DATA302-2024S: Term Project
Environment-Aware Pedestrian Trajectory Prediction Using Swin Transformer and Cross-Attention Mechanisms

Jiwon Park (2024KU0144)
Korea University College of Informatics
jjuny040627@gmail.com

Julia Hartmann (2024951578)
Korea University College of Informatics
juliahartmann1.jh@gmail.com

Jinseong Jeong (2019170609)
Korea University College of Informatics
dw9030@korea.ac.kr

Utae Jeong (2021320176)
Korea University College of Informatics
utaejeong@korea.ac.kr

Elisenda Gual (2023952462)
Korea University College of Informatics
elisendag2002@gmail.com

Abstract

This semester's project presents a novel approach to understanding pedestrian paths and predicting future behavior. Goal-SAR[2] made significant advancements in pedestrian path prediction using map images that display observed paths by sampling goal points and learning the correlation between predicted goal points and observed pedestrian paths. In this research, we aim to develop a new model based on Goal-SAR that not only better predicts pedestrian paths and goals but also makes the model environment-aware. To achieve this, we utilize the SWIN Transformer to extract features from map images, allowing the model to understand the correlation between observed paths and the surrounding map. This enables the model to gain a deeper understanding of the environment, thereby improving path learning and prediction.

1. Introduction

Autonomous driving systems, like self-driving cars, are changing the transport industry by wanting to make traveling safer and more efficient. However, the complexity of the world around us presents different challenges, one of which is understanding and predicting pedestrian trajectories.

Pedestrian trajectory is the path that a person is likely to take within a certain timeframe. By accurately predicting the path of the people around the vehicle, autonomous transit vehicles can anticipate dangerous situations and adjust their tracks accordingly, preventing accidents and ensuring the safety of both the passengers and the pedestrians. In busy environments, pedestrians can appear suddenly and move in unpredictable ways, so the correct prediction generation can be difficult to achieve since the system must be able to quickly and accurately predict it.

The goal of this project is to develop an accurate pedestrian trajectory prediction for autonomous driving systems. We aim to improve the performance of already existing predicting models by integrating environmental context through advanced feature extraction, thus enhancing the model's ability to anticipate pedestrian movement patterns in varying and complex scenarios.

2. Related works

The last few years multiple different approaches were developed to monitor people's movement and based on these observations predict the future trajectories of people. This project used some of these approaches as a basis to improve the performance of a chosen model.

2.1. Social LSTM.

One of the major improvements in human trajectory prediction is introduced by Alahi et al. [1]. Aiming at learning general movement and predicting the future trajectories of people, Alahi et al. [1] developed a new model called Social LSTM. This model is provided with an additional "Social" pooling layer that enables the hidden layers of the network to share their hidden states. Using this additional layer, the model is able to learn typical interactions between people taking place along their trajectories. Using one LSTM for each person in the observed scene the model is able to learn the state of each person and based on this information can predict the person's future position. Every person in a scene adapts its path or trajectory influenced by its immediate neighbors which leads to an alteration of this person's behavior over time. These time-varying motion properties can be captured by the LSTM's hidden states. Sharing the states between neighboring LSTMs is necessary to capture the variations and predict the future trajectories of multiple people.

The model replaces the actual coordinates with the anticipated positions to generate the new trajectory. In a group or pair of persons walking together, the model can jointly forecast their paths. Using two publicly available datasets Alahi et al. [1] showed that their model outperforms the most recent approaches used for trajectory prediction.

Challenges: Although using an LSTM combined with a social pooling layer provides a lot of improvements for trajectory prediction, there are two challenges when using an LSTM. The first one is the vanishing gradient problem making it difficult to learn dependencies in the data over a longer period. The second challenge is overfitting. When there is little or no training data available, or when the training data is noisy, LSTMs are sensitive to overfitting.

2.2. Swin Transformer.

The transformer architecture design proposed by Liu et al. [5] provides us with key innovations that can be used to process high-resolution images and extract feature maps from our datasets. To achieve this a hierarchical structure is used, starting by splitting the input image into patches that are later embedded into fixed-dimensional vectors, which are the initial tokens served to the transformer.

The model preprocesses these tokens through multiple stages with different resolution levels, where the patches are merged and its resolution is reduced. Each stage extracts features at a different scale, ending with more abstract features at higher levels. To allow the model to focus on local features in each window, the self-attention is computed when these don't overlap. The windows are shifted at each layer so they share information, generating a global relationship between distant patches.

Overall, the feature maps are generated in every stage taking that information into account, with various resolutions. Lower-stage feature maps will capture detailed local features while higher-stage feature maps will capture more abstract, global features.

2.3. AgentFormer.

As another approach to properly predict human trajectories Yuan et al. [7] introduced the AgentFormer. This model renovates the social agent-to-agent predictions through the improvement of the transformer's application in the architecture. Here, there are the main aspects that build on this model. First, it solves the problem of time information by giving a time stamp feature to each agent, a method called time encoder, which allows it to focus on the agent positions through time. Secondly, it prevents the loss of agent information by creating an agent-aware attention mechanism, that through the generation of sets of keys and queries, we can give importance to both the agent alone and the reaction between itself and other agents.

A main disadvantage that can be found in the AgentFormer is that it's highly dependent on the quality of input data, meaning that the performance will be dependent on how good the preprocessed data is.

2.4. Goal-SAR.

The paper of Goal-SAR [2] also focuses on the self-attentive RNN that predicts pedestrian trajectories using the past observed positions as we could see in the previous model AgentFormer, but without taking into consideration the humans around the predicted subject. This model also incorporates a goal-estimation module that uses semantic information to predict the future final destinations that the pedestrian may have, enhancing the prediction of the path.

It is also worth mentioning how the model preprocesses the input data with different methods. On the one hand, it uses trajectory embedding by encoding sequences of 2D locations for the self-attentive mechanism. On the other hand, in the goal-estimation module, semantic segmentation preprocessing is applied, where the network identifies several semantic classes of the environment around the agent and encodes them into a semantic tensor. This tensor is combined with the observed behavior record into the destination encoder-decoder, where the U-net-based architecture generates a spatial probability map (future locations). Those destinations are sampled from this map to use them in the trajectory prediction.

3. Method

In the realm of autonomous driving systems, we need to accurately predict pedestrian trajectories by deeply understanding how the environment and pedestrian behavior affect an agent's trajectory. Traditional models often overlook the relationship between the pedestrian and its surrounding environment. This motivates us to build a model that takes this influence into account when predicting its trajectory by using known trajectory prediction models and transformers.

3.1. Map Encoder

The input to the model consists of a semantic map image, combined with the observed paths represented as a map in the channel dimension. The semantic map image is the result of processing the original image using a U-Net network, providing detailed information about various regions and objects within the environment. The motion history is represented as a map in the channel dimension, indicating the observed paths of the agent by the number of observed coordinates. This input image is fed into a Swin Transformer-based map encoder in patches, allowing the model to learn the correlations between paths and images through a self-attention mechanism that combines the understanding of the surrounding environment with the motion history and the semantic map.

3.2. Goal Module

The Goal Module processes the concatenated motion history and semantic scene segmentation (segmented image) to predict a set of potential future goals of the agent with associated probabilities. The Goal Estimator within the module outputs the Goal Loss ($\mathcal{L}_{goal}$), which measures the accuracy of the predicted goals. These predicted goals are then sampled to generate specific goal candidates used as input for the Trajectories Transformer Encoder.

3.3. Cross-Attention Transformer

The outputs of the Trajectories Transformer Encoder and the Map Encoder are fed into a Cross-Attention Transformer. The Trajectories Transformer Encoder processes the observed paths and goal sampling results, extracting relevant features and learning temporal relationships within the input data through self-attention. The Map Encoder extracts spatial information and features about environmental elements from the semantic map. The Cross-Attention Transformer enhances prediction accuracy by learning the correlations between spatial information from the map (provided by the Map Encoder) and path information (provided by the Trajectories Transformer Encoder). This approach allows the model to effectively capture and understand the complex interactions between the environment and the paths, going beyond merely understanding and predicting future coordinates.

3.4. Transformer Decoder

The output of the Cross-Attention Transformer is processed by the Transformer Decoder to predict the agent's future trajectory. This decoder refines the combined features from previous stages to generate accurate trajectory predictions. By using the decoder, the representations learned through cross-attention are processed again through a self-attention mechanism, helping to convert them into precise future coordinates of pedestrian movements. The output of the Transformer Decoder is combined with additional features and passed through a final fully connected (FC) layer. These additional features can include latent variables sampled from a standard normal distribution ($z_i \sim \mathcal{N}(0, 1)$) to handle the uncertainty of future trajectories. Through the FC layer, these features are transformed into the predicted future coordinates.

Each component of the model plays a crucial role in accurately predicting the agent's trajectory. The use of the Swin Transformer and the Cross-Attention Transformer is particularly beneficial in learning complex spatial patterns and effectively reflecting the interactions between the observed trajectories and the environment. This comprehensive approach ensures that the model can accurately predict pedestrian movements in various environments, making it a robust solution for autonomous driving systems.

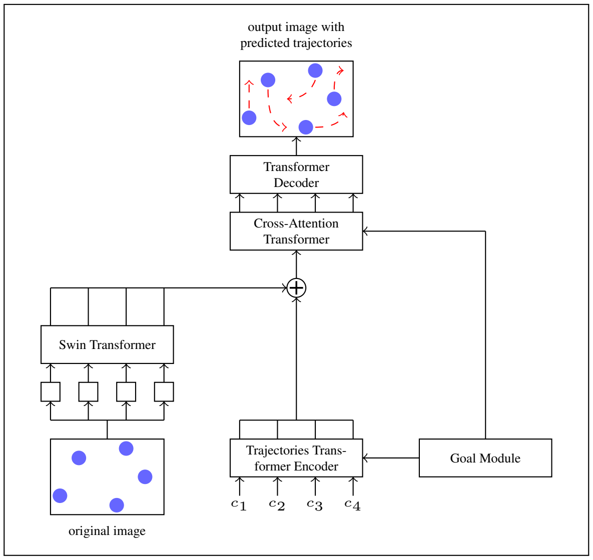
output image with predicted trajectories
Transformer Decoder
Cross-Attention Transformer
Swin Transformer
original image
Trajectories Transformer Encoder
Goal Module
c_1
c_2
c_3
c_4

Figure 1. Our final model

4. Experiments

All experiments were conducted on Google Colab. The implementation was done using Python and the PyTorch library. Both Goal_SAR and Our model were trained with a batch size of 32 for 300 epochs. For Our model, training took 15 hours on an NVIDIA A100 GPU. The Swin Transformer used was pretrained on IMAGENET1K_V1 and was fine-tuned for this task.

4.1. Datasets

To evaluate our method, we use the ETH [6] and UCY [4] datasets, which are classic well-established benchmark datasets for pedestrian trajectory prediction. These datasets provide a diverse and challenging set of scenarios that are crucial for a robust evaluation of our method.

The datasets contain five different scenes, each providing a unique set of pedestrian trajectories data with a varying interaction and environment. This diversity helps us to test the generalizability and adaptability of our model across different settings. The five scenes include ETH-Univ, ETH-Hotel, UCY-Zara01, UCY-Zara02 and UCY-Univ.

The data in each frame is structured to include both the positional information of the pedestrians and the scene context. The format of the data is as follows:

$[(Frame_{id}, Ped_{id}, X_{coor}, Y_{coor}), (map_{image})]$

Where:

- **Frame**$_{id}$: The unique identifier for each frame in the sequence.
- **Ped**$_{id}$: The unique identifier for each pedestrian in the frame.
- **X**$_{coor}$ and **Y**$_{coor}$: The X and Y coordinates representing the pedestrian's position in the scene.
- **map**$_{image}$: The image of the environment, providing the contextual map of the scene.

4.2. Training procedure

In our experiments, we trained our model using the ETH dataset and subsequently tested it on other datasets such as UCY-Zara and UCY-univ. However, the results that will be discussed after focus on the experiments where both training and testing were conducted using the ETH dataset. This consistent dataset usage allows us to observe a more controlled evaluation of our model performance.

The training process involved the following steps:

- **Data Preprocessing:** The raw trajectory data was normalized and segmented into observed and predicted segments. The observed segment consists of the initial part of the trajectory, while the predicted segment represents the future positions to be forecasted.
- **Model Training:** The model was trained using the observed trajectories and the corresponding map images. We employed the Adam optimizer with an initial learning rate of 0.001. The learning rate was decayed by a factor of 0.1 every 10 epochs.
- **Loss Function:** We used a combination of trajectory loss ($\mathcal{L}_{traj}$) and goal loss ($\mathcal{L}_{goal}$) to train the model. The trajectory loss measures the discrepancy between the predicted and actual trajectories, while the goal loss assesses the accuracy of the predicted goals.

4.3. Metrics.

The evaluation of the results uses two different metrics - the Average Displacement Error and the Final Displacement Error [3].

Average Displacement Error (ADE): ADE measures the average distance between the predicted pedestrian trajectory and the actual pedestrian trajectory using the following formula

$$ADE = \frac{1}{T}\sum_{t=1}^{T}\sqrt{(x_t - \hat{x}_t)^2 + (y_t - \hat{y}_t)^2} \qquad (1)$$

Therefore, ADE represents the average error for the 8 predicted paths.

Final Displacement Error (FDE): FDE measures the distance between the final point of the predicted trajectory and the final point of the actual trajectory based on the following formula

$$FDE = \sqrt{(x_T - \hat{x}_T)^2 + (y_T - \hat{y}_T)^2} \qquad (2)$$

Within this project, FDE represents the error for the last predicted coordinate.

4.4. Preliminary Study.

Aiming at verifying whether using images can improve model performance a preliminary study is conducted. In the preliminary study, we conducted experiments to validate the basic functionality of our model and to perform hyperparameter tuning. We experimented with different configurations of the learning rate, batch size, and number of layers in the neural network. Grid search was used to find the optimal parameters that minimized the validation error.

To evaluate the generalizability of the model the preliminary experiments were conducted using the initial model without the Swin Transformer:

- **Goal-SAR [2]:** The full model with goal estimation and all the different modules explained previously.
- **SAR [2]:** Standard model that does not apply goal estimation, only takes into account the historical trajectory of the pedestrian for the prediction.

The preliminary study shows that the base model experiences overfitting, while the model that used Goal sampling from images showed a lower validation loss and more stable learning. This highlighted the need for a new approach that integrates high-dimensional image features directly into the model.

The results of these tests are summarized in table 1.

Model	**Goal_SAR**		**SAR**	
	ADE	**FDE**	**ADE**	**FDE**
eth	0.3448	0.5913	0.3711	0.6852
hotel	0.1407	0.2076	0.2413	0.4500
zara1	0.1782	0.2627	0.4103	0.8160
zara2	0.1659	0.2606	0.3333	0.6666
univ	0.2243	0.3425	0.4908	0.9754

Table 1. Performance of Goal_SAR and SAR models on various datasets (world metrics).

4.5. Result.

Model	Our model		Goal_SAR	
	ADE	FDE	ADE	FDE
eth	0.38394	0.59277	**0.34480**	**0.59134**
hotel	**0.13330**	**0.19624**	0.14072	0.20764
zara1	**0.17013**	**0.25000**	0.17816	0.26270
zara2	**0.15906**	**0.24813**	0.16590	0.26064
univ	0.22478	**0.33429**	**0.22428**	0.34245

Table 2. Performance comparison of Our model and Goal_SAR on various datasets. Better performance values are highlighted.

Table 2 presents the performance results of the model on these test datasets.

In terms of ADE, for the eth dataset, Goal_SAR (0.34480) performed better than SAR (0.37111) and Our Model (0.38394). For the hotel dataset, Our Model (0.13330) outperformed Goal_SAR (0.14072) and SAR (0.24132). On the zara1 dataset, Our Model (0.17013) had the lowest ADE compared to Goal_SAR (0.17816) and SAR (0.41031). Similarly, for the zara2 dataset, Our Model (0.15906) outperformed Goal_SAR (0.16590) and SAR (0.33334). In the univ dataset, Goal_SAR (0.22428) had a slightly lower ADE than Our Model (0.22478) and significantly outperformed SAR (0.49077).

Regarding FDE, on the eth dataset, Goal_SAR (0.59134) was better than Our Model (0.59277) and SAR (0.68516). For hotel, Our Model (0.19624) had a lower FDE than Goal_SAR (0.20764) and SAR (0.45004). On the zara1 dataset, Our Model (0.25000) outperformed Goal_SAR (0.26270) and SAR (0.81601). Similarly, for zara2, Our Model (0.24813) was better than Goal_SAR (0.26064) and SAR (0.66661). Finally, for the univ dataset, Our Model (0.33429) outperformed Goal_SAR (0.34245) and SAR (0.97535).

Goal_SAR generally outperforms SAR and sometimes Our Model in both ADE and FDE, particularly in the eth and univ datasets. Our Model shows the best performance, with the lowest ADE and FDE in most datasets, including hotel, zara1, and zara2. SAR consistently performs the worst in both ADE and FDE across all datasets.

5. Analysis

5.1. Ablation Study - Without Cross Attention

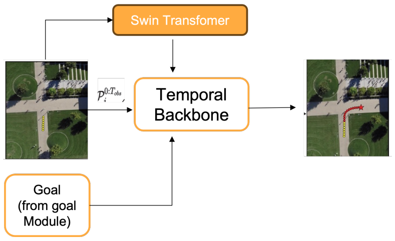

Figure 2. Integrating Swin Transformer base Map Encoder features with the Temporal Backbone without cross attention.

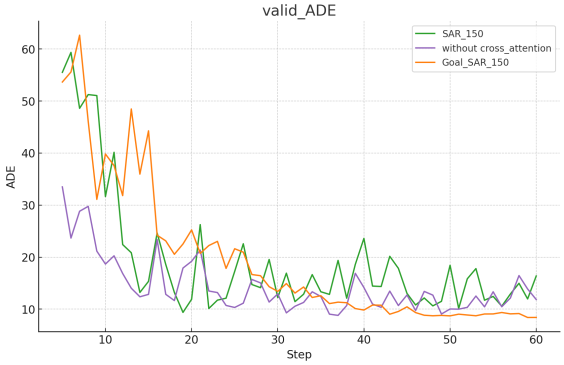

Figure 3. The above picture is a graph showing the valid ADE between SAR, Goal SAR, and a model with cross attention removed from our final model.

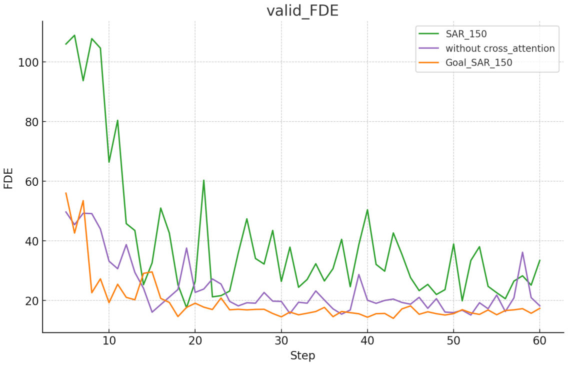

Figure 4. The above picture is a graph showing the valid FDE between SAR, Goal SAR, and our final model with cross attention removed.

Looking at the above graph, when the Swin Transformer base Map Encoder features are simply passed as input to the Trajectory Decoder (Temporal Backbone) along with the trajectory and sampled goal without applying cross attention, we can observe significant variations in validation and a learning pattern similar to SAR. Comparing this to Goal_SAR, which reduced validation variations and solved overfitting by using goal sampling, it indicates that despite using goals, the model failed to effectively learn the high-dimensional correlations between goals, trajectories, and map features. Although more information and features were provided compared to Goal_SAR, the model's performance declined. This suggests that simply adding Swin features to the Temporal Backbone and using self-attention does not necessarily enhance performance. In fact, it may introduce additional complexity that hinders the model's ability to accurately predict trajectories and means map features act more like noise. Therefore, these results reaffirm the necessity of using a cross-attention mechanism, which can more effectively manage the integration of high-dimensional features and trajectory information.

5.2. Significance of Swin Transformer

The addition of the Swin Transformer to our model aimed to enhance the accuracy of path prediction by efficiently processing input map images. Here's a detailed explanation of the improvements:

- **Image and Trajectory Processing:** The Swin Transformer applies a sliding window approach over multiple patches, allowing the model to learn the relationship between the observed paths and the map within the map image where the trajectory is added to the channel dimension.
- **Feature Extraction:** The Swin Transformer is incorporated to extract high-quality features from the combined map and trajectory images. By doing so, the model can better understand the intricate details and correlations between the pedestrian paths and the surrounding environment. This makes the Swin Transformer an excellent choice for the map encoder, as it excels at extracting relevant features from images where the trajectory and map information are combined in the channel dimension.
- **Pre-trained Parameters:** The Swin Transformer leverages over 100 million pre-trained parameters, which helps the model find good features without extensive backpropagation. This pre-training advantage allows our model to achieve high performance in significantly fewer epochs compared to Goal-SAR. The ability to use these high-quality pre-trained parameters means that the initial performance of the model is significantly boosted without requiring a long training time. This results in improved accuracy and efficiency, making the model highly effective even with a limited number of training epochs.

In our model, the Swin Transformer plays a crucial role in the cross-attention mechanism between the map and the trajectory. By effectively encoding the map and the observed trajectories, it allows the model to focus on the most relevant features and thus significantly improves the prediction accuracy. The Swin Transformer's ability to capture fine-grained details and contextual information from the map images enhances the model's overall performance in predicting pedestrian paths and goals.

5.3. Goal Sampling

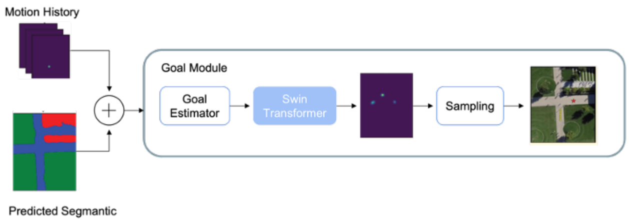

Figure 5. Integrating the Swin Transformer with the Goal Module did not significantly improve performance, indicating limitations of goal features alone and prompting exploration of higher-dimensional image features.

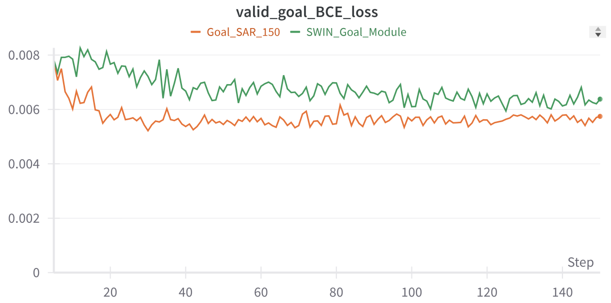

Figure 6. The picture above is a graph showing the loss value for the Goal Map used to predict the destination between the two models.

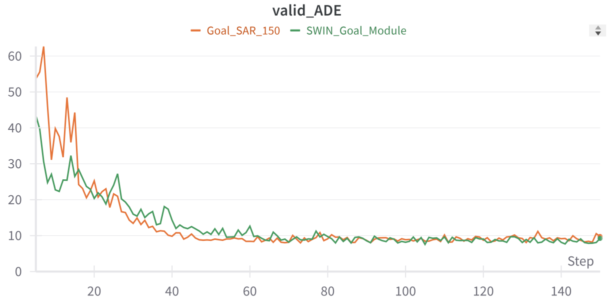

Figure 7. The picture above shows Valid ADE between the two models. Training was conducted up to 150 epochs, and you can see that the performance difference between the two models is almost similar.

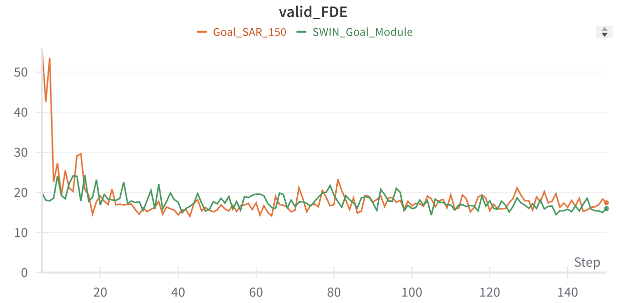

Figure 8. The picture above shows Valid FDE between the two models. Training was conducted up to 150 epochs, and you can see that the performance difference between the two models is almost similar.

To improve goal sampling, we experimented by adding *Swin Transformer* to the *Goal Module*. However, the results did not show significant improvement in goal loss performance. This suggests that there is a limit to improving performance by only extracting goal features very accurately. Therefore, we hypothesized that better features might exist that could further enhance the model's performance when combined with goal features, and we considered extracting features that capture the correlation between the surrounding environment and the trajectory through *Swin Transformer*.

In Goal_SAR, the goal is very important and provides meaningful features, but our research findings indicate that as long as the sampled goal is close to the actual path within a certain boundary, the actual point prediction performance is not significantly affected. Thus, instead of focusing too much on perfect goal sampling, it is crucial to find other valuable features that can enhance model performance.

We aimed to improve the model's performance by supplementing the goal with additional high-quality features derived from the relationship between the observed trajectories and the surrounding environment. By using the *Swin Transformer* to extract these features, we sought to enrich the model's understanding of the environment and provide features that enhance the model's performance in conjunction with goal features.

5.4. Ablation Study - Without Goal Sampling

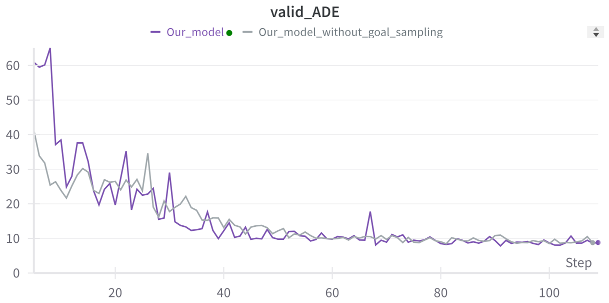

Figure 9. The picture above shows the valid ADE values between our final model and the model with the goal sampling function removed.

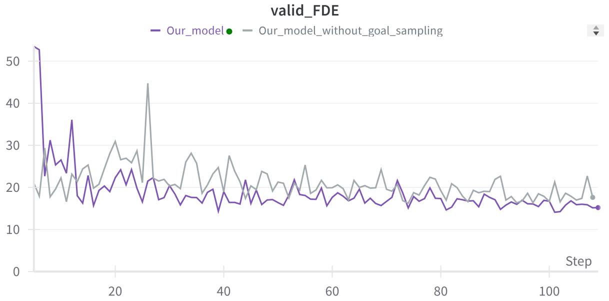

Figure 10. The picture above shows the valid FDE values between our final model and the model with the goal sampling function removed.

When applying without goal sampling, the ADE remains similar, but the variance in FDE validation increases significantly. This indicates that while the cross attention between the map and the trajectory allows for good average path prediction without goal sampling, there is a discrepancy in predicting the final destination point. This means that the goal feature is very intuitive and effective for FDE, and the goal_sampling module is more effective. Cross attention, when used with the goal, learns both the long-term goal and the correlation between the map and the trajectory, producing good results in both ADE and FDE. While cross attention alone is effective enough to reduce ADE, it has limitations in predicting lower FDE, indicating that goals are still necessary. The role of cross attention is to find features that synergize with the goal.

6. Conclusion

We explored various models for predicting pedestrian trajectories in autonomous driving systems. Our model integrates map encoding using Swin Transformer and effectively combines these features with trajectory information through a cross-attention mechanism. While existing models often overlook the complex interactions between pedestrians and their environment, our approach learns the correlations between the surrounding environment and the paths.

Through experiments, we confirmed the effectiveness of directly integrating map encoding using Swin Transformer into the model. Various ablation studies and additional experiments highlighted the importance of the map encoder, goal, and path, reaffirming the need for a cross-attention mechanism to effectively integrate spatial and trajectory information.

Our model significantly outperforms the baseline SAR, showing results similar to Goal-SAR but with slightly higher accuracy, especially in complex environments like 'hotel', 'zara1', and 'zara2'. These results emphasize the importance of considering environmental context in trajectory prediction tasks and the potential of advanced Transformer architectures.

7. Limitation

While our model demonstrates improved performance, it still faces several limitations. First, the computational complexity of the Swin Transformer and cross-attention mechanisms results in increased training and inference times. Second, the model's performance can be sensitive to the quality of the map data and the resolution of the input images. Lastly, the model may struggle in scenarios with highly dynamic environments or in cases where pedestrian behavior is influenced by factors not captured in the map data, such as social interactions or weather conditions. Future work should address these limitations by optimizing the model's efficiency, exploring data augmentation techniques, and incorporating additional contextual information.

8. Future Work

Our approach has shown promising results, but there are several potential areas for improvement and future research. First, we propose exploring methods to improve FDE without using a goal module by finding features that are better for FDE than goals. This includes attempting a fully transformer-based structure by identifying features as efficient as or more efficient than goals. This research aims to enhance performance through better features.

Secondly, one of the main limitations we faced was the limited scope of training due to constraints on computational resources. Using more powerful hardware and extending training time can significantly optimize the model's performance. Here, we can consider further training the pretrained Swin Transformer directly. Additionally, using advanced GPU or TPU resources allows for larger batch sizes and more complex model architectures, resulting in improved accuracy and robustness.

Thirdly, future research requires a more comprehensive exploration of hyperparameters. The first priority is to optimize the number of transformers and the embedding size. This includes experimenting with different learning rates, batch sizes, dropout rates, and other model-specific parameters. Automatic hyperparameter tuning methods such as Bayesian optimization or grid search can systematically find the optimal settings.

Finally, it is important to conduct experiments under various conditions to test the robustness of the model. These include tests for different environmental conditions, pedestrian interactions, and geographic locations. These experiments will provide deeper insights into the generalizability of the model and identify specific areas that require further improvement.

Addressing these areas will further improve the effectiveness and applicability of pedestrian trajectory prediction models, ultimately contributing to safer and more efficient autonomous driving systems.

References

[1] Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social lstm: Human trajectory prediction in crowded spaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 961–971, 2016. 1, 2

[2] Luigi Filippo Chiara, Pasquale Coscia, Sourav Das, Simone Calderara, Rita Cucchiara, and Lamberto Ballan. Goal-driven self-attentive recurrent networks for trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2518–2527, 2022. 2, 4

[3] Jaime B. Fernández R. Error metrics for trajectory prediction accuracy. https://jaimefernandezdcu.wordpress.com/2019/02/07/error-metrics-for-trajectory-prediction-accuracy/, 2019. [Accessed: 2024-06-20]. 4

[4] Ariel Lerner, Yiorgos Chrysanthou, and Dani Lischinski. Crowds by example. In *Computer Graphics Forum*, pages 655–664. Wiley Online Library, 2007. 3

[5] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows, 2021. 2

[6] Stefano Pellegrini, Andreas Ess, Konrad Schindler, and Luc Van Gool. 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, 2009. 3

[7] Ye Yuan, Xinshuo Weng, Yanglan Ou, and Kris M Kitani. Agentformer: Agent-aware transformers for socio-temporal multi-agent forecasting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9813–9823, 2021. 2