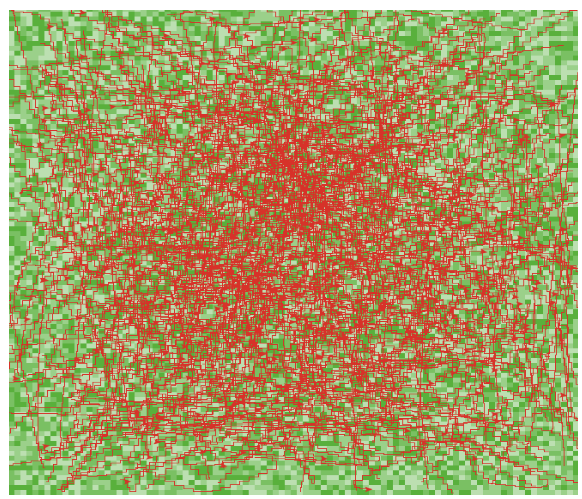
(a) Intersection simulation, in which people move from the top to their targets in the bottom, and from left to their targets on the right. They collide in the intersection before continuing to their targets.



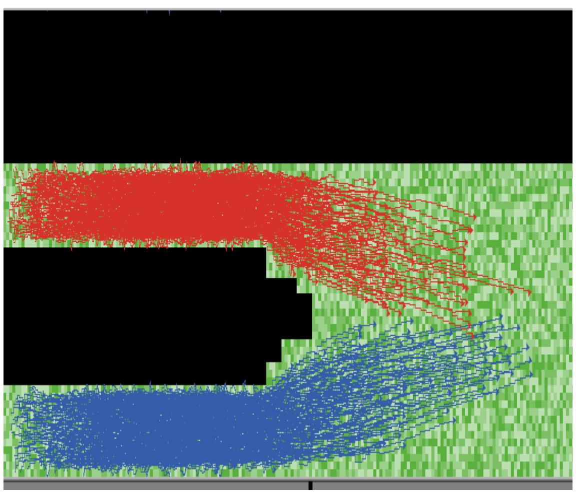
(b) Structured crowd simulation, in which people tend to move from left to right.



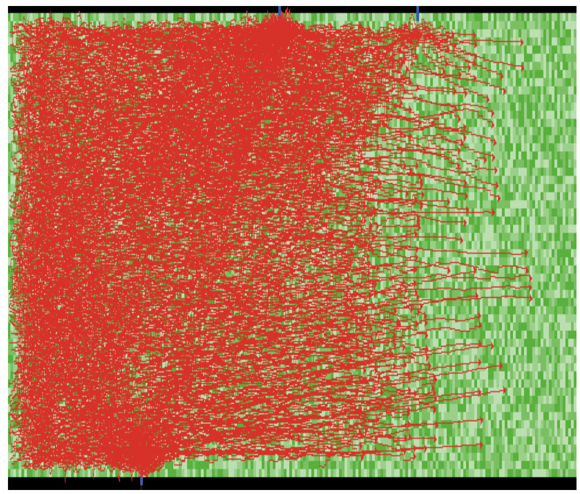
(c) Unstructured crowd simulation (part1), in which people on the sides move at random to a paired random location on the opposite side, initially at random speed.



(d) Unstructured crowd simulation (part 2), in which people move from one place to another, randomly selected, goal point; once the agent reached his target he immediately headed to another randomly selected target.



(e) Merging paths, in which people come from different paths and merge into one path.



(f) Waypoints path, in which people stop at different points for a while before continuing to their final goal

Figure 6.1: The figure illustrates NetLogo simulations of all scenarios in crowds

6.3 INTERSECTION

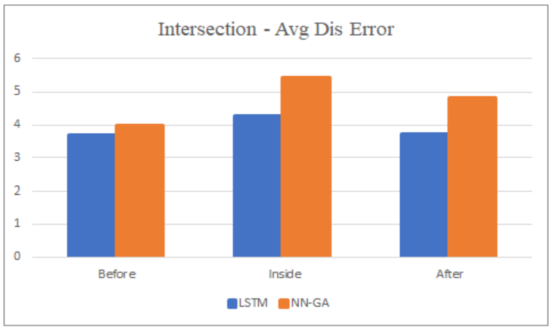
In the intersection, agents come from different paths, then pass through the intersection to their destination, which causes two structured crowds to overlap while in the intersection. There are three locations where we evaluated our prediction: before the agents enter the intersection, when the agents overlap in the intersection, and after the agents pass through the intersection. Our evaluations in this scenario used 5 time-steps, and the total number of predictions was 25 time-steps. In all three locations, we observed the agent every fifth time-step for five steps, then we predicted the next location at every fifth time-step, for a total prediction time of 25 time-steps.Although the results for the LSTM-Direction are close inside the intersection and after the intersection, we observed a small difference between the outcomes. The average displacement error (ADE) results, the final displacement error (FDE) results, and the cumulative distance error (CDE) results all produced larger errors inside the intersection. On the other hand, the NN-GA showed various results in these three locations; before agents entered the intersection had the lowest error in comparison with other locations. Finally, the LSTM-Direction did a better job in all three locations than the NN-GA.

6.4 STRUCTURED CROWDED AREA

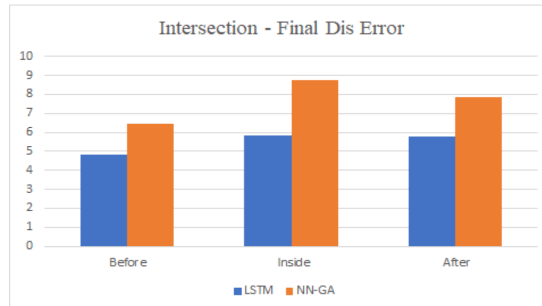
The structured crowded area is described as one in which the whole population moves in one direction, and they head toward one targeted destination. In this environment, the agents moved from the left to the right to their goals with different speeds. Our evaluations in this scenario were based on three factors: these included the 1 time-step where the total number of predictions was 5 time-steps; 5 time-steps where the total number of predictions was 25 time-steps, and 10 time-steps where the total number of predictions is was 50 time-steps. That is, 10 time-steps means that we observed the agent every tenth time-steps for five steps, then we predicted the next locations at every tenth time-step, which is 50 time-steps in total prediction time. The results for in table 6.2 both the LSTM-Direction and the NN-GA showed growth in ADE and FDE as the time passed, but the CDE decreased during the same period. Our new

Table 6.1: Illustrates the difference between the LSTM-Direction and the NN-GA in the three locations that included before the intersection, inside the intersection, and after the intersection. Three metrics were used: ADE, FDE, and CDE. Note the larger number in the error rate of the NN-GA model. Note the larger number in the error rate of the NN-GA model.

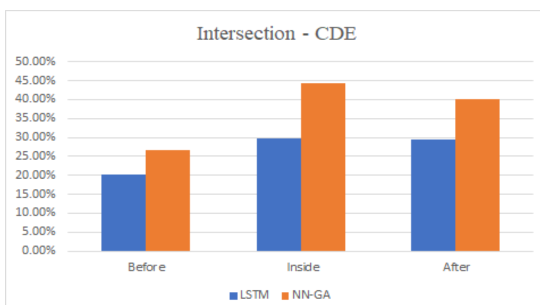
Before Intersection- 5 Time-steps		
Metrics	LSTM	NN-GA
Average Displacement Error	3.686	3.989
Final Displacement Error	4.831	6.434
Cumulative Distance Error	20.07%	26.73%
Inside Intersection- 5 Time-steps		
Metrics	LSTM	NN-GA
Average Displacement Error	4.245	5.447
Final Displacement Error	5.833	8.719
Cumulative Distance Error	29.63%	44.29%
After Intersection- 5 Time-steps		
Metrics	LSTM	NN-GA
Average Displacement Error	3.724	4.819
Final Displacement Error	5.752	7.828
Cumulative Distance Error	29.47%	40.11%



(a) The average distance error in the LSTM-Direction and the NN-GA for before intersection, inside intersection, and after intersection.

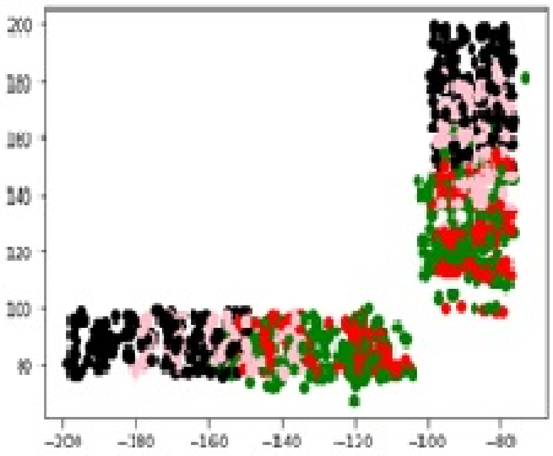


(b) The final distance error in the LSTM-Direction and the NN-GA for before intersection, inside intersection, and after intersection.

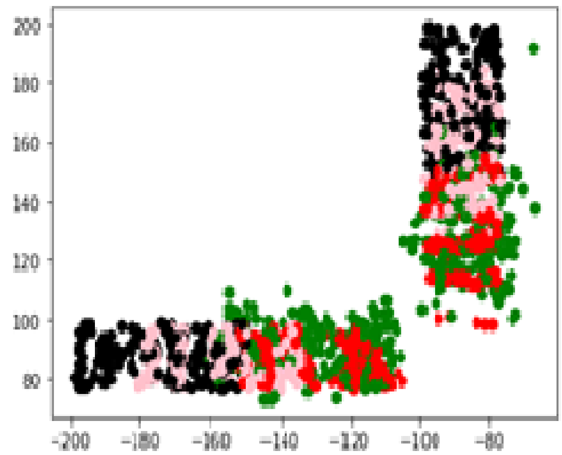


(c) The cumulative distance error in the LSTM-Direction and the NN-GA for before intersection, inside intersection, and after intersection.

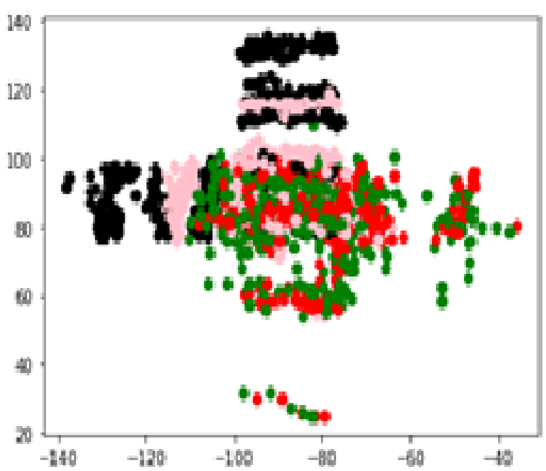
Figure 6.2: Intersection: before intersection, inside intersection, and after intersection ADE, FDE, and CDE for the LSTM-Direction model and the NN-GA model.



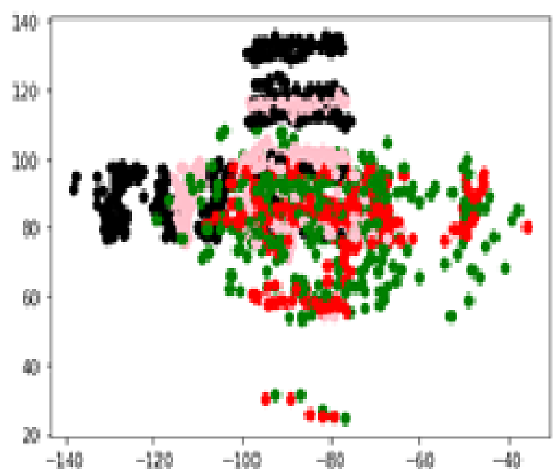
(a) Before intersection LSTM-Direction (all agents).



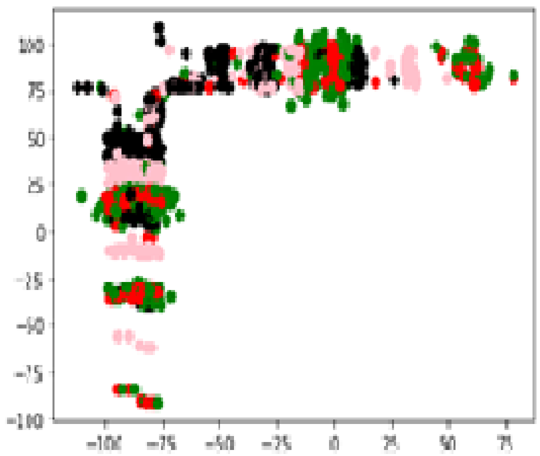
(b) Before intersection: NN-GA (all agents).



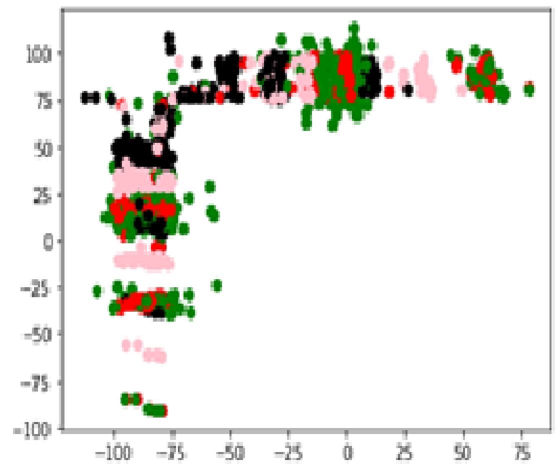
(c) Inside intersection:LSTM-Direction (all agents).



(d) Inside intersection:NN-GA (all agents).

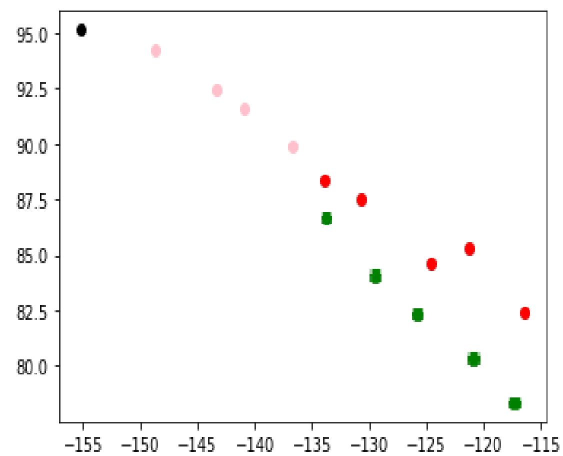


(e) After intersection:LSTM-Direction (all agents)

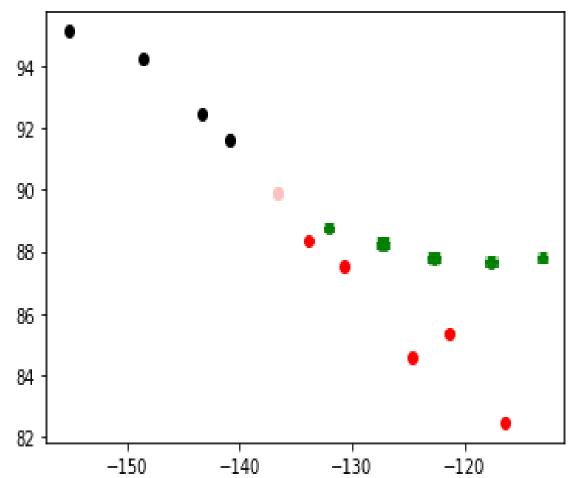


(f) After intersection:NN-GA (all agents)

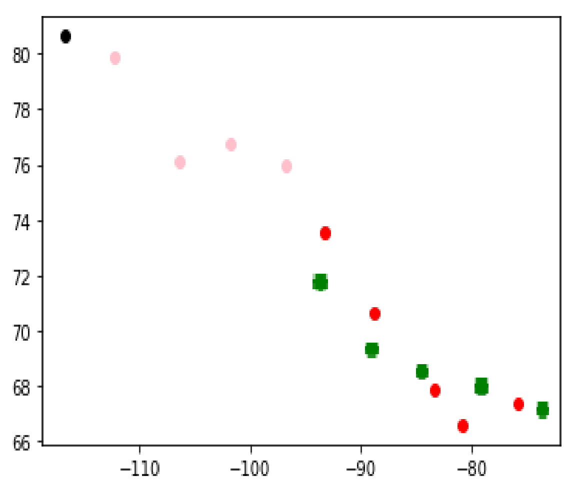
Figure 6.3: The graphs illustrate all agents in the three places before reaching the intersection, inside the intersection, and after passing through the intersection. The colors represent the following: black indicates the points where the agents start moving; pink indicates the points where the prediction starts; green indicates the last step of prediction; and red indicates the last step of ground truth. Note how the spread of prediction in green points inside the intersection, which has the worst outcomes between the three places for each method. Note the spread of prediction (green points) inside the intersection; this produced the worst outcomes among the three places for each method.



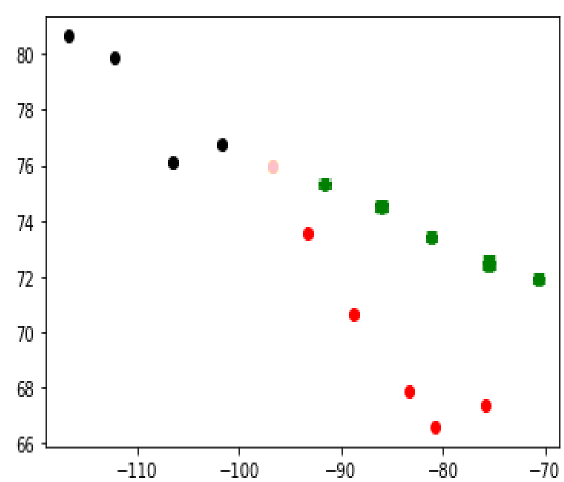
(a) Before intersection LSTM-Direction one agents) with final distance error= 4.259



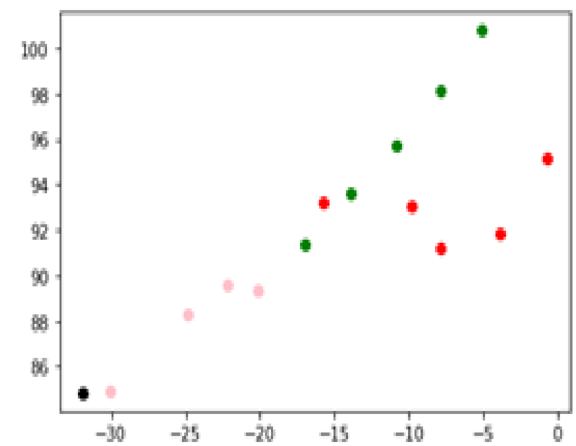
(b) Before intersection: NN-GA (one agents) with final distance error= 6.177



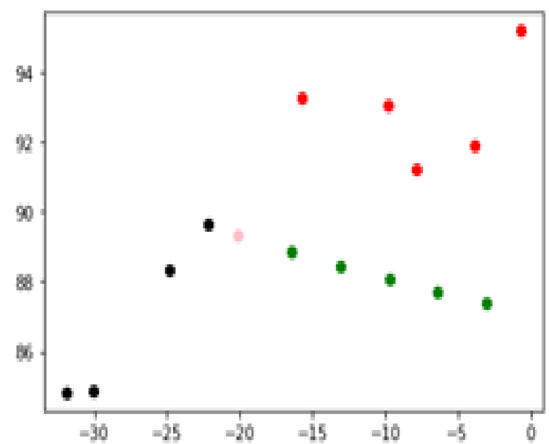
(c) Inside intersection:LSTM-Direction (one agents) with final distance error= 1.573



(d) Inside intersection:NN-GA (one agents)with final distance error= 6.142



(e) After intersection:LSTM-Direction (one agents) with final distance error= 7.145



(f) After intersection:NN-GA (one agents) with final distance error= 8.121

FIGURE 6.4: These figures illustrates an example of one agent from each place (before reaching the intersection, inside the intersection, and after passing through the intersection). The colors represent the following: black indicates the points where there is no observation or prediction; pink indicates an observation point; green indicates prediction points; and red indicates ground truth points.Note the predictions tend to be smoother, more of a straight line, than the actual movement.

measurement "CDE" gave us a new view of our results. The LSTM-Direction gave better results in all scenarios.

6.5 Unstructured crowded area (part 1)

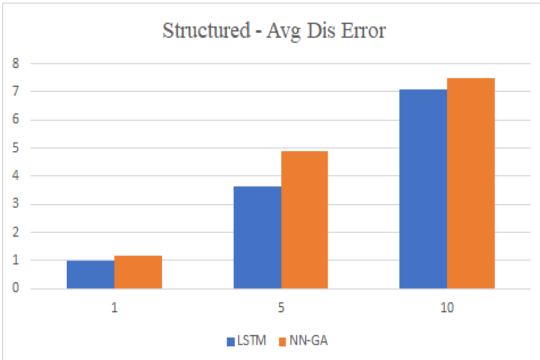
In the unstructured crowded scene (part 1), we created an environment that has two populations, one heading from right to left, and the other moving in the opposite way. Each agent had a specific target and tried to reach the goal in the shortest path. The time strategy in this unstructured crowded area used 5 time-steps, as in the intersection scenario, where the total number of predictions was 25 time-steps. Our evaluations were based on three different locations, before the two crowds met with each other, during the meeting, and after the meeting. The results for this experiment showed promising outcomes in the LSTM-Direction. In the NN-GA tests, we realized the experiment's outcomes improved after the interference, but the worst results for both methods were during the interference.

6.6 Unstructured crowded area (part 2)

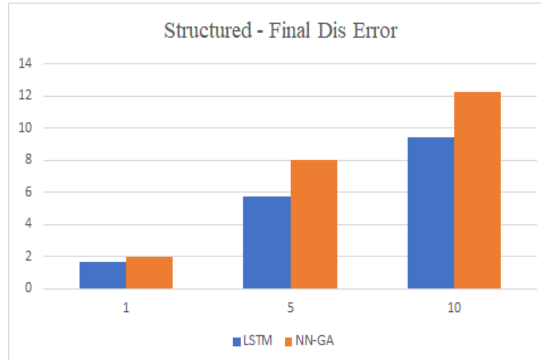
In the scenario of an unstructured crowded area (part 2), there were some changes in factors that differed from the unstructured crowded area (part 1). In this specific scenario, every agent moved from one point to another, randomly selected, goal point. Once the agent reached a goal point it immediately headed to another randomly selected goal point (i.e. the agents bounced from goal point to goal point), and so forth. Our evaluations in this unstructured crowded area were based on three different approaches, as they were in the structured crowded area: the time strategy of the 1 time-step, 5 time-steps, and 10 time-steps. The errors for ADE, FDE, and CDE increased as the time-steps increased. The CDE results for the last approach, 10 time-steps, produced a large number of errors. Although the LSTM-Direction produced a smaller number than the NN-GA, neither could be trusted at the specific scenario, which was the 10 time-steps of unstructured crowded area (part 2).

TABLE 6.2: This table illustrates the difference between the LSTM-Direction and the NN-GA in the structured crowded area and included 1 time-step, 5 time-steps, and 10 time-steps. Three metrics were used: ADE, FDE, and CDE.

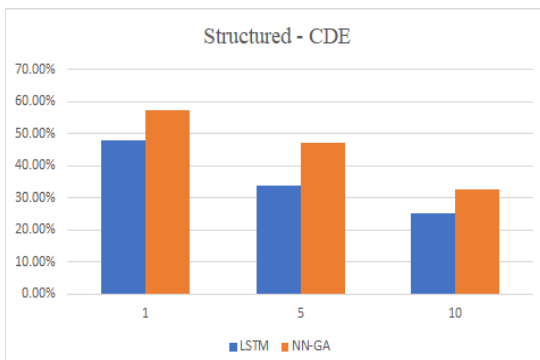
Structured - 1 Time-step		
Metrics	LSTM	NN-GA
Average Displacement Error	0.984	1.159
Final Displacement Error	1.639	1.967
Cumulative Distance Error	47.73%	57.26%
Structured- 5 Time-steps		
Metrics	LSTM	NN-GA
Average Displacement Error	3.618	4.88
Final Displacement Error	5.756	7.993
Cumulative Distance Error	33.96%	47.16%
Structured- 10 Time-steps		
Metrics	LSTM	NN-GA
Average Displacement Error	7.081	7.472
Final Displacement Error	9.399	12.211
Cumulative Distance Error	25.11%	32.63%



(a) The average distance error in the LSTM-Direction and the NN-GA for 1 time-step, 5 time-steps, and 10 time-steps.

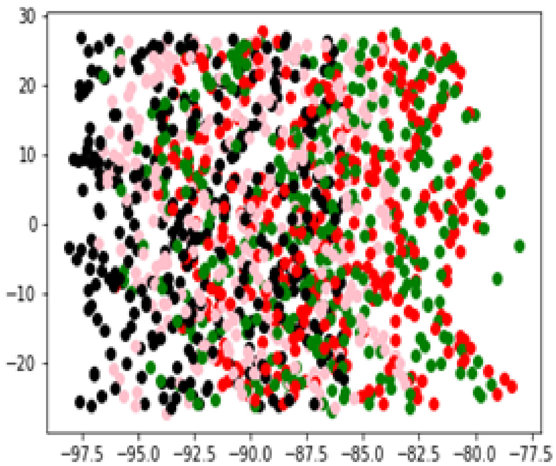


(b) The final distance error in the LSTM-Direction and the NN-GA for 1 time-step, 5 time-steps, and 10 time-steps.

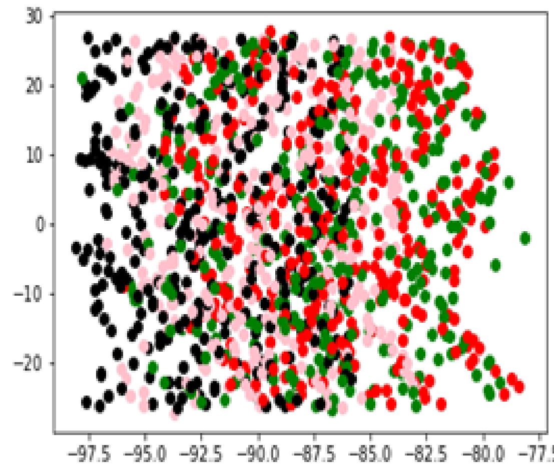


(c) The cumulative distance error in the LSTM-Direction and the NN-GA for 1 time-step, 5 time-steps, and 10 time-steps.

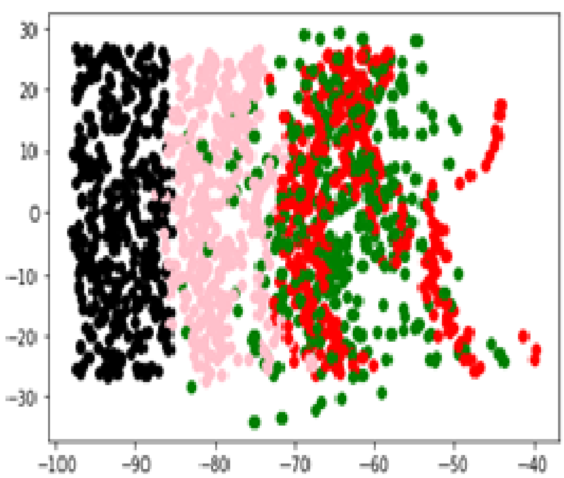
FIGURE 6.5: Structured: 1 time-step, 5 time-steps, and 10 time-steps ADE, FDE, and CDE for the LSTM-Direction model and the NN-GA model.The error increase in the ADE, and FDE as the time passed, but the CDE error decreased as the time passed. Note that in this structured crowded area scenario, we are obtaining a more accurate results in the CDE when time passed. The error increased in the ADE, and the FDE as time passed, but the CDE error decreased as time passed. Note that in this structured crowded area scenario, we obtained more accurate results in the CDE when time passed.



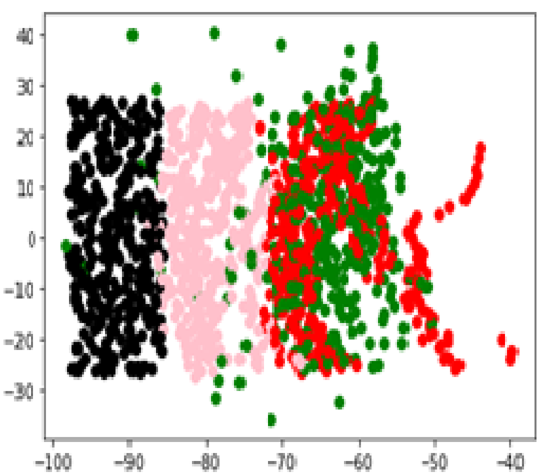
(a) 1 time-step: LSTM-Direction (all agents).



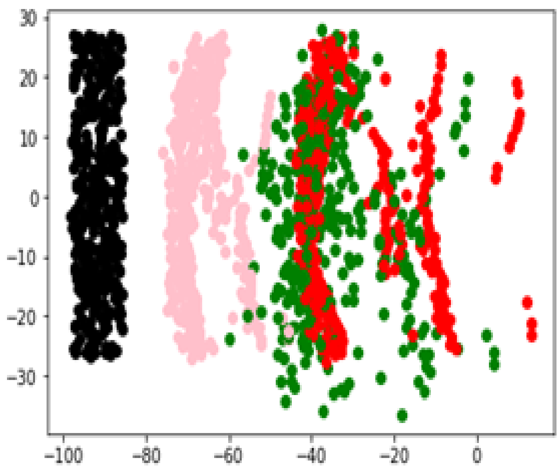
(b) 1 time-step: NN-GA (all agents).



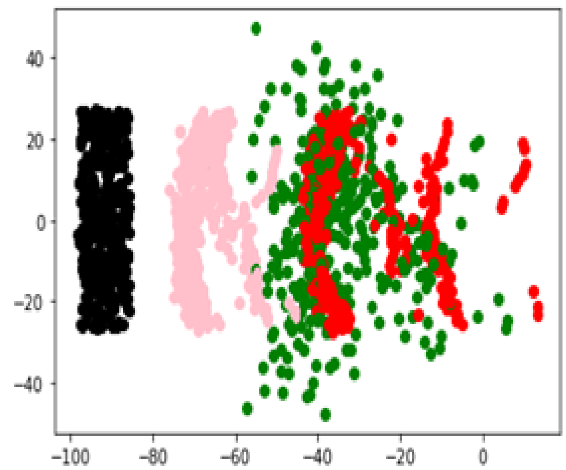
(c) 5 time-steps: LSTM-Direction (all agents).



(d) 5 time-steps:NN-GA (all agents).

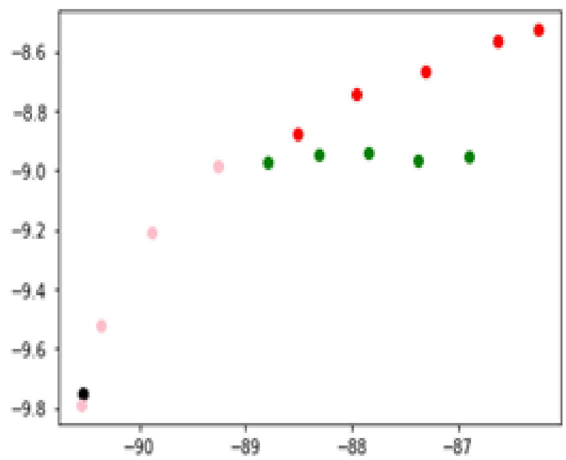


(e) 10 time-steps:LSTM-Direction (all agents)

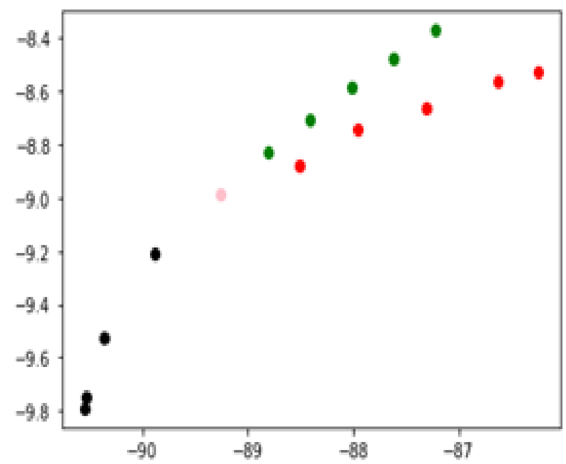


(f) 10 time-steps:NN-GA (all agents)

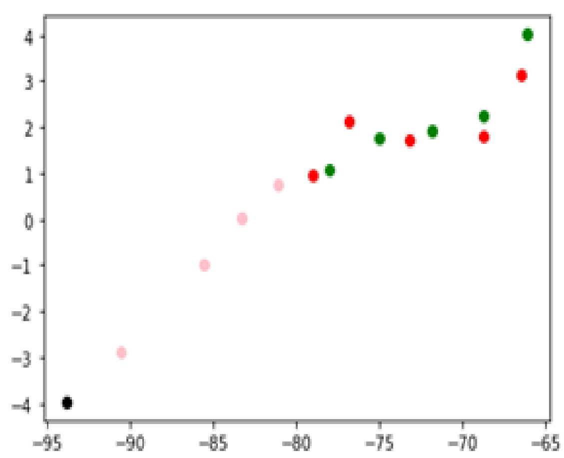
FIGURE 6.6: Structured crowded area: The following illustrates all agents in the three scenarios that included 1 time-step, 5 time-steps, and 10 time-steps. The colors represent the following: black indicates the points where the agents start moving; pink indicates the points where the prediction starts; green indicates the last step of prediction; and red indicates the last step of ground truth.



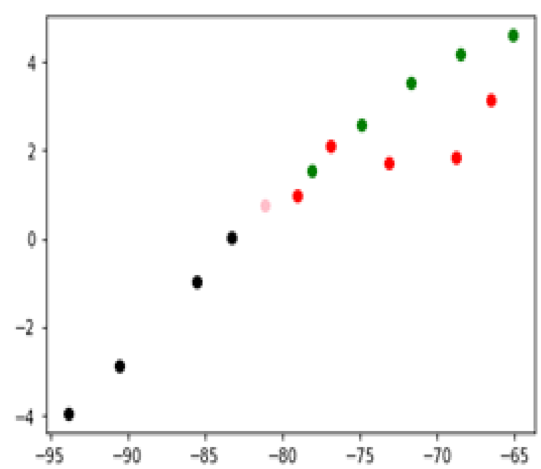
(a) Structured 1 time-step: LSTM-Direction one agents) with final distance error= 0.782



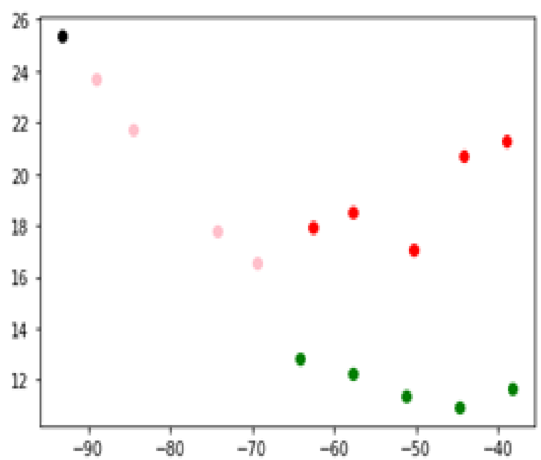
(b) Structured 1 time-step: NN-GA (one agents) with final distance error= 0.984



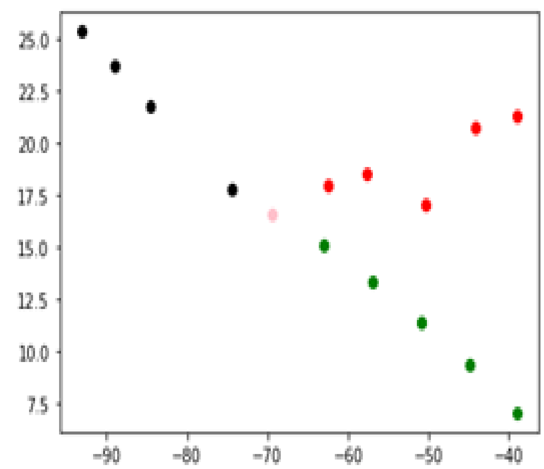
(c) Structured 5 time-steps:LSTM-Direction (one agents) with final distance error= 0.919



(d) Structured 5 time-steps:NN-GA (one agents)with final distance error= 1.963

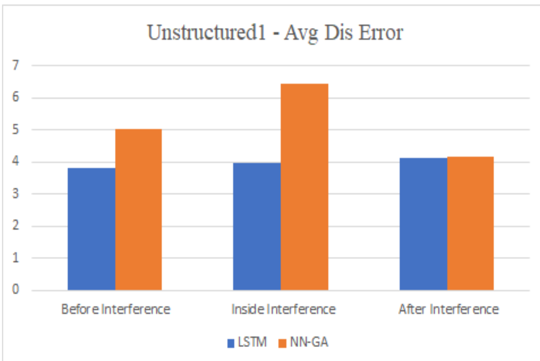


(e) Structured 10 time-steps:LSTM-Direction (one agents) with final distance error= 9.622

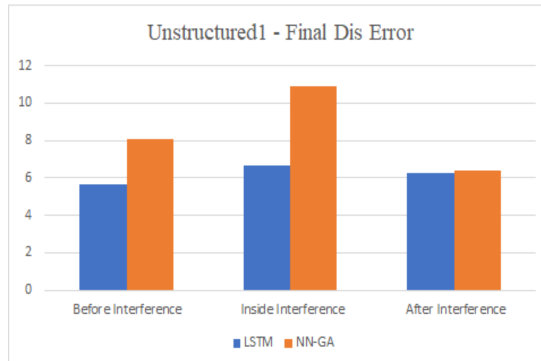


(f) Structured 10 time-steps:NN-GA (one agents) with final distance error= 14.276

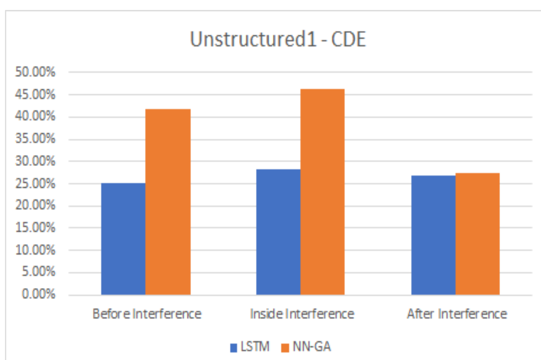
Figure 6.7: These figures illustrates an example of one agent from each place and includes 1 time-step, 5 time-steps, and 10 time-steps. The colors represent the following: black indicates the points where there is no observation or prediction; pink indicates an observation point; green indicates prediction points; and red indicates ground truth points.



(a) The average distance error in the LSTM- Direction and the NN-GA for before interference, inside interference, and after interference.

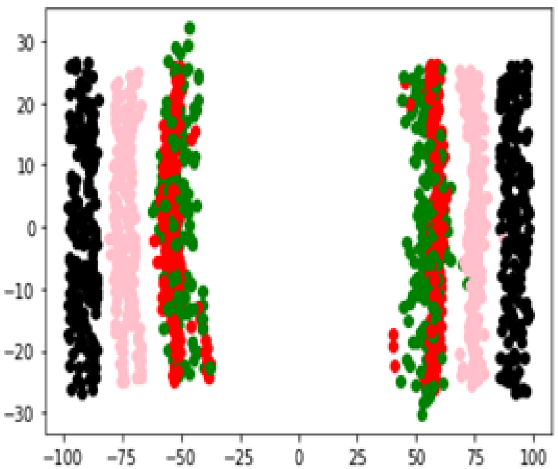


(b) The final distance error in the LSTM- Direction and the NN-GA for before interference, inside interference, and after interference.

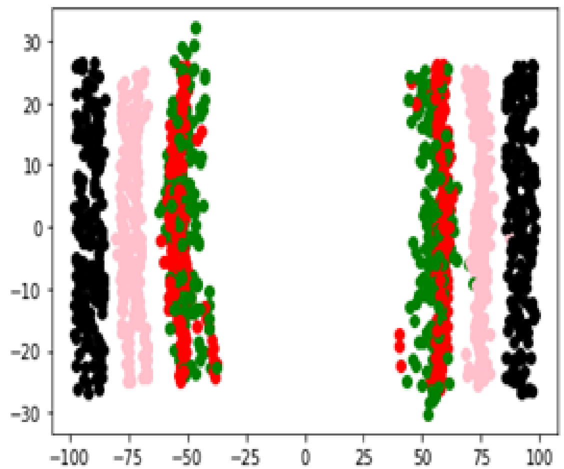


(c) The cumulative distance error in the LSTM-Direction and the NN-GA for before interference, inside interference, and after interference.

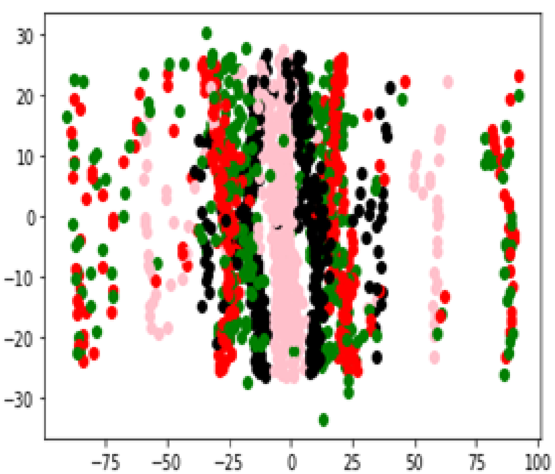
Figure 6.8: Unstructured (part 1): before interference, inside interference, and after interference ADE, FDE, and CDE for the LSTM-Direction model and the NN-GA model.



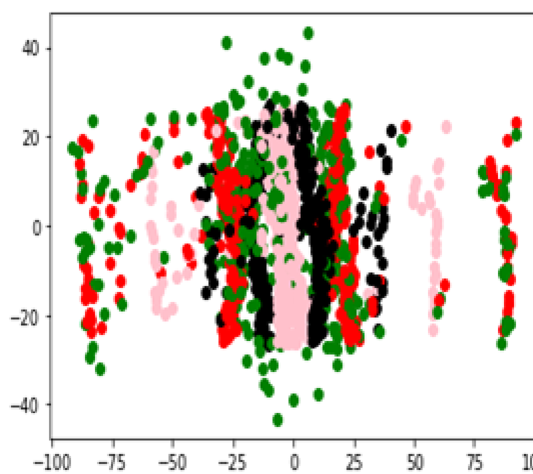
(a) Before the interference: LSTM-Direction (all agents)



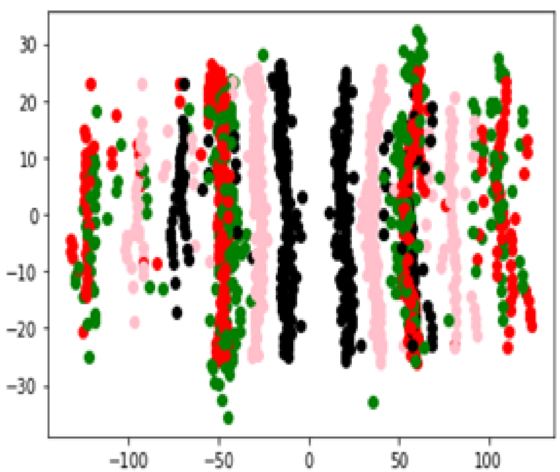
(b) Before the interference:NN-GA (all agents)



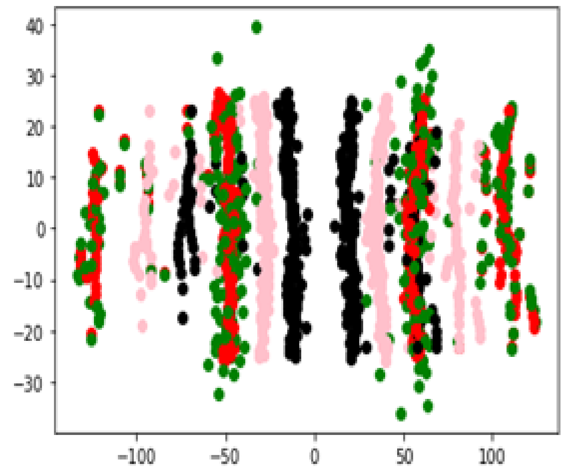
(c) During the interference:LSTM-Direction (all agents)



(d) During the interference:NN-GA (all agents)

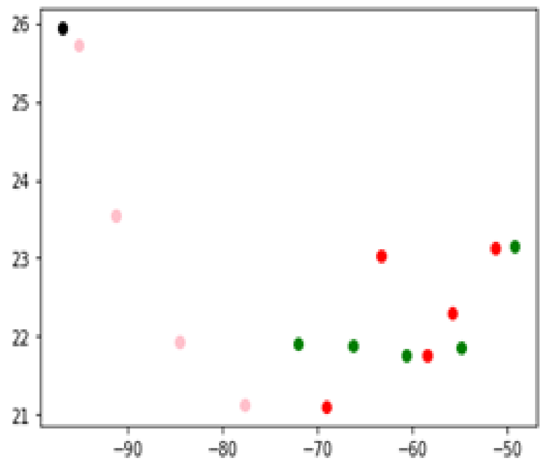


(e) After the interference:LSTM-Direction (all agents)

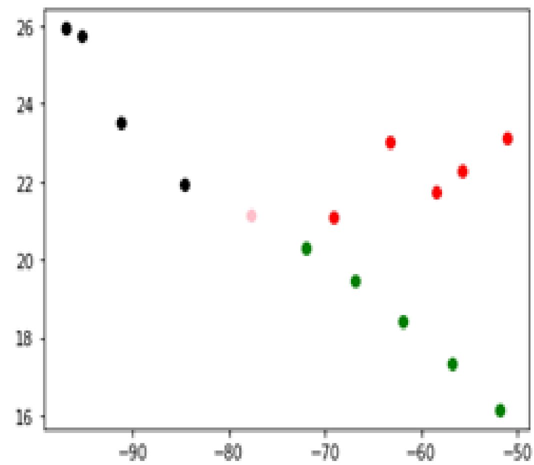


(f) After the interference:NN-GA (all agents)

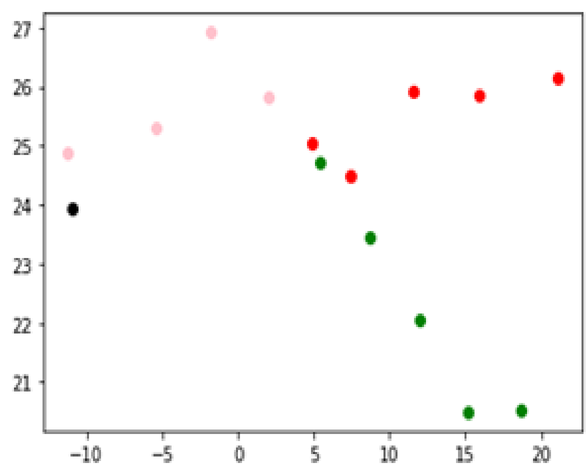
FIGURE 6.9: Unstructured crowded area (part 1): The following illustrates all agents in the three scenarios and include before the interference, during the interference, and after the interference. The colors represent the following: black indicates the points where the agents start moving; pink indicates the points where the prediction starts; green indicates the last step of prediction; and red indicates the last step of ground truth.



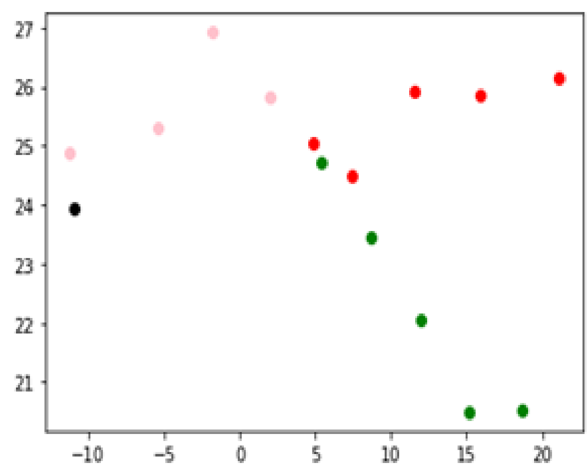
(a) Unstructured 1 before the interference: LSTM-Direction one agents) with final distance error= 1.883



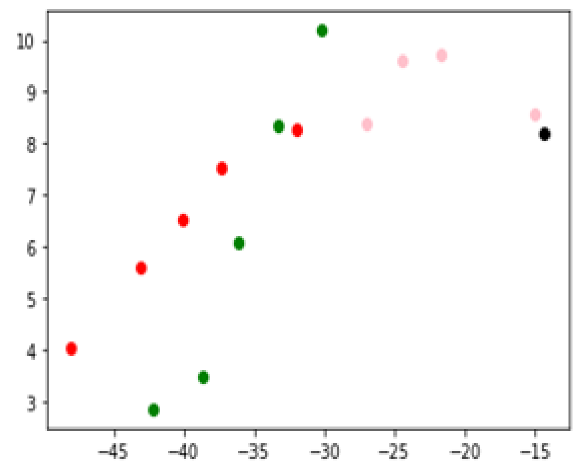
(b) Unstructured 1 before the interference: NN-GA (one agents) with final distance error= 7.015



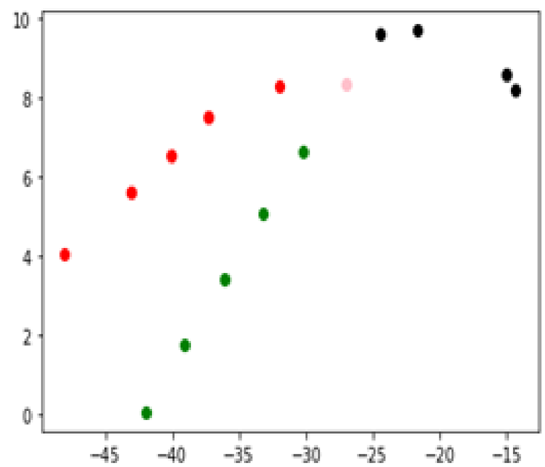
(c) Unstructured 1 during the interference:LSTM-Direction (one agents) with final distance error= 6.105



(d) Unstructured 1 during the interference:NN-GA (one agents)with final distance error= 7.233

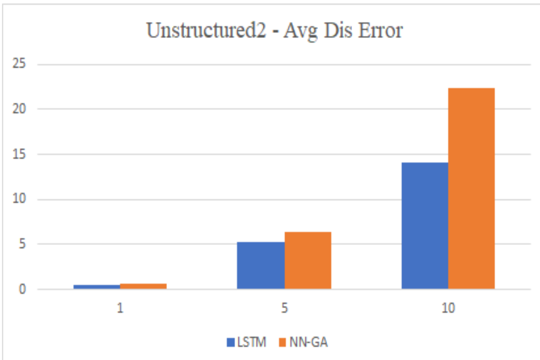


(e) Unstructured 1 after the interference:LSTM-Direction (one agents) with final distance error= 5.954

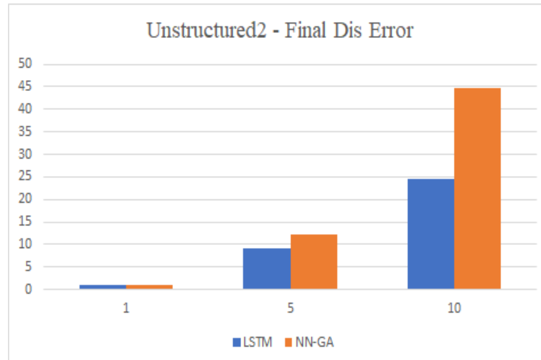


(f) Unstructured 1 after the interference:NN-GA (one agents) with final distance error= 7.283

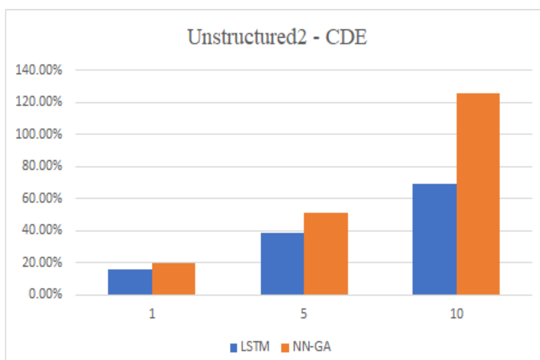
FIGURE 6.10: Unstructured crowded area (part 1): These figures illustrate an example of one agent from each place and includes before the interference, during the interference, and after the interference. The colors represent the following: black indicates the points where there is no observation or prediction; pink indicates an observation point; green indicates prediction points; and red indicates ground truth points.



(a) The average distance error in the LSTM-Direction and the NN-GA for 1 time-step, 5 time-steps, and 10 time-steps.

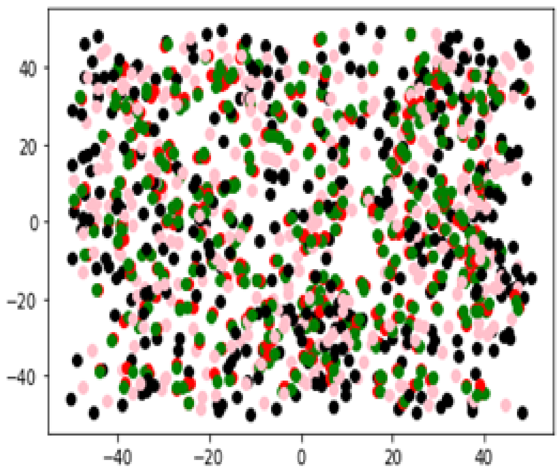


(b) The final distance error in the LSTM-Direction and the NN-GA for 1 time-step, 5 time-steps, and 10 time-steps.

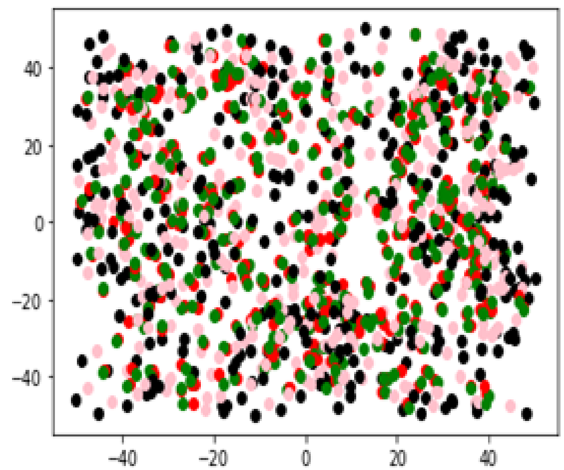


(c) The cumulative distance error in the LSTM-Direction and the NN-GA for 1 time-step, 5 time-steps, and 10 time-steps.

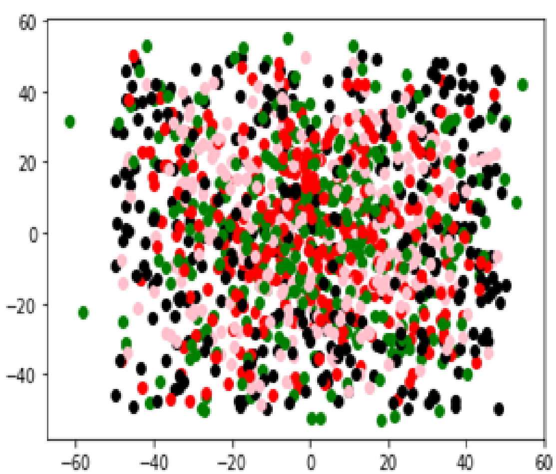
FIGURE 6.11: : Unstructured crowded area (part 2): 1 time-step, 5 time-steps, and 10 time-steps ADE, FDE, and CDE for the LSTM-Direction model and the NN-GA model.



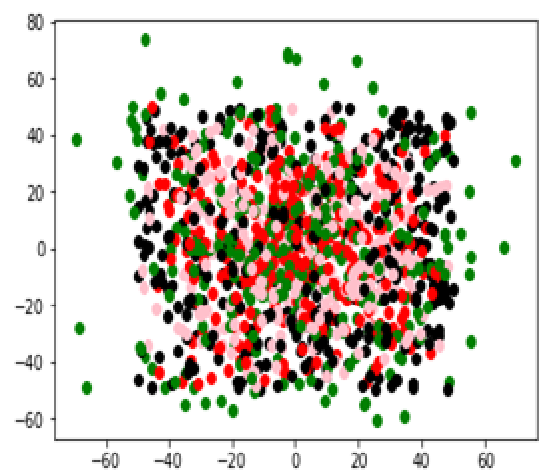
(a) 1 time-step: LSTM-Direction (all agents).



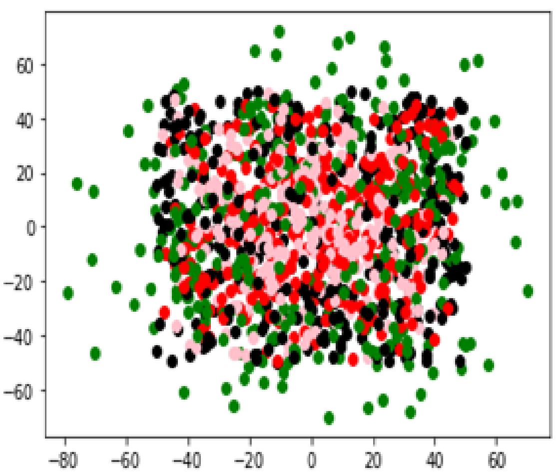
(b) 1 time-step: NN-GA (all agents).



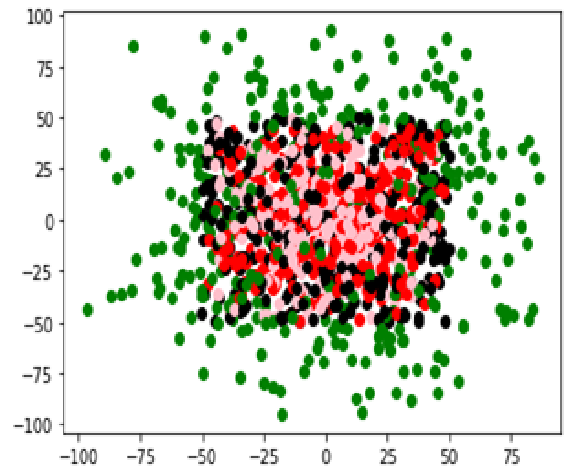
(c) 5 time-steps: LSTM-Direction (all agents).



(d) 5 time-steps:NN-GA (all agents).

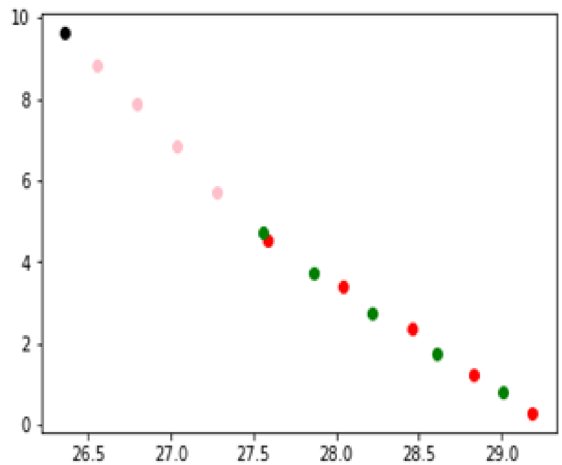


(e) 10 time-steps:LSTM-Direction (all agents)

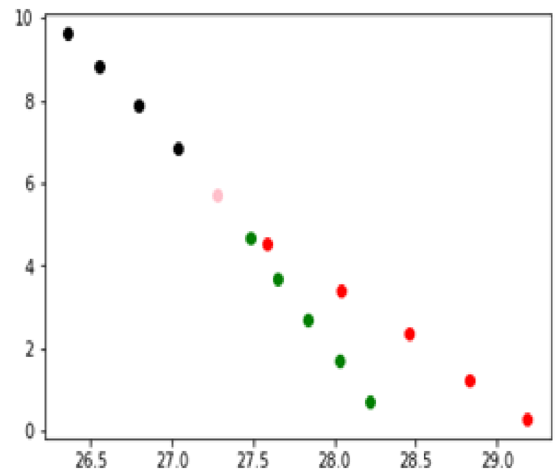


(f) 10 time-steps:NN-GA (all agents)

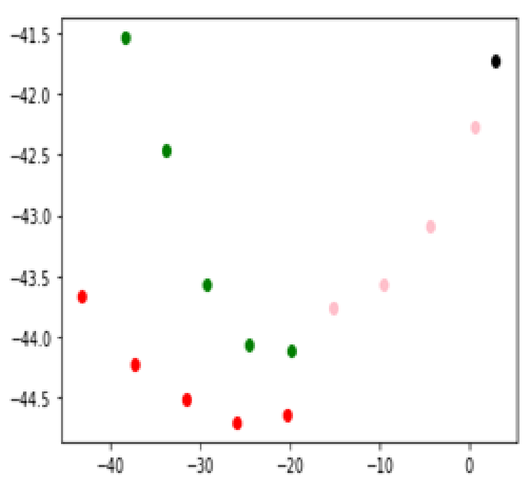
Figure 6.12: Unstructured crowded area (part2): The following illustrates all agents in the three scenarios that included 1 time-step, 5 time-steps, and 10 time-steps. The colors represent the following: black indicates the points where the agents start moving; pink indicates the points where the prediction starts; green indicates the last step of prediction; and red indicates the last step of ground truth.



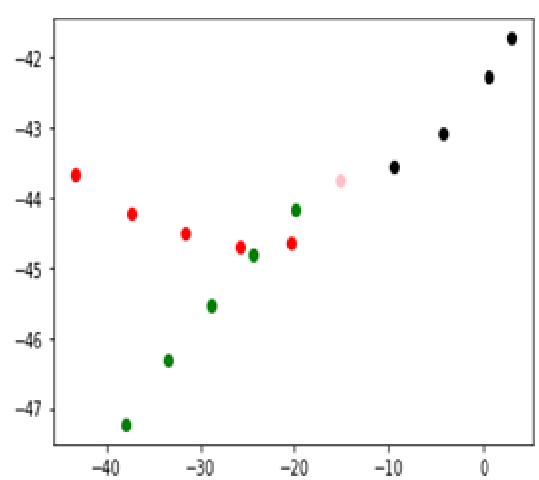
(a) Unstructured 2: 1 time-step: LSTM-Direction one agents) with final distance error= 0.542



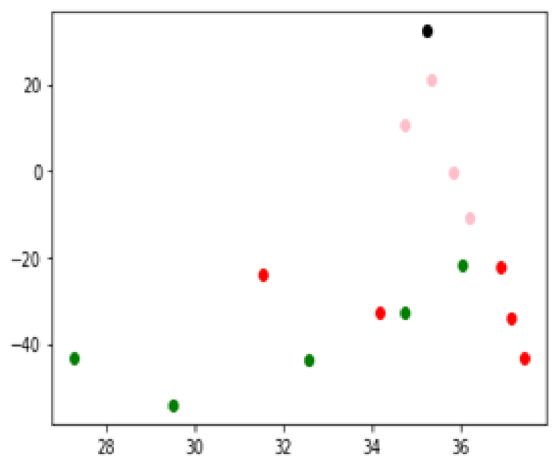
(b) Unstructured 2: 1 time-step: NN-GA (one agents) with final distance error= 1.059



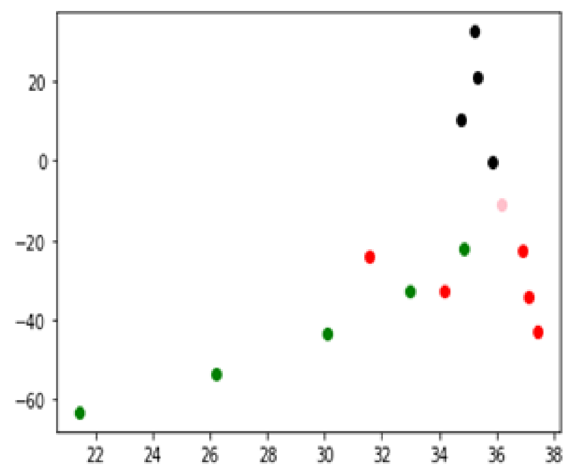
(c) Unstructured 2: 5 time-steps:LSTM-Direction (one agents) with final distance error= 5.243



(d) Unstructured 2: 5 time-steps:NN-GA (one agents)with final distance error= 6.346



(e) Unstructured 2: 10 time-steps:LSTM-Direction (one agents) with final distance error= 20.110



(f) Unstructured 2: 10 time-steps:NN-GA (one agents) with final distance error= 40.61

Figure 6.13: Unstructured crowded area (part 2): These figures illustrate an example of one agent from each place that included 1 time-step, 5 time-steps, and 10 time-steps. The colors represent the following: black indicates the points where there is no observation or prediction; pink indicates an observation point; green indicates prediction points; and red indicates ground truth points.

Table 6.3: Illustrates the difference between the LSTM-Direction and the NN-GA in the unstructured crowded area (part 1) and included before the interference, during the interference, and after the interference. Three metrics were used: ADE, FDE, and CDE.

Unstructured (part 1) - Before Interference 5 time-steps		
Metrics	LSTM	NN-GA
Average Displacement Error	3.789	5.031
Final Displacement Error	5.656	8.09
Cumulative Distance Error	25.25%	41.85%
Unstructured (part 1) - Inside Interference 5 time-steps		
Metrics	LSTM	NN-GA
Average Displacement Error	3.976	6.449
Final Displacement Error	6.671	10.867
Cumulative Distance Error	28.34%	46.24%
Unstructured (part 1) - After Interference 5 time-steps		
Metrics	LSTM	NN-GA
Average Displacement Error	4.11	4.17
Final Displacement Error	6.272	6.394
Cumulative Distance Error	26.85%	27.37%

6.7 MERGING PATHS

Another scenario was deemed worth testing was merging paths, when agents from two different paths merge into one path because this is a common crowd scenario and can lead to serious problems. We choose two locations from this scenario for predicting the agents' motion: one from before the paths merge; and the second from after the agents have merged into one path. Both methods produced better

TABLE 6.4: This table illustrates the difference between the LSTM-Direction and the NN-GA in the unstructured crowded area (part 2) that included 1 time-step, 5 time-steps, and 10 time-steps. Three metrics were used: ADE, FDE, and CDE.

Unstructured 2 - 1 Time-step		
Metrics	LSTM	NN-GA
Average Displacement Error	0.513	0.61
Final Displacement Error	0.881	1.067
Cumulative Distance Error	15.98%	19.33%
Unstructured 2 - 5 Time-steps		
Metrics	LSTM	NN-GA
Average Displacement Error	5.313	6.346
Final Displacement Error	9.175	12.126
Cumulative Distance Error	38.89%	51.40%
Unstructured 2 - 10 Time-steps		
Metrics	LSTM	NN-GA
Average Displacement Error	14.123	22.309
Final Displacement Error	24.522	44.614
Cumulative Distance Error	68.84%	125.24%

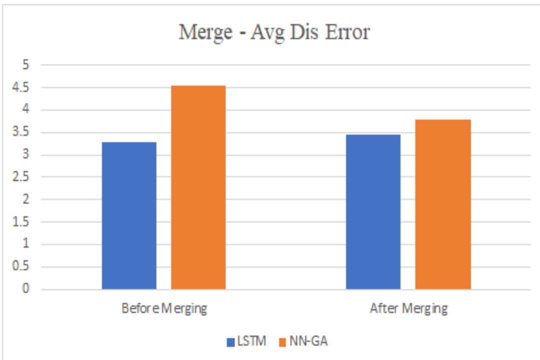
Table 6.5: This table illustrates the difference between the LSTM-Direction and the NN-GA in the merging paths that included before merging, and after merging. Three metrics were used: ADE, FDE, and CDE.

Merge - Before merging		
Metrics	LSTM	NN-GA
Average Displacement Error	3.288	4.555
Final Displacement Error	5.142	7.193
Cumulative Distance Error	29.68%	41.52%
Merge - After merging		
Metrics	LSTM	NN-GA
Average Displacement Error	3.44	3.783
Final Displacement Error	4.193	5.97
Cumulative Distance Error	16.99%	24.20%

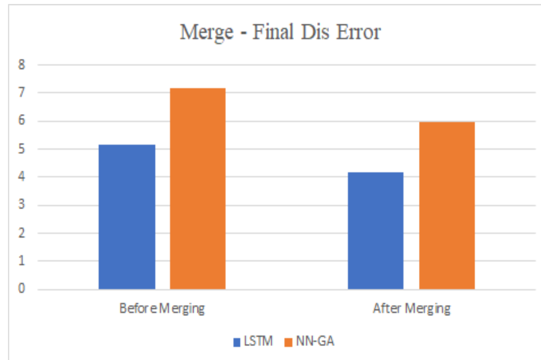
results after the paths merged, but the LSTM-Direction surpassed the NN-GA in both scenarios. It was surprising to us that the results for after the merging of paths had better overall outcomes than for before merging. The explanation for this is that there seemed to be less comfort with crowd density before merging than the comfort level with crowd density after merging; there is a narrowing of paths before merging. This resulted in less accuracy of prediction for before merging.

6.8 waypoints path

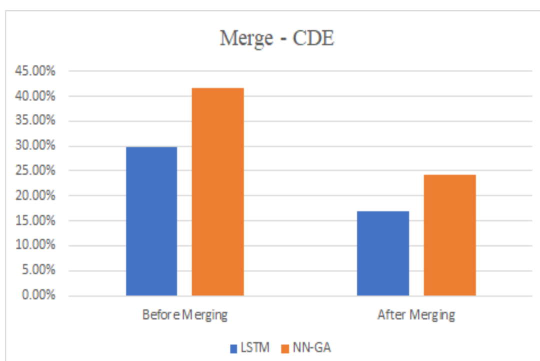
The last scenario we tested was a path with waypoints, in which agents stopped at some points along their way, then continued moving to their targets. This is one of the most common scenarios that occurs during the Hajj, where people move from one of the ritual sites of the pilgrimage to another ritual area. Two strategies were applied in this scenario: one was 5 time-steps, in which the total time of prediction



(a) The average distance error in the LSTM-Direction and the NN-GA for before merging and after merging.

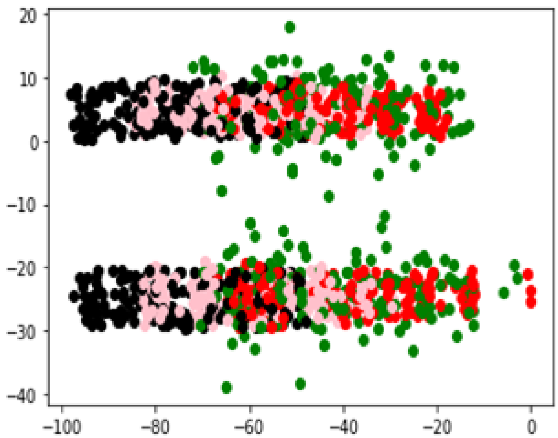


(b) The final distance error in the LSTM-Direction and the NN-GA for before merging and after merging.

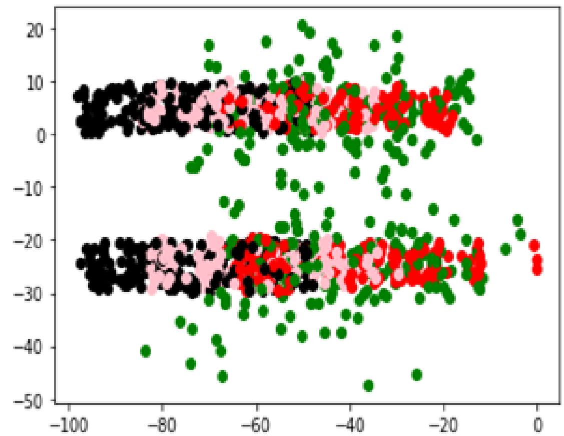


(c) The cumulative distance error in theLSTM-Direction and the NN-GA for before merging and after merging.

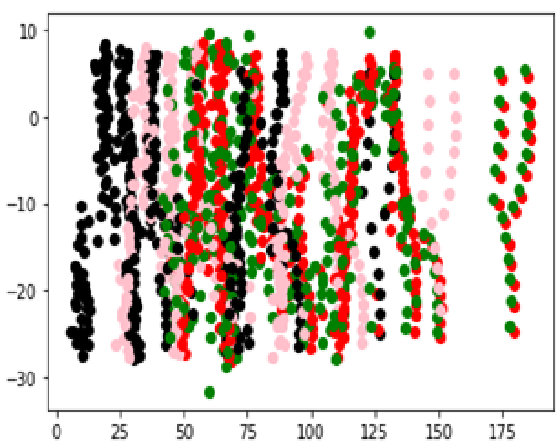
FIGURE 6.14: Merge: before merging and after merging ADE, FDE, and CDE for the LSTM-Direction model and the NN-GA model.



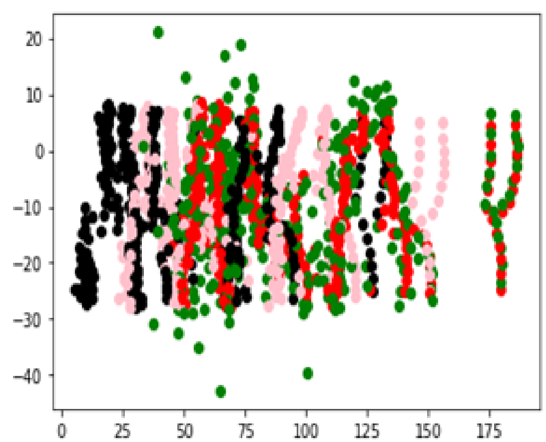
(a) Before Merging: LSTM-Direction (all agents).



(b) Before Merging: NN-GA (all agents).

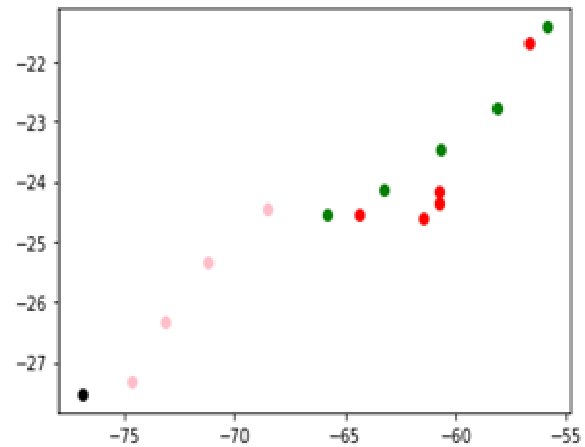


(c) After Merging: LSTM-Direction (all agents).

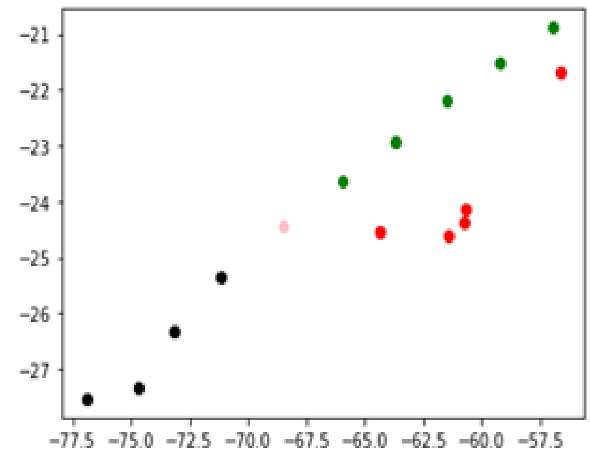


(d) After Merging:NN-GA (all agents).

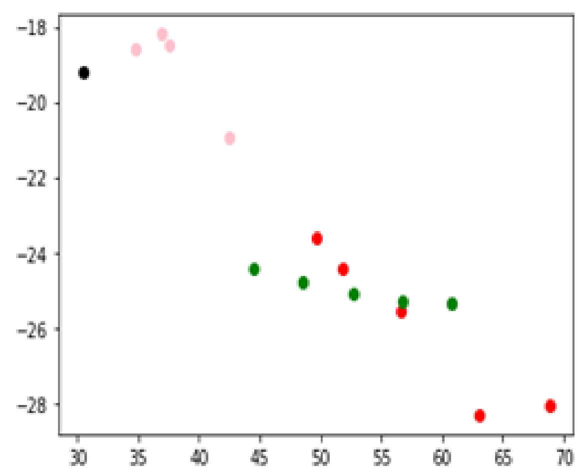
FIGURE 6.15: Merge: The following illustrates all agents in the two scenarios that included before merging, and after merging. The colors represent the following: black indicates the points where the agents start moving; pink indicates the points where the prediction starts; green indicates the last step of prediction; and red indicates the last step of ground truth.



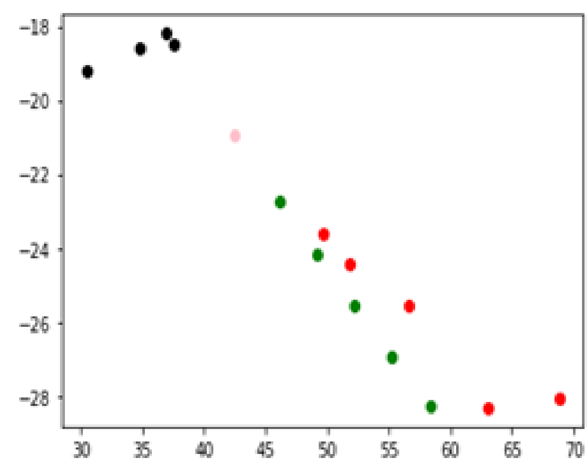
(a) Merge: Before merging, LSTM-Direction one agents) with final distance error= 0.840



(b) Merge: Before merging, NN-GA (one agents) with final distance error= 0.882



(c) Merge: After merging,:LSTM-Direction (one agents) with final distance error= 8.476



(d) Merge: After merging,NN-GA (one agents)with final distance error= 10.488

Figure 6.16: Merge: These figures illustrate an example of one agent from each place that included before merging, and after merging. The colors represent the following: black indicates the points where there is no observation or prediction; pink indicates an observation point; green indicates prediction points; and red indicates ground truth points.

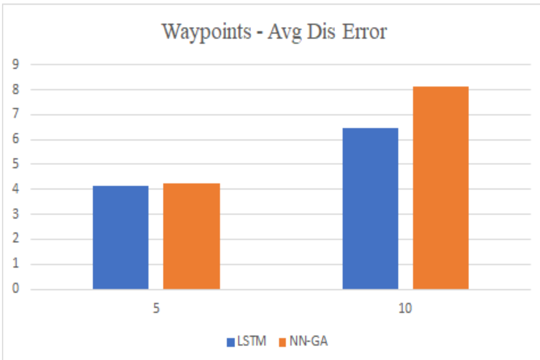
TABLE 6.6: Waypoints path:This table illustrates the difference between the LSTM-Direction and the NN-GA in thewaypoints path that included 5 time-steps, and 10 time-steps. Three metrics were used: ADE, FDE, and CDE.

Waypoints - 5 time-steps		
Metrics	LSTM	NN-GA
Average Displacement Error	4.118	4.236
Final Displacement Error	6.613	7.021
Cumulative Distance Error	29.18%	30.98%
Waypoints - 10 time-steps		
Metrics	LSTM	NN-GA
Average Displacement Error	6.471	8.105
Final Displacement Error	10.587	13.926
Cumulative Distance Error	21.79%	28.67%

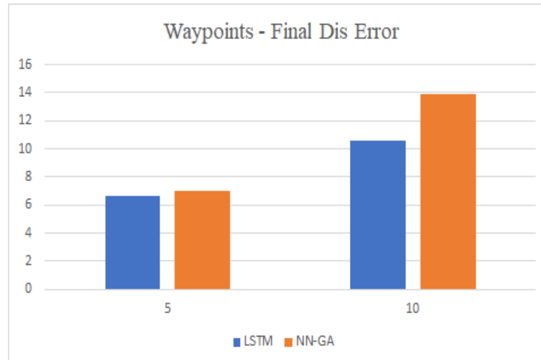
was 25 time-steps; the second strategy is was 10 time-steps, in which the total time of prediction is was 50 time-steps. The outcomes had a growth error in ADE, and FDE when prediction time lengthened. In contrast, the CDE decreased when the prediction time lengthened. The LSTM-Direction was superior to the NN-GA in both approaches.

6.9 CONCLUSIONS

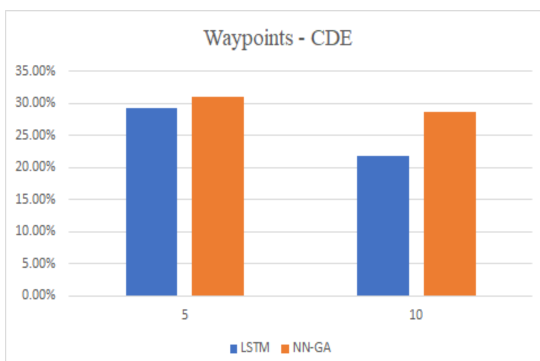
Avoiding static, or dynamic obstacles plays a very important part in human movement prediction. Our work covered most of the scenarios that happen in daily life. We focused on the scenarios that usually have higher crowd density. We tested two methods, the LSTM-Direction and the NN-GA, on all scenarios. We conclusively found that the LSTM-Direction outperformed the NN-GA in every scenario. Although the LSTM-Direction is superior, but the results in some scenarios are close. That



(a) The average distance error in the LSTM-Direction and the NN-GA for 5 time-steps and 10 time-steps.

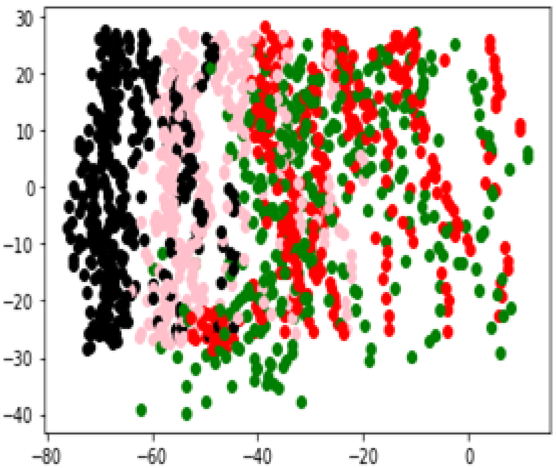


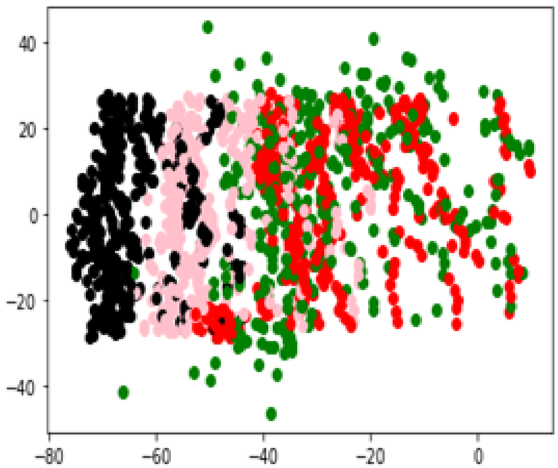
(b) The final distance error in the LSTM-Direction and the NN-GA for 5 time-steps and 10 time-steps.



(c) The cumulative distance error in the LSTM-Direction and the NN-GA for 5 time-steps and 10 time-steps.

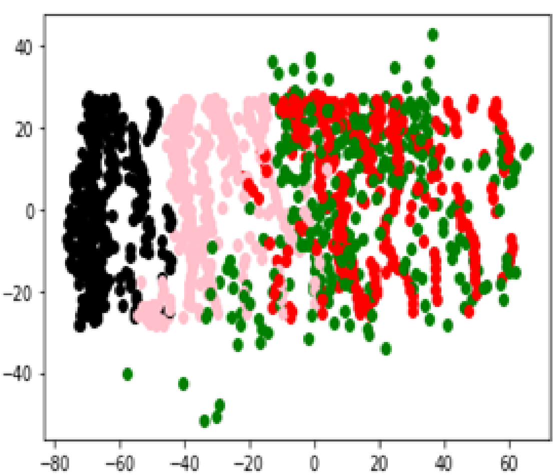
FIGURE 6.17: Waypoints: 5 time-steps and 10 time-steps ADE, FDE, and CDE for the LSTM-Direction model and the NN-GA model.

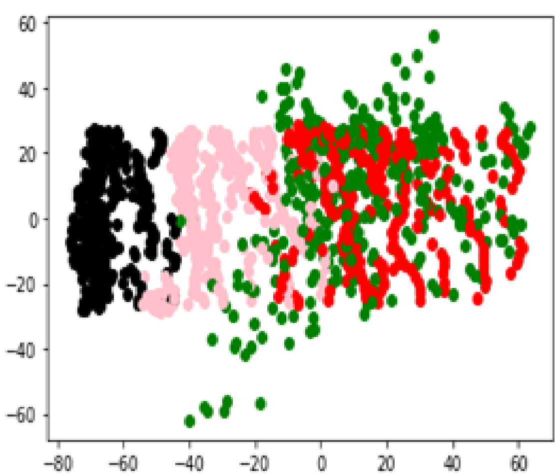




(a) Waypoints 5 time-steps: LSTM-Direction (all agents).

(b) Waypoints 5 time-steps: NN-GA (all agents).

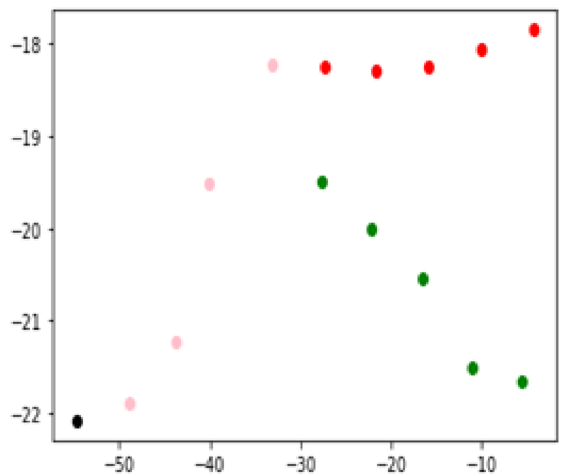




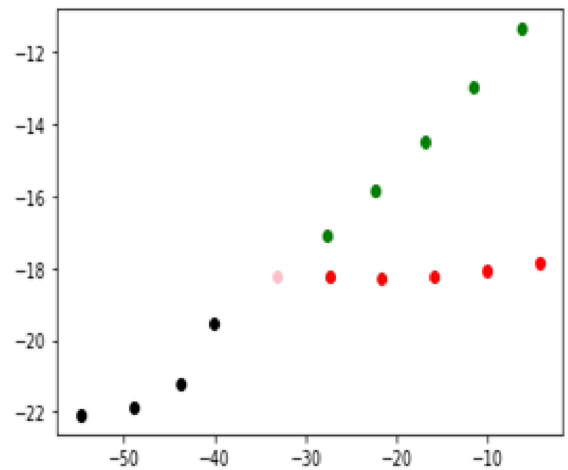
(c) Waypoints 10 time-steps: LSTM-Direction (all agents).

(d) Waypoints 10 time-steps:NN-GA (all agents).

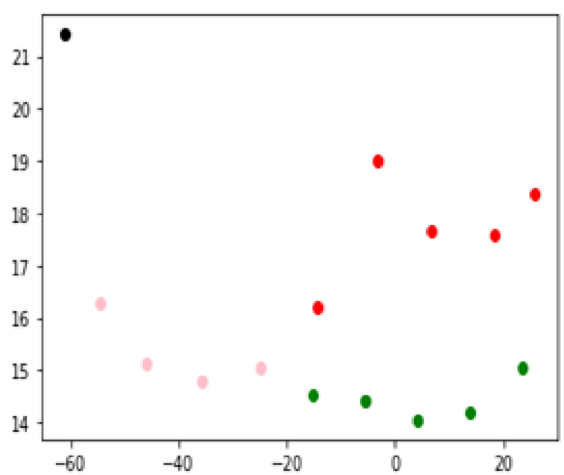
Figure 6.18: Waypoints: The following illustrate all the agents in the two scenarios that included 5 time-steps, and 10 time-steps. The colors represent the following: black indicates the points where the agents start moving; pink indicates the points where the prediction starts; green indicates the last step of prediction; and red indicates the last step of ground truth.



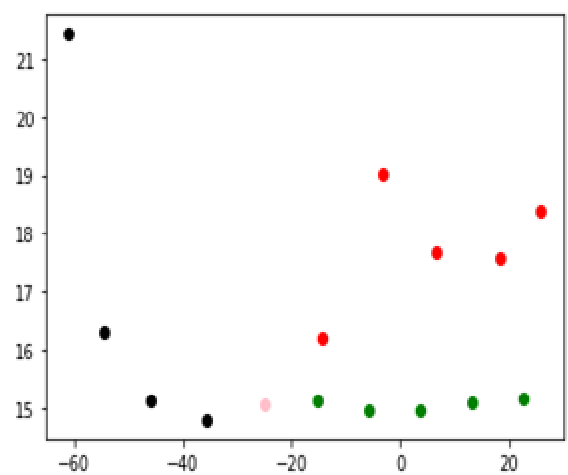
(a) Waypoints: 5 time-steps, LSTM-Direction one agents) with final distance error= 4.004



(b) Waypoints: 5 time-steps, NN-GA (one agents) with final distance error= 6.813



(c) Waypoints: 10 time-steps,LSTM-Direction (one agents) with final distance error= 4.008



(d) Waypoints: 10 time-steps,NN-GA (one agents)with final distance error= 4.357

FIGURE 6.19: Waypoints: These figures illustrate an example of one agent from each place that included 5 time-steps, and 10 time-steps. The colors represent the following: black indicates the points where there is no observation or prediction; pink indicates an observation point; green indicates prediction points; and red indicates ground truth points.

encourages us to test more scenarios in the future, since we cannot be certain that the LSTM-Direction will always surpass the NN-GA. Both produced better Cumulative Distance Error (CDE) outcomes when the prediction was longer, with the exception of the unstructured crowded area (part 2). Yet, we can say the LSTM-Direction outperformed NN-GA in all of the scenarios we tested.

CHAPTER 7

Conclusion

7.1 CONCLUSION AND FUTURE WORK

The primary goal of delving into this research was to develop novel techniques for predicting human movements. Our research produced a new metric, new methods, and a new perspective on this problem. This research addressed a number of specific issues in the field.

First, our investigation compared the performance of two algorithms: neural networks trained via a genetic algorithms (NN-GA), and long short-term memory (LSTM). The objectives was to discover if the NN-GA is effective in the prediction of human movements. In our first investigation in Chapter 3, we used the NN-GA with two types of scenarios: structured crowded areas, and unstructured crowded areas. To test the effectiveness of using the NN-GA, we compared the two methods, one of which used the NN-GA to predict the future steps, and the other that did not use any algorithm for prediction. Our results showed how that the NN-GA algorithm was effective in trajectory prediction.

Second, we expanded our work to include many scenarios, several behaviors, and more than one method using the two prediction algorithms: NN-GA and LSTM-Direction. We demonstrated that the LSTM-Direction outperformed NN-GA in all scenarios and in different behaviors in Chapter 5. Our tests included several scenarios: low density and high density crowds, structured and unstructured crowds, and a short period of prediction time or a long period of prediction time. For behaviors we tested the flocking model (FM) and social force model (SFM).

Third, we demonstrated that each scenario leads to different prediction results, which was demonstrated using three metrics, average distance error (ADE), final distance error (FDE), and cumulative distance error (CDE). The CDE is the most convincing metric of the three; it can address questions regarding prediction over different time periods with different datasets and methodologies by normalizing the error over time. Using CDE we observed different prediciton accuracy in different scenarios, which demonstrated that in making predictions it is necessary to differentiate between scenarios and take each one individually. The outcomes of this research was published in Alajlan *et al*. (2021). Our evaluations were based on the three metrics, ADE, FDE, and CDE. In the ADE, we took the average distance between the predicted step and the ground truth at each step as in Pellegrini *et al*. (2009).

$$ADE(\hat{P}, P) = \frac{1}{T}\sum_{i=1}^{T}\sqrt{(X_i - \hat{X}_i)^2 + (Y_i - \hat{Y}_i)^2} \tag{7.1}$$

The second metric, FDE, measures the average distance between the final predicted step and the final ground truth.

$$FDE(\hat{P}, P) = \sqrt{(X_F - \hat{X}_F)^2 + (Y_F - \hat{Y}_F)^2} \tag{7.2}$$

The third metric, CDE, measures the average distance between the final predicted step and the final ground truth divided by the average distance between the final ground truth location and the location from which we started the prediction. We then multiplied that result by 100 to obtain the percentage error in distance as a function of the distance moved.

$$CDE(\hat{P}, P) = \frac{1}{N}\sum_{i=1}^{N}\frac{\sqrt{(X_F - \hat{X}_F)^2 + (Y_F - \hat{Y}_F)^2}}{\sqrt{(X_F - \hat{X}_S)^2 + (Y_F - \hat{Y}_S)^2}} \times 100 \tag{7.3}$$

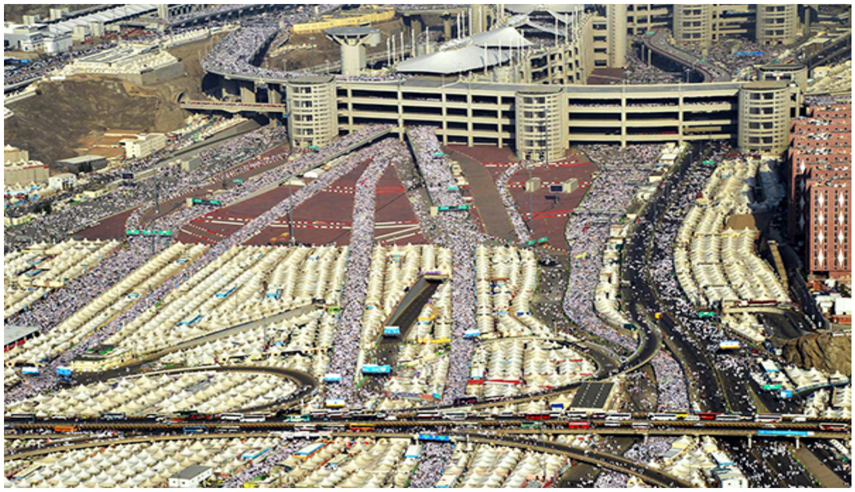
7.2 SUMMARY OF IMPLEMENTATIONS

We used the Python programming language, a high-level programming language, as our primary language to execute and train the prediction algorithms in this research. For the algorithms, we used python open-source library Keras Chollet *et al*. (2015) in both algorithms. The Python open-source library PyGAD that works with keras was used with the NN-GA. The second language in this research was the Netlogo, a multi-agent programmable modeling environment, to model the behaviors of agents. The movement behaviors were based on existing simulations used in published papers: the Social forces Model (SFM) introduced by Helbing and Molnar (1995), and the Flocking Model (FM) that introduced by Reynolds (1987). During simulation we recorded the linear difference between any two steps for each agent as the distance and speed for that specific agent at that time-step. The base information for every agent at each time-step was based on the following data: the agent ID, the specific time, and the (X,Y) for this agent. We then transferred all this information to a text files for analysis in Python.

7.3 FUTURE WORKS FOR PREDICTING HUMAN MOVEMENTS IN DENSE CROWDS

This research has focused on dense crowds and the small spaces that pedestrians may have to move through in specific situations. We have studied how people behave in these situations, and based on that study, developed and tested techniques for predicting pedestrian's movement in crowds.

Working closely on behalf of the pilgrims will be a focus in future research. The Saudi Arabian authorities are going to require each pilgrim at the Hajj to wear a watch or wristband beginning in 2030. These tracking devices will provide the opportunity to locate each person at every spot during the five days of Hajj. Since this research focuses on dense crowds, we plan to work with the authorities to optimize the prediction of the pilgrims' movement. Future research focus will be on the Jamarat in the Mina ritual place, which is where the three walls (formerly pillars) are located. In this cases a pilgrim stops at the first pillar to throw stones seven times and then, in an orderly fashion, moves to the next two pillars to do the same. We have designed a scenario in Chapter 6, and specifically in section 6.8 that has waypoints for the period when the agents stop at specified point and then move to either another point or to his/her destination.



(a) Image of Jamarat building and bridges that let people go through.
Image from:
https://www.nytimes.com/interactive/2015/09/24/world/middleeast/mecca-mina-stampede-hajj-maps.html

(b) Another Image of Jamarat building and bridges.
Image from: https://www.bbc.com/news/world-middle-east-34361122

Figure 7.1: Examples of two major regimes of nominal crowd flow.

Table 7.1: Social distancing: This table illustrates the difference between the LSTM-Direction and the NN-GA in the social distancing scenario includes 1 time-step. Three metrics were used: ADE, FDE, and CDE.

Structured- 1 time-step With social distancing		
Metrics	LSTM	NN-GA
Average Displacement Error	1.066	2.814
Final Displacement Error	1.803	4.461
Cumulative Distance Error	51.63%	127.74%

One of the reasons why we will focus our future work on the Jamarat ritual place, is because it has had many sad accidents that have resulted a large number of deaths. Even though the accidents at the Jamarat building have lessened during the last ten years, the Saudi Arabian authorities intend to raise the pilgrims' numbers to twice the 2019 number, which was 2.5 million pilgrims, by the year 2030.

Another future focus will be on the real actual structured crowded areas in the Hajj, where people move from one point to another with the same direction and the same goal. These paths are considered as important areas for more organization during the Hajj. Figures 7.1 show how pilgrims move in these pedestrians' paths.

7.4 SOCIAL DISTANCING SCENARIO

In the last two years, we have seen how important social distancing is, specifically in gatherings such as sport events. Based on Brauer (2021) more variants of COVID-19 are expected in the future, and that the virus will evolve, and mutate, which means we can expect the situation of mask-wearing and distancing socially to continue at some point in the future. We tested one scenario incorporating social distancing, and we intend to test more scenarios in our future work. In the social forces model social distancing is modeled by increasing the social avoidance force, causing individuals to be more strongly 'repelled' by each other. Obtaining a real dataset for social distancing in the future will help us to be more accurate in our predictions, especially in crowds. Table 7.1 illustrates the difference of results between the NN-GA and the LSTM-Direction for one scenario in a structured crowded area (1 time-step). As in previous experiments LSTM performed much better than NN-GA in predicting human movement. Given the risk of future pandemics, combined with worldwide population growth, and ever increasing crowds, improved methods to predict human movement in large crowds under varying distancing scenarios will be critical for protecting human lives and thus a research area of growing importance.

Bibliography

Ahmad A., Rahman M.A., Rehman F.U., Lbath A., Afyouni I., Khelil A., Hussain S.O., Sadiq B., and Wahiddin M.R. 2014. A framework for crowd-sourced data collection and context-aware services in hajj and umrah. *In* 2014 IEEE/ACS 11th International Conference on Computer Systems and Applications (AICCSA), pages 405–412, IEEE.

Al-Hashedi A.H., Arshad M.R.H.M., Baharudin A.S., Mohamed H.H., and Osman O. 2014. Strategic information systems planning for rfid implementation in hajj management systems. *In* Proceedings of the 6th International Conference on Information Technology and Multimedia, pages 146–150, IEEE.

Al-Hashedi A.H., Arshad M.R.M., Baharudin A.S., and Mohamed H.H. 2013. Rfid applications in hajj management system. *In* 2013 IEEE International Conference on RFID-Technologies and Applications (RFID-TA), pages 1–6, IEEE.

Al-Kodmany K. 2013. Crowd management and urban design: New scientific approaches. Urban Design International 18:282–295.

Alahi A., Goel K., Ramanathan V., Robicquet A., Fei-Fei L., and Savarese S. 2016. Social lstm: Human trajectory prediction in crowded spaces. *In* Proceedings of the IEEE conference on computer vision and pattern recognition, pages 961–971.

Alajlan, Edris, and Soule. 2021. Predicting human movements using machine learning. *In* The 23rd International Conference on Artificial Intelligence, Springer.

Alajlan, Heckndorn E., and Soule. 2020. Using neural networks and genetic algorithms for predicting human movement in crowds. *In* The 22nd Int'l Conf on Artificial Intelligence, Springer.

Alarabiya. 2018. Hajj vision 2030. `https://english.alarabiya.net/webtv/reports/2018/08/20/WATCH-Saudi-Arabia-introduces-digitalized-plan-for-Hajj-2030`.

Alshalani H., Alnaghaimshi N., and Eljack S. 2020. Ict system for crowd management: Hajj as a case study. *In* 2020 International Conference on Computing and Information Technology (ICCIT-1441), pages 1–5, IEEE.

Alsubhy A.M., Abi Sen A.A., Alahmadi B.A., Bahbouh N.M., and Abi Sen H.A. 2020. A model for tracking people and property in crowds. *In* 2020 7th International Conference on Computing for Sustainable Global Development (INDIACom), pages 244–248, IEEE.

Amirian J., Van Toll W., Hayet J.B., and Pettré J. 2019. Data-driven crowd simulation with generative adversarial networks. *In* Proceedings of the 32nd International Conference on Computer Animation and Social Agents, pages 7–10.

Aoude G., Joseph J., Roy N., and How J. 2011. Mobile agent trajectory prediction using bayesian nonparametric reachability trees. *In* Infotech@ Aerospace 2011, page 1512.

Bahdanau D., Cho K., and Bengio Y. 2014. Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473 .

Bartoli F., Lisanti G., Ballan L., and Del Bimbo A. 2018. Context-aware trajectory prediction. *In* 2018 24th International Conference on Pattern Recognition (ICPR), pages 1941–1946, IEEE.

Bighashdel A. and Dubbelman G. 2019. A survey on path prediction techniques for vulnerable road users: From traditional to deep-learning approaches. *In* 2019 IEEE Intelligent Transportation Systems Conference (ITSC), pages 1039–1046, IEEE.

Binsalleeh H., Mohammed N., Sandhu P.S., Aljumah F., and Fung B.C. 2009. Using rfid tags to improve pilgrimage management. *In* 2009 International Conference on Innovations in Information Technology (IIT), pages 1–5, IEEE.

Blanke U., Tröster G., Franke T., and Lukowicz P. 2014. Capturing crowd dynamics at large scale events using participatory gps-localization. *In* 2014 IEEE Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), pages 1–7, IEEE.

Brauer S. 2021. Rethinking our covid-19 strategy. Precision Nanomedicine 4:738–749.

Byrisetty N.C. 2013. Agent-based modeling to simulate the movement of a flock of birds. Masters thesis, North Dakota State University.

Camillen F., Caprì S., Garofalo C., Ignaccolo M., Inturri G., Pluchino A., Rapisarda A., and Tudisco S. 2009. Multi agent simulation of pedestrian behavior in closed spatial environments. *In* 2009 IEEE Toronto International Conference Science and Technology for Humanity (TIC-STH), pages 375–380, IEEE.

Chan W., Jaitly N., Le Q., and Vinyals O. 2016. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. *In* 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 4960–4964, IEEE.

Chen K., Song X., Han D., Sun J., Cui Y., and Ren X. 2020. Pedestrian behavior prediction model with a convolutional lstm encoder–decoder. Physica A: Statistical Mechanics and its Applications 560:125132.

Cheng B., Xu X., Zeng Y., Ren J., and Jung S. 2018. Pedestrian trajectory prediction via the social-grid lstm model. The Journal of Engineering 2018:1468–1474.

Choi I., Song H., and Yoo J. 2019. Deep learning based pedestrian trajectory prediction considering location relationship between pedestrians. *In* 2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIC), pages 449–451, IEEE.

Chollet F. *et al.* 2015. Keras. `https://github.com/fchollet/keras`.

Darwin C. 1964. On the origin of species: A facsimile of the first edition, vol. 49. Harvard University Press.

Das A., Kolvig-Raun E.S., and Kjærgaard M.B. 2020. Accurate trajectory prediction in a smart building using recurrent neural networks. *In* Adjunct Proceedings of the 2020 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2020 ACM International Symposium on Wearable Computers, pages 619–628.

Deka A., Narayanan V.K., Miyashita T., and Hagita N. 2018. Adaptive attention-aware pedestrian trajectory prediction for robot planning in human environments. Unpublished .

Dendorfer P., Osep A., and Leal-Taixé L. 2020. Goal-gan: Multimodal trajectory prediction based on goal position estimation. *In* Proceedings of the Asian Conference on Computer Vision.

Dewi M., Hariadi M., and Purnomo M.H. 2011. Simulating the movement of the crowd in an environment using flocking. *In* 2011 2nd International Conference on Instrumentation, Communications, Information Technology, and Biomedical Engineering, pages 186–191, IEEE.

Eiben A.E., Smith J.E., *et al*. 2003. Introduction to evolutionary computing, vol. 53. Springer.

Ek-Hobak A., Sanchez A., and Hayet J.B. 2020. Evaluation of output representations in neural network-based trajectory predictions systems. *In* 2020 International Conference on Systems, Signals and Image Processing (IWSSIP), pages 447–452, IEEE.

Gambs S., Killijian M.O., and del Prado Cortez M.N. 2012. Next place prediction using mobility markov chains. *In* Proceedings of the First Workshop on Measurement, Privacy, and Mobility, pages 1–6.

Gupta A., Johnson J., Fei-Fei L., Savarese S., and Alahi A. 2018. Social gan: Socially acceptable trajectories with generative adversarial networks. *In* Proceedings of the IEEE conference on computer vision and pattern recognition, pages 2255–2264.

Gupta J.N. and Sexton R.S. 1999. Comparing backpropagation with a genetic algorithm for neural network training. Omega 27:679–684.

Haddad S. and Lam S.K. 2020. Graph2kernel grid-lstm: A multi-cued model for pedestrian trajectory prediction by learning adaptive neighborhoods. arXiv preprint arXiv:2007.01915 .

Haddad S., Wu M., Wei H., and Lam S.K. 2019. Situation-aware pedestrian trajectory prediction with spatio-temporal attention model. arXiv preprint arXiv:1902.05437 .

Hamandi M., D'Arcy M., and Fazli P. 2019. Deepmotion: Learning to navigate like humans. *In* 2019 28th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN), pages 1–7, IEEE.

Hargrove W.W. and Westervelt J.D. 2012. An implementation of the pathway analysis through habitat (path) algorithm using netlogo. *In* Ecologist-Developed Spatially-Explicit Dynamic Landscape Models, pages 211–222, Springer.

Hasan I., Setti F., Tsesmelis T., Belagiannis V., Amin S., Del Bue A., Cristani M., and Galasso F. 2019. Forecasting people trajectories and head poses by jointly reasoning on tracklets and vislets. IEEE transactions on pattern analysis and machine intelligence 43:1267–1278.

Hasan I., Setti F., Tsesmelis T., Del Bue A., Galasso F., and Cristani M. 2018. Mx-lstm: mixing tracklets and vislets to jointly forecast trajectories and head poses. *In* Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 6067–6076.

Helbing D. and Molnar P. 1995. Social force model for pedestrian dynamics. Physical review E 51:4282.

Henry P., Vollmer C., Ferris B., and Fox D. 2010. Learning to navigate through crowded environments. *In* 2010 IEEE International Conference on Robotics and Automation, pages 981–986, IEEE.

Hochreiter S. and Schmidhuber J. 1997. Long short-term memory. Neural computation 9:1735–1780.

Huang Y., Bi H., Li Z., Mao T., and Wang Z. 2019. Stgat: Modeling spatial-temporal interactions for human trajectory prediction. *In* Proceedings of the IEEE/CVF international conference on computer vision, pages 6272–6281.

Huynh M. and Alaghband G. 2019. Trajectory prediction by coupling scene-lstm with human movement lstm. *In* International Symposium on Visual Computing, pages 244–259, Springer.

Jamil S., Basalamah A., Lbath A., and Youssef M. 2015. Hybrid participatory sensing for analyzing group dynamics in the largest annual religious gathering. *In* Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing, pages 547–558.

Johansson A., Helbing D., Al-Abideen H.Z., and Al-Bosta S. 2008. From crowd dynamics to crowd safety: a video-based analysis. Advances in Complex Systems 11:497–527.

Ju C., Wang Z., Long C., Zhang X., Cong G., and Chang D.E. 2019. Interaction-aware kalman neural networks for trajectory prediction. arXiv preprint arXiv:1902.10928 .

Karasev V., Ayvaci A., Heisele B., and Soatto S. 2016. Intent-aware long-term prediction of pedestrian motion. *In* 2016 IEEE International Conference on Robotics and Automation (ICRA), pages 2543–2549, IEEE.

Kong Y. and Fu Y. 2018. Human action recognition and prediction: A survey. arXiv preprint arXiv:1806.11230 .

Kosaraju V., Sadeghian A., Martín-Martín R., Reid I., Rezatofighi H., and Savarese S. 2019. Social-bigat: Multimodal trajectory forecasting using bicycle-gan and graph attention networks. Advances in Neural Information Processing Systems 32.

Koshak N. and Fouda A. 2008. Analyzing pedestrian movement in mataf using gps and gis to support space redesign. *In* The 9th international conference on design and decision support systems in architecture and urban planning.

Krausz B. and Bauckhage C. 2012. Loveparade 2010: Automatic video analysis of a crowd disaster. Computer Vision and Image Understanding 116:307–319.

Lakhdari A., Bouguettaya A., Mistry S., and Neiat A.G. 2020. Composing energy services in a crowdsourced iot environment. IEEE Transactions on Services Computing .

Lee N., Choi W., Vernaza P., Choy C.B., Torr P.H., and Chandraker M. 2017. Desire: Distant future prediction in dynamic scenes with interacting agents. *In* Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 336–345.

Lisotto M., Coscia P., and Ballan L. 2019. Social and scene-aware trajectory prediction in crowded spaces. *In* Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, pages 0–0.

Manh H. and Alaghband G. 2018. Scene-lstm: A model for human trajectory prediction. arXiv preprint arXiv:1808.04018 .

Mehran R., Oyama A., and Shah M. 2009. Abnormal crowd behavior detection using social force model. *In* 2009 IEEE Conference on Computer Vision and Pattern Recognition, pages 935–942, IEEE.

Miller G.F., Todd P.M., and Hegde S.U. 1989. Designing neural networks using genetic algorithms. *In* ICGA, vol. 89, pages 379–384.

Minoura H., Hirakawa T., Yamashita T., and Fujiyoshi H. 2019. Path predictions using object attributes and semantic environment. *In* VISIGRAPP (5: VISAPP), pages 19–26.

Mitchell R.O., Rashid H., Dawood F., and AlKhalidi A. 2013. Hajj crowd management and navigation system: People tracking and location based services via integrated mobile and rfid systems. *In* 2013 International Conference on Computer Applications Technology (ICCAT), pages 1–7, IEEE.

Mohamed H.H., Arshad M.R.H.M., Rashid N.A., Zainol Z., Husain W., Abd Majid O., Ghazali M., Rahim M.Y.A., and Mahmod A.R.H. 2013. M-umrah: An android-based application to help pilgrims in performing umrah. *In* 2013 International Conference on Advanced Computer Science Applications and Technologies, pages 385–389, IEEE.

Mohamed M.F., Shabayek A.E.R., and El-Gayyar M. 2019. Iot-based framework for crowd management. *In* Mobile Solutions and Their Usefulness in Everyday Life, pages 47–61, Springer.

Mohammad Y. and Ades Y. 2009. Crowd management with rfid & wireless technologies. *In* Proceedings of First International Conference on Networks & Communications, IEEE Computer Society Washington, DC, USA.

Mohandes M., Haleem M.A., Abul-Hussain A., and Balakrishnan K. 2011. Pilgrims tracking using wireless sensor network. *In* 2011 IEEE Workshops of International Conference on Advanced Information Networking and Applications, pages 325–328, IEEE.

Montana D.J., Davis L., *et al*. 1989. Training feedforward neural networks using genetic algorithms. *In* IJCAI, vol. 89, pages 762–767.

Naser M., Rafie M., Budiarto R., and Alsalihy W. 2010. Security considerations in embedding rfid in'hajj'system. European Journal of Scientific Research 42:133–138.

Nasser N., Anan M., Awad M.F.C., Bin-Abbas H., and Karim L. 2017. An expert crowd monitoring and management framework for hajj. *In* 2017 International Conference on Wireless Networks and Mobile Communications (WINCOM), pages 1–8, IEEE.

Ozturk O., Yamasaki T., and Aizawa K. 2010. Detecting dominant motion flows in unstructured/structured crowd scenes. pages 3533–3536.

Pelechano N. and Malkawi A. 2008. Evacuation simulation models: Challenges in modeling high rise building evacuation with cellular automata approaches. Automation in construction 17:377–385.

Pellegrini S., Ess A., Schindler K., and Van Gool L. 2009. You'll never walk alone: Modeling social behavior for multi-target tracking. *In* 2009 IEEE 12th International Conference on Computer Vision, pages 261–268, IEEE.

Peng Y., Zhang G., Shi J., Xu B., and Zheng L. 2021. Sra-lstm: Social relationship attention lstm for human trajectory prediction. arXiv preprint arXiv:2103.17045 .

Pfeiffer M., Paolo G., Sommer H., Nieto J., Siegwart R., and Cadena C. 2018. A data-driven model for interaction-aware pedestrian motion prediction in object cluttered environments. *In* 2018 IEEE International Conference on Robotics and Automation (ICRA), pages 5921–5928, IEEE.

Pluchino A., Garofalo C., Inturri G., Rapisarda A., and Ignaccolo M. 2013. Agent-based simulation of pedestrian behaviour in closed spaces: a museum case study. arXiv preprint arXiv:1302.7153 .

Qolomany B., Al-Fuqaha A., Benhaddou D., and Gupta A. 2017. Role of deep lstm neural networks and wi-fi networks in support of occupancy prediction in smart buildings. *In* 2017 IEEE 19th International Conference on High Performance Computing and Communications; IEEE 15th International Conference on Smart City; IEEE 3rd International Conference on Data Science and Systems (HPCC/SmartCity/DSS), pages 50–57, IEEE.

Quan R., Zhu L., Wu Y., and Yang Y. 2021. Holistic lstm for pedestrian trajectory prediction. IEEE transactions on image processing 30:3229–3239.

Rahman A., Hassanain E., and Hossain M.S. 2017. Towards a secure mobile edge computing framework for hajj. IEEE Access 5:11768–11781.

Rehder E., Wirth F., Lauer M., and Stiller C. 2018. Pedestrian prediction by planning using deep neural networks. *In* 2018 IEEE International Conference on Robotics and Automation (ICRA), pages 1–5, IEEE.

Reynolds C.W. 1987. Flocks, herds and schools: A distributed behavioral model. *In* Proceedings of the 14th annual conference on Computer graphics and interactive techniques, pages 25–34.

Ridel D., Rehder E., Lauer M., Stiller C., and Wolf D. 2018. A literature review on the prediction of pedestrian behavior in urban scenarios. *In* 2018 21st International Conference on Intelligent Transportation Systems (ITSC), pages 3105–3112, IEEE.

Rodriguez M., Ali S., and Kanade T. 2009. Tracking in unstructured crowded scenes. *In* 2009 IEEE 12th International Conference on Computer Vision, pages 1389–1396, IEEE.

Rozenberg R., Gesnouin J., and Moutarde F. 2021. Asymmetrical bi-rnn for pedestrian trajectory encoding. arXiv preprint arXiv:2106.04419 .

Rudenko A., Palmieri L., Herman M., Kitani K.M., Gavrila D.M., and Arras K.O. 2020. Human motion trajectory prediction: A survey. The International Journal of Robotics Research 39:895–935.

Sabarish B., Karthi R., and Gireeshkumar T. 2015. A survey of location prediction using trajectory mining. *In* Artificial Intelligence and Evolutionary Algorithms in Engineering Systems, pages 119–127, Springer.

Schmidt S. and Faerber B. 2009. Pedestrians at the kerb–recognising the action intentions of humans. Transportation research part F: traffic psychology and behaviour 12:300–310.

Schubert J., Ferrara L., Hörling P., and Walter J. 2008. A decision support system for crowd control. *In* Proceedings of the 13th International Command and Control Research Technology Symposium, pages 1–19.

Schubert J. and Suzic R. 2007. Decision support for crowd control: Using genetic algorithms with simulation to learn control strategies. *In* MILCOM 2007-IEEE Military Communications Conference, pages 1–7, IEEE.

Shao L., Cai Z., Liu L., and Lu K. 2017. Performance evaluation of deep feature learning for rgb-d image/video classification. Information Sciences 385:266–283.

Shi X., Shao X., Fan Z., Jiang R., Zhang H., Guo Z., Wu G., Yuan W., and Shibasaki R. 2020. Multimodal interaction-aware trajectory prediction in crowded space. *In* Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pages 11982–11989.

Shi X., Shao X., Guo Z., Wu G., Zhang H., and Shibasaki R. 2019. Pedestrian trajectory prediction in extremely crowded scenarios. Sensors 19:1223.

Shi X., Shao X., Wu G., Zhang H., Guo Z., Jiang R., and Shibasaki R. 2021. Social dpf: Socially acceptable distribution prediction of futures. *In* Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pages 2550–2557.

Shirazi M.S. and Morris B. 2015. Observing behaviors at intersections: A review of recent studies & developments. *In* 2015 IEEE Intelligent Vehicles Symposium (IV), pages 1258–1263, IEEE.

Singhal A. and Indu S. 2020. Enhancing accuracy for human trajectory forecasting in crowded scenes. *In* Proceedings of the 2020 3rd International Conference on Image and Graphics Processing, pages 146–152.

Sun L., Yan Z., Mellado S.M., Hanheide M., and Duckett T. 2018. 3dof pedestrian trajectory prediction learned from long-term autonomous mobile robot deployment data. *In* 2018 IEEE International Conference on Robotics and Automation (ICRA), pages 5942–5948, IEEE.

Tisue S. and Wilensky U. 2004. Netlogo: A simple environment for modeling complexity. *In* International conference on complex systems, vol. 21, pages 16–21, Boston, MA.

Varsamopoulos S., Bertels K., Almudever C.G., *et al*. 2018. Designing neural network based decoders for surface codes. arXiv preprint arXiv:1811.12456 .

Vemula A., Muelling K., and Oh J. 2018. Social attention: Modeling attention in human crowds. *In* 2018 IEEE international Conference on Robotics and Automation (ICRA), pages 4601–4607, IEEE.

Wang A.Y. and Wang L. 2017. Walking step prediction based on ga optimized neural network algorithm. *In* 2017 2nd IEEE International Conference on Computational Intelligence and Applications (ICCIA), pages 295–298, IEEE.

wikipedia. 2019. Incidents during the hajj. `https://en.wikipedia.org/wiki/Incidents_during_the_Hajj`.

Xu M., Ge Z., Jiang X., Cui G., Lv P., Zhou B., and Xu C. 2019. Depth information guided crowd counting for complex crowd scenes. Pattern Recognition Letters 125:563–569.

Xu Y., Piao Z., and Gao S. 2018. Encoding crowd interaction with deep neural network for pedestrian trajectory prediction. *In* Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 5275–5284.

Xue H., Huynh D., and Reynolds M. 2019. Location-velocity attention for pedestrian trajectory prediction. *In* 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), pages 2038–2047, IEEE.

Xue H., Huynh D.Q., and Reynolds M. 2017. Bi-prediction: pedestrian trajectory prediction based on bidirectional lstm classification. *In* 2017 International Conference on Digital Image Computing: Techniques and Applications (DICTA), pages 1–8, IEEE.

Xue H., Huynh D.Q., and Reynolds M. 2018. Ss-lstm: A hierarchical lstm model for pedestrian trajectory prediction. *In* 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), pages 1186–1194, IEEE.

Yamin M. 2008. A framework for improved hajj management and future research. ENTIC Bull 2.

Yamin M. 2019. Managing crowds with technology: cases of hajj and kumbh mela. International journal of information technology 11:229–237.

Yamin M., Al-Ahmadi H.M., and Al Muhammad A. 2016. Integrating social media and mobile apps into hajj management. *In* 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom), pages 1368–1372, IEEE.

Yamin M., Mohammadian M., Huang X., and Sharma D. 2008. Rfid technology and crowded event management. *In* 2008 International Conference on Computational Intelligence for Modelling Control & Automation, pages 1293–1297, IEEE.

Yang F. and Peters C. 2019a. App-lstm: Data-driven generation of socially acceptable trajectories for approaching small groups of agents. *In* Proceedings of the 7th International Conference on Human-Agent Interaction, pages 144–152.

Yang F. and Peters C. 2019b. Appgan: Generative adversarial networks for generating robot approach behaviors into small groups of people. *In* 2019 28th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN), pages 1–8, IEEE.

Yang L., Dawson C.W., Brown M.R., and Gell M. 2006. Neural network and ga approaches for dwelling fire occurrence prediction. Knowledge-Based Systems 19:213–219.

Yokojima Y. and Sakai T. ???? Learning pedestrian crowd as clusters using abstracted features .

Zhang P., Ouyang W., Zhang P., Xue J., and Zheng N. 2019. Sr-lstm: State refinement for lstm towards pedestrian trajectory prediction. *In* Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 12085–12094.

APPENDIX A

Supplementary Information to Chapter 2

A.1 TABLE OF PAPERS

Table A.1: The table briefly summarizes all papers to provide the comprehensive aspects of each paper. Overall approach, output, and the metrics used are the labels that describe the method. The list of metrics' abbreviations can be found at the end of table.

Paper	Overall Approach	Objective/Output	Metrics
Zhang *et al*. (2019)	State refinement module for LSTM (SR-LSTM)	To consider the current neighbor states for timely inference. To introduce a socially aware information selection mechanism to support the extraction of social information.	ADE, FDE
Xue *et al*. (2019)	Location-Velocity Attention LSTM	To combine the location and velocity of information optimally to enhance trajectory prediction accuracy.	ADE, FDE

Continued on next page

Table a.1 – continued from previous page

Paper	Overall Approach	Objective/Output	Metrics
Shi *et al.* (2020)	Multimodal interaction-aware trajectory prediction	To consider spatial social awareness and temporal movements of the agents when predicting trajectories. To utilize coordinate transformation to represent the relative motion between people.	ADE, FDE
Dendorfer *et al.* (2020)	The development of the Goal-GAN model and then evaluating it using different datasets.	Multimodal trajectory prediction	ADE, FDE
Singhal and Indu (2020)	Developing an algorithm combined with more features and then evaluating it using datasets	Enhancing the accuracy of human trajectory forecasting by using visualizations of crowded scenes	ADE, FDE
Ek-Hobak *et al.* (2020)	Examining whether the use of residual output representations enhanced neural network-based trajectory forecasting systems	Accuracy of network-based trajectory forecasting systems using residual output representations	ADE, FDE

Continued on next page

Table a.1 – continued from previous page

Paper	Overall Approach	Objective/Output	Metrics
Hasan *et al.* (2019)	Developing MX-LSTM model and then examining whether the consideration of head poses improved the accuracy of pedestrian trajectory prediction.	To introduce MX-LSTM model and to assess whether head poses enhance the accuracy of forecasting	ADE, FDE
Das *et al.* (2020)	Comparison of two neural networks.	To compare the accuracy of the gated recurrent unit (GRU) with the long short-term memory (LSTM) models	Mean square error and mean absolute error
Haddad and Lam (2020)	Developing Graph-to-Kernel LSTM and evaluating its performance	To model pedestrian trajectories by learning adaptive neighborhoods using Graph-to-Kernel LSTM	ADE, FDE
Huang *et al.* (2019)	Development of STGAT (spatial-temporal graph attention network) and evaluating its performance.	To develop and utilize STGAT in predicting future human movement in crowded spaces	ADE, FDE
Yokojima and Sakai	Conceptualization of pedestrian trajectory prediction by perceiving pedestrians as crowds and using abstracted features	To develop a framework for modeling pedestrian trajectories as crowds and using abstracted features	N/A

Continued on next page

Table a.1 – continued from previous page

Paper	Overall Approach	Objective/Output	Metrics
Hasan *et al.* (2018)	MX-LSTM	To integrate vislets into the MX-LSTM model to increase the accuracy of trajectory prediction.	ADE, FDE, MAE
Minoura *et al.* (2019)	Utilization of object attributes and semantic environment together with LSTM	To consider object attributes and semantic environments to improve trajectory prediction accuracy.	ADE, FDE
Chen *et al.* (2020)	Convolutional LSTM encoder–decoder	To utilize multi-channel tensors to represent information relating to pedestrians. To develop an end-to-end fully convolutional LSTM model for encoding and decoding information and predicting trajectories.	ADE, FDE
Cheng *et al.* (2018)	Social-Grid LSTM model	To improve trajectory prediction using the Social-Grid LSTM model.	ADE, FDE
Qolomany *et al.* (2017)	Deep LSTM neural networks	To utilize Wi-Fi network data in a deep LSTM neural network to predict human occupancy in a smart building.	RMSE

Continued on next page

Table a.1 – continued from previous page

Paper	Overall Approach	Objective/Output	Metrics
Manh and Alaghband (2018)	Scene-LSTM	To utilize scene information to enhance the prediction of movement trajectories using LSTM.	ADE, FDE, NDE
Huynh and Alaghband (2019)	Coupling scene-LSTM with human movement LSTM	To develop a forecasting system that combines pedestrian-LSTM with scene-LSTM to predict pedestrian trajectories in static crowded scenes.	ADE, FDE, NDE
Haddad *et al*. (2019)	Spatio-temporal graph-based LSTM	To develop an algorithm that considers the interaction between people and dynamic elements, as well as static objects, in trajectory prediction.	ADE, FDE
Lisotto *et al*. (2019)	LSTM-based model for social and scene-aware trajectory prediction	To incorporate semantics of the environment, human interactions, and past observations into an LSTM model to predict human trajectories	ADE, FDE
Vemula *et al*. (2018)	Social attention model	To consider the relative significance of each pedestrian in a crowd to predict human trajectories.	ADE, FDE

Continued on next page

Table a.1 – continued from previous page

Paper	Overall Approach	Objective/Output	Metrics
Shi *et al.* (2021)	Social-DPF	To consider jointly different interacting motion sequences and forecasts of future multimodal socially-acceptable distributions.	ADE, FDE
Gupta *et al.* (2018)	Social generative adversarial network (GAN)	To develop and evaluate a novel pooling mechanism to aggregate information, observe motion histories, and forecast future behavior. To train the model against a recurrent discriminator to predict socially plausible future trajectories. To utilize a novel variety loss to enable diverse predictions.	ADE, FDE
Kosaraju *et al.* (2019)	Social-BiGAT	To predict trajectories by modeling pedestrian social interactions in each scene.	ADE, FDE
Peng *et al.* (2021)	SRA-LSTM	To model social relationships using the temporal correlation of relative positions.	ADE, FDE

Continued on next page

Table a.1 – continued from previous page

Paper	Overall Approach	Objective/Output	Metrics
Xue *et al.* (2018)	SS-LSTM	To develop a model comprising three LSTMs (scene, social setting, and pedestrian) and using scene information to enhance human trajectory prediction accuracy. To utilize a circular-shaped neighborhood to improve prediction accuracy.	ADE, FDE
Sun *et al.* (2018)	Using 3D LiDAR pose trajectories instead of 2d positions, and more inputs data (such as the rotation) for trajectory prediction, and orientation prediction.	Trajectory prediction, orientation Prediction.	ADE, AEDE
Xu *et al.* (2018)	Each pedestrian has an LSTM to form his/her motion; then they scaling the movement characteristic depending on the affinity space.	Trajectory prediction.	ADE, FDE, ANDE

Continued on next page

Table a.1 – continued from previous page

Paper	Overall Approach	Objective/Output	Metrics
Pfeiffer *et al*. (2018)	Using three factors as inputs, velocity, surrounding static objects, and neighboring pedestrians. Modeling LSTM to predict pedestrian behaviors.	Trajectory forecasting.	AEM
Deka *et al*. (2018)	Using the positions, and the surrounding environment information to feed the SRNN.	Trajectory prediction.	ADE, FDE
?	The LSTM receives the spatial interaction context that is coupled with the encoding to catch the movement feature.	Trajectory prediction.	ADE, FDE
Yang and Peters (2019b)	LSTM-based GAN with a group interactivities that combines the agent's position and his/her head orientation in the collection	Trajectory prediction.	ADE, FDE
Yang and Peters (2019a)	Using App-LSTM, which is to obtain a (GIM), and taking into consideration the positions, and orientations.	Generating factual paths for pedestrians.	ADE, FDE
Rozenberg *et al*. (2021)	Using asymmetrical bidirectional RRNs to predict future trajectories.	Predicting future pedestrians' trajectories.	ADE, FDE, Col-I, Col-II

Continued on next page

Table a.1 – continued from previous page

Paper	Overall Approach	Objective/Output	Metrics
Xue *et al*. (2017)	Bi-Prediction: Pedestrian Trajectory Prediction based on Bidirectional LSTM Classification	Prediction of pedestrian trajectory based on destination.	ADE, FDE
Bartoli *et al*. (2018)	Context-Aware Trajectory Prediction	To predict human trajectory using context aware model.	ADE
Amirian *et al*. (2019)	Data-Driven Crowd Simulation with Generative Adversarial Networks	To predict human trajectory in a data-driven crowd. To make simulations for crowded areas.	The distribution of entry points
Choi *et al*. (2019)	Predict Pedestrian Trajectory Using LSTM	Predict pedestrian trajectory.	ADE
Hamandi *et al*. (2019)	Robots navigating like humans using deep MoTIon algorithm.	Reduce robot-human collisions.	SPD, DTW, Proximity, number of collisions, and target location.
Xue *et al*. (2017)	Bi-prediction divides the scene into regions, and predicts a path to all possible destinations	Predicting pedestrians' trajectories	ADE, FDE
Alajlan *et al*. (2021)	Using LSTM-Direction (cone of vision) to predict future directions for agents	Predicting agents' trajectories	ADE, FDE, CDE

Metrics abbreviations:

ADE: Average Displacement Error

FDE: Final Displacement Error

CDE : Cumulative Distance Error

AEDE: Average Eulerian angle Difference Error

ANDE: Average Non-linear Displacement Error

AEM: Average prediction Error in Meter

MAE: Mean Angular Error (in degrees)

RMSE: Root Mean Square Error

NDE: Average Nonlinear Displacement Error

SPT: Squared Path Difference

DTW: Dynamic Time Warping

(Collision rate) Col-I, Col-II: Prediction Collision, and Ground Truth Collision