Spatial-Temporal-Spectral LSTM: A Transferable Model for Pedestrian Trajectory Prediction

Chi Zhang, Zhongjun Ni, and Christian Berger

***Abstract*—Predicting the trajectories of pedestrians is critical for developing safe advanced driver assistance systems and autonomous driving systems. Most existing models for pedestrian trajectory prediction focused on a single dataset without considering the transferability to other previously unseen datasets. This leads to poor performance on new unseen datasets and hinders leveraging off-the-shelf labeled datasets and models. In this paper, we propose a transferable model, namely the "Spatial-Temporal-Spectral (STS) LSTM" model, that represents the motion pattern of pedestrians with spatial, temporal, and spectral domain information. Quantitative results and visualizations indicate that our proposed spatial-temporal-spectral representation enables the model to learn generic motion patterns and improves the performance on both source and target datasets. We reveal the transferability of three commonly used network structures, including long short-term memory networks (LSTMs), convolutional neural networks (CNNs), and Transformers, and employ the LSTM structure with negative log-likelihood loss in our model since it has the best transferability. The proposed STS LSTM model demonstrates good prediction accuracy when transferring to target datasets without any prior knowledge, and has a faster inference speed compared to the state-of-the-art models. Our work addresses the gap in learning knowledge from source datasets and transferring it to target datasets in the field of pedestrian trajectory prediction, and enables the reuse of publicly available off-the-shelf datasets.**

***Index Terms*—Pedestrian trajectory prediction, deep learning, transferable models, spectral representation, Fourier transform**

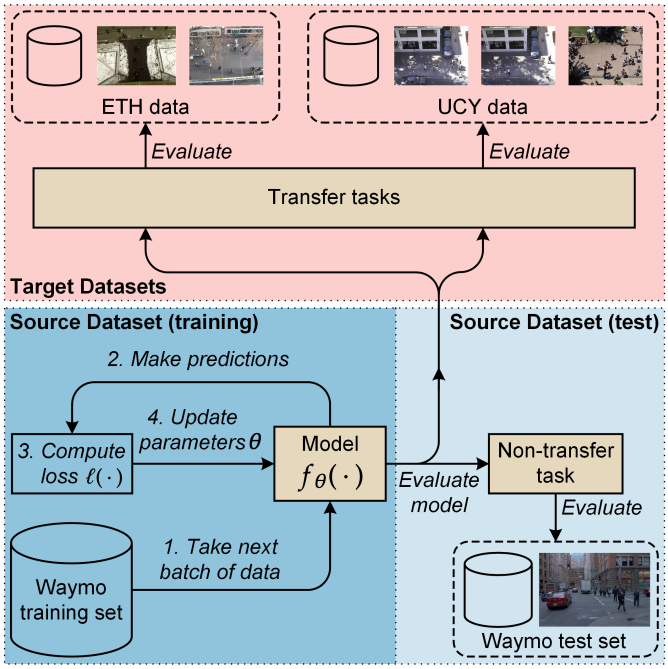


Fig. 1. The concept of a transferable model. The model is trained on the source dataset, Waymo training set (shown in the dark blue block). The transferable model can perform well on both the non-transfer task on the source dataset, Waymo test set (shown in the light blue block) and transfer tasks on target datasets, ETH and UCY datasets (shown in the pink block).

I. Introduction

THE demand for road safety stimulates the rapid development of driver assistance systems and automated driving systems that require vehicles to understand the behavior of other road users. Pedestrians are the most vulnerable among all road users, accounting for 23% of all road deaths globally, according to the World Health Organization (WHO)'s safety report [1]. Therefore, accurately predicting pedestrian behavior in complex traffic scenarios is pivotal for developing automated vehicles. Many research studies have focused on pedestrian intention prediction (e.g. [2]–[4]) and trajectory prediction (e.g., [5]–[7]) to avoid potential pedestrian-vehicle conflicts and ensure driving safety.

Deep learning-based models have shown their strong potential for pedestrian trajectory prediction as stated in [8]. Long short-term memory (LSTM)-based models are capable to deal with time series, as used in Social LSTM proposed by Alahi et al. [5] and its extensions [9]–[11]. Gupta et al. [6] proposed Social GAN that used the LSTM-based encoder-decoder as the generator in generative adversarial networks (GANs). Convolutional neural networks (CNNs) such as temporal convolutional networks (TCNs) can also be used for trajectory prediction, as used by Mohamed et al. [7] and Zhang et al. [12], [13]. Recently, Transformers [14] have been utilized in pedestrian trajectory prediction, such as the methods proposed by Giuliari et al. [15] and Yu et al. [16].

However, most existing deep learning methods focus on single tasks and are trained on single datasets [17]. These models assume that the training and test data follow the same distribution, and usually get poor predictive accuracy on new unseen and untrained datasets of different scenarios. Focusing on single tasks without considering transferability and generalizability hinders us from utilizing off-the-shelf labeled datasets and models. Therefore, there is a great need for developing transferable models that can learn the knowledge

Chi Zhang and Christian Berger are with the Department of Computer Science and Engineering, University of Gothenburg, Gothenburg, Sweden (email: chi.zhang@gu.se, christian.berger@gu.se).

Zhongjun Ni is with the Department of Science and Technology, Linköping University, Campus Norrköping, Norrköping, Sweden (email: zhongjun.ni@liu.se).

Corresponding author: Chi Zhang.

TABLE I
STATISTICS OF THE SOURCE DATASET (WAYMO OPEN DATASET) AND TARGET DATASETS (ETH AND UCY DATASETS).

Dataset name	Number of Frames (@2.5Hz)	Number of pedestrian sequences	Average number of targets per frame	Average speed (m/s)
Waymo (train)	17,127	8,328	29.33	0.91
Waymo (test)	3,570	1,978	30.91	0.95
ETH-univ	1,448	603	6.27	0.92
ETH-hotel	1,168	301	5.60	1.04
UCY-zara1	872	602	5.91	1.07
UCY-zara2	1,052	921	9.24	0.79
UCY-univ	985	947	40.37	0.63

from source datasets and apply it to different target datasets without a sharp decline in performance.

The transferability of a model refers to its ability to transfer knowledge learned from one or more source tasks and then reuse it in new related target tasks [17]. As illustrated in Fig. 1, a transferable model is trained on the source data and is expected to have the capability to handle unseen and untrained cases in target datasets. The statistics of the source and target datasets are shown in Table I. Different datasets vary in the number of frames, the number of pedestrian sequences, the density of crowds or the average number of pedestrians in each frame, and also the average walking speed of pedestrians.

Instead of designing a complex model with complicated structures, our goal is to propose a method with simple structures that can be easily generalized to other datasets and can be combined with other methods. We propose a novel data representation method, and evaluate existing prediction structures and loss functions to find the best combination. We aim to provide guidance to other researchers who want to design transferable models. In particular, we are looking into the following research questions (RQs):

RQ1 How does our proposed Spatial-Temporal-Spectral (STS) LSTM model perform on non-transfer and transfer tasks compared to existing state-of-the-art (SOTA) pedestrian trajectory prediction models?

RQ2 Which commonly used structure among LSTM-based, CNN-based, and Transformer-based models is the most transferable that can learn a generic motion pattern regardless of the dataset?

RQ3 What performance improvement can be achieved using our proposed spatial-temporal-spectral representation compared to regular time series representation?

To achieve our research goal and answer the research questions, we propose the STS LSTM that can learn general pedestrian motion features and perform well on both non-transfer and transfer tasks. The main contributions of this paper are as follows:

- We propose a novel transferable pedestrian trajectory prediction model, namely the STS LSTM that performs well on both source and target datasets, with a faster inference speed compared with the SOTA methods.
- We propose a method to represent input features of pedestrian trajectories in spatial, temporal, and spectral domains that can better represent pedestrian motion patterns to enable model transferability.
- We reveal the transferability of three commonly used network structures, including LSTMs, CNNs, and Transformers, and compare the performance of two commonly used loss functions, including L2 loss and negative log-likelihood (NLL) loss for pedestrian trajectory prediction.
- We quantitatively analyze how pedestrian trajectory prediction performance decreases when the model transfers from the source task to target tasks. We visualize and qualitatively analyze the input and output of the model.

The structure of the following sections is: Sec. II presents related work. Sec. III describes the proposed transferable model. Sec. IV presents the experiment details. The results and analysis are provided in Sec. V. Conclusions and future works are described in Sec. VI.

II. RELATED WORK

A. Deep Learning-based Pedestrian Trajectory Prediction

Pedestrian trajectory prediction models aim to predict the future positions of pedestrians in temporal order based on their past positions. In recent years, deep learning-based methods have made great progress. In this section, we introduce three commonly used deep learning structures for trajectory prediction.

1) LSTM-based models: Long short-term memory (LSTM) networks are an improved version of recurrent neural networks (RNNs). LSTMs have a strong ability to handle long sequences and have been introduced to trajectory prediction by Alahi et al. [5]. The authors proposed Social LSTM, using the social pooling layer over LSTMs to learn social interactions between pedestrians. Many researchers followed this trend of using LSTM-based prediction, and extended Social LSTM by improving the interaction module (e.g., [9]–[11]). In addition to spatial interactions, Wu et al. [18] considered temporal interactions and proposed a hierarchical spatio-temporal attention model that jointly considers both spatial and temporal interactions across time steps of all agents. Gupta et al. [6] argued that there could be multiple plausible trajectories given the past trajectories, and proposed Social GAN based on a multi-modal distribution assumption. They applied generative adversarial networks (GANs) with LSTM-based generators. Some studies followed the multi-modal distribution assumption and simultaneously generated several possible future trajectories (e.g., [19]–[21]).

2) CNN models: Convolutional neural networks (CNNs) have gained great success in the computer vision field. Bai et al. [22] pointed out that CNNs can also be used for sequence modeling to replace RNNs because they have higher efficiency. Nikhil and Morris [23] employed CNNs for trajectory prediction and achieved competitive results with computational efficiency. In contrast to RNNs whose later time step prediction depends on previously predicted time steps, CNNs such as Temporal Convolutional Networks (TCNs) predict all time steps simultaneously. This can alleviate accumulated errors, and enable parallelization. Mohamed et al. [7] and Zhang et al. [12], [13] used CNNs and TCNs for prediction and considered social

interactions. Bae and Jeon [24] followed this direction, and improved the social interaction module using a disentangled multi-relational graph, considering multi-scale aggregation and temporal aggregation, and achieved better performance.

3) Transformer-based models: Recently, Transformers [14] are becoming popular as they can better memorize the information in long sequences compared to RNNs. The attention mechanism of Transformers can create shortcuts between the context vector and the entire input instead of only the last hidden state. Transformers gain better performance compared to RNNs, and allow parallelization to save training time. In recent years, Transformers have achieved ground-breaking progress in Natural Language Processing (NLP) field, and are adopted for predicting pedestrian trajectories. Giuliari et al. [15] used Transformers on individual trajectories as input and achieved better performance than previous LSTM- and CNN-based models. Other studies [16], [25], [26] using Transformer also considered the interaction with other road users and context information.

In this paper, we apply all three prediction structures to find the most transferable structure for trajectory prediction. As presented and discussed in Sec. V-B, LSTMs have better transferability and are employed in our proposed model.

B. Input and Output Representations

There are many methods to represent input and output data, and accordingly, different loss functions are used for prediction. In this paper, we aim to develop a transferable model that can be applied to new target datasets. Since the sensors for collecting data vary in different datasets, to avoid the influence of sensors, we only consider using pedestrian trajectories as model input.

Pedestrian trajectories can be represented as discrete variables by grid-based representations. The frame scene can be discretized into grids to encode the location information, as in studies [27]–[30]. Besides, the input trajectory data can also be quantized into discrete velocity bins and represented by the one-hot encoding. Thus, the prediction can be treated as a classification problem as proposed in Giuliari et al.'s study [15]. Although the discrete representation of pedestrian trajectories enables a parameter-free approximation of distributions, it requires high dimensionality, and the quantization errors may cause poor prediction results.

Instead of representing pedestrian positions as discrete variables, most existing methods used observed spatial positions of pedestrians in (x, y) coordinates as continuous values to represent the input trajectories, and directly regress the output trajectories. The output of the trajectory prediction model is mainly represented in several ways. Using deterministic positions of (x, y) coordinates is the simplest way to represent the output, with mean square error (MSE) or L2 loss as the loss function, as in studies [9], [15], [23], [31]. Another way to represent the trajectory output is using uni-modal distributions with parameters with negative log-likelihood as loss function, as in studies [5], [7], [11], [12], [32]–[35]. Multi-modal distributions that consider multiple plausible trajectories are also considered by researchers, developing generative models such as GANs. The adversarial loss is used together with L2 loss in such models, as in [6], [19], [21], [36], [37].

In addition to treating trajectory prediction as time series generation, a recent study by Wong et al. [38] investigates the trajectories from the spectral domain. They represented the input and output by the Fourier spectrum. After predicting the output spectrum, an inverse Discrete Fourier Transform (DFT) is performed to obtain the output time series. The model is trained with point-wise L2 loss over output time series.

In this paper, we combine the representation of pedestrian trajectories in spatial, temporal, and spectral domains as input features. Both the positions in temporal order and the Fourier spectrum of pedestrian trajectories are used to represent pedestrian motion patterns. Regarding the output representation, we compare uni-modal distribution prediction with NLL loss and deterministic prediction with L2 loss. As discussed and analyzed in Sec. V-B, the uni-modal distribution prediction with NLL loss demonstrates better performance, so it is employed by our model.

C. Model Transferability

Human drivers can inherently transfer knowledge between similar driving scenarios. They can recognize and apply relevant knowledge from previous driving experiences when encountering new scenarios. When developing learning-based algorithms, since the data collection and annotation of pedestrians in traffic scenarios is time-consuming and expensive, we may face the situation that we cannot obtain any prior information about a new environment, and need the model to have transferability and can be used on untrained data. Some studies on unsupervised learning tried to develop structural developmental neural networks to simulate the growth and development of the human brain. Ding et al. [39] proposed a structural developmental neural network using competitive learning rules and dynamic neurons with information saturation to improve model performance when sufficient training samples and prior knowledge of the task are not available.

When it comes to supervised learning, classic supervised deep learning methods address isolated tasks [17], i.e., a predictive model for a specific task is trained on a single dataset and only solves the prediction on that particular dataset. The ability to transfer the learned knowledge between related but different scenarios is called transferability. *Transfer learning* [17] attempts to enable better transferability by developing models that can transfer knowledge learned from one or several source tasks to apply to new related target tasks. Transfer learning has been successfully used in the computer vision (CV) field (e.g., [40]–[42]) and natural language processing (NLP) field (e.g., [43], [44]). With good transferability, deep learning models are rarely trained from scratch, and off-the-shelf models and datasets are widely used which can save annotation and computation resources.

Inspired by the benefits of model transferability, researchers in the field of pedestrian behavior prediction recently attempted to develop transferable models. Shen et al. [45] and Jaipuria et al. [46] tried to develop transferable prediction models that are trained on pedestrian trajectories collected at one intersection and can be generalized to other previously unseen intersections. They focused on learning features that represent

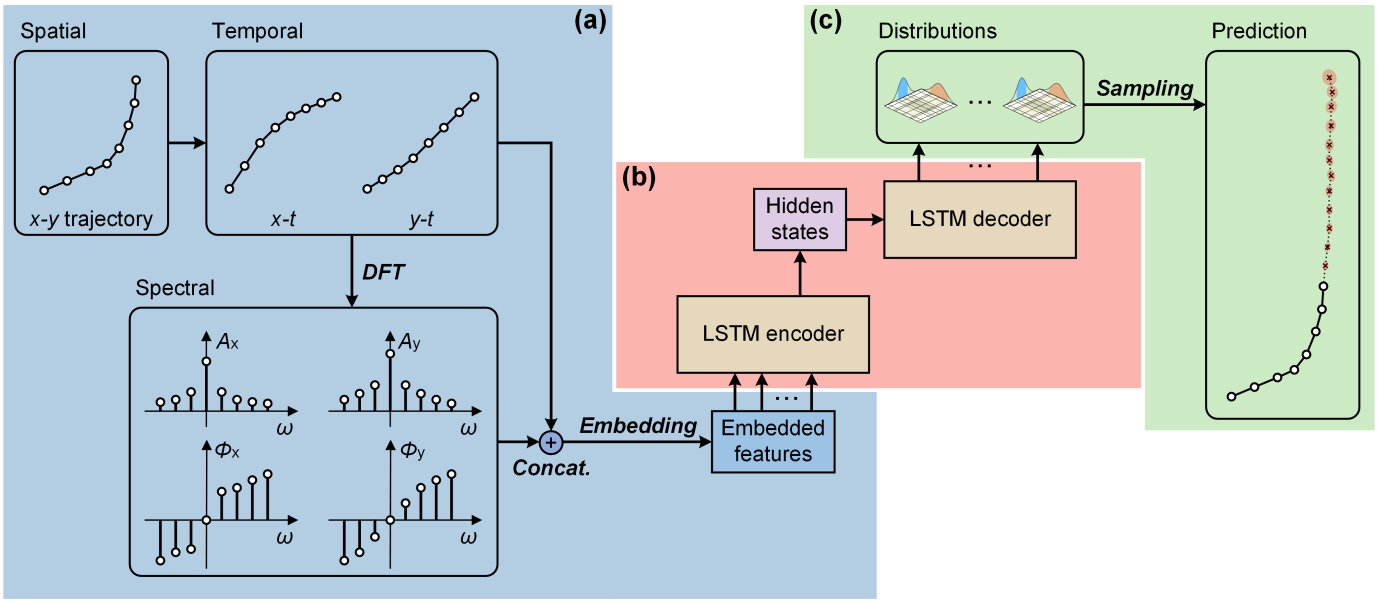


Fig. 2. The overall framework of the Spatial-Temporal-Spectral (STS) LSTM model. The model contains three components: a) the spatial-temporal-spectral feature encoding module for feature extraction (shown in the blue block), b) the LSTM encoder-decoder module for predicting the distribution (shown in the red block), and c) the sampling module to get the final predicted trajectories from the predicted distributions (shown in the green block).

the relationship between pedestrian behavior and intersection geometry, which can be generalized to new intersections. Zhang et al. [47] developed a heterogeneous agent trajectory prediction model, and transferred the learned knowledge from vehicles and pedestrians to the prediction of rarely-seen cyclists to improve the prediction accuracy. Xu et al. [48] developed an adaptive trajectory prediction model. They proposed an attention-based adaptive knowledge learning module to learn the domain-invariant individual-level transferable features. But this model requires access to a validation set of target data.

In this paper, we are interested in transferring learned knowledge from one source dataset to other previously unseen target datasets. We propose a transferable model that learns the general motion pattern of pedestrians instead of focusing on trajectories in a specific scenario alone. Our proposed model represents the input features in spatial, temporal, and spectral domains, that can well reflect the general motion patterns of pedestrians, and utilizes well-generalized prediction structures. The proposed model can achieve high accuracy in both non-transfer and transfer tasks.

III. Methodology

The overall framework of the proposed STS LSTM model is shown in Fig. 2. There are three components of the model, a) the spatial-temporal-spectral feature encoding module for feature extraction, b) the LSTM encoder-decoder module for predicting the distributions, and c) the sampling module to get the final predicted trajectories from the predicted distributions.

A. Problem definition

1) Pedestrian trajectory prediction: Given the observed trajectories of pedestrians in the past, we aim to predict the most likely trajectories of pedestrians in the future. The position of a pedestrian in a scene is represented as $p = (x, y)$ in the x-y-coordinate. In a recorded sequence with n pedestrians, the i^{th} observed trajectories of pedestrians are denoted as $X^i = [p_1, p_2, \ldots, p_{T_{obs}}]$, where $i \in \{1, \ldots, n\}$, and T_{obs} is the observed time steps. The predicted trajectories of pedestrians are denoted as $\hat{Y}_i = [\hat{p}_{T_{obs}+1}, \ldots, \hat{p}_{T_{pred}}]$ in future time steps. The ground truth of the i^{th} pedestrian's future trajectory is $Y_i = [p_{T_{obs}+1}, \ldots, p_{T_{pred}}]$, where $i \in \{1, ..., n\}$.

2) Transferability: In addition to conventional pedestrian trajectory prediction, we particularly investigate the transferability of the prediction models. Both non-transfer and transfer tasks are evaluated. For the non-transfer task, the model is evaluated on the source dataset, i.e., the test set is collected in the same scenarios as the training set. For transfer tasks, the model is evaluated on target datasets, i.e., the tests are collected in different scenarios from the training set. No prior information or access to the target datasets is provided. There are no overlaps between training and test datasets for both non-transfer and transfer tasks.

For better transferability to different datasets, we need to avoid the impact of different sensor types (e.g., camera vs. LiDAR), different image resolutions, and different calibration parameters. Therefore, we do not use other input information such as camera images and maps. The model only takes the past trajectories of pedestrians in 2D real-world (x, y) coordinates from the bird's-eye-view as input.

B. Spatial-Temporal-Spectral Feature Representation

With pedestrian trajectories as input, most existing methods use only pedestrians' positions in time series to represent input features. This works well on a single task when the training and test data are collected in the same place, where most pedestrians follow the same motion pattern, traffic rules, and

culture. When it comes to a new dataset collected in a different place, the motion pattern represented by pedestrian position information may become different.

To find a more general way to represent pedestrian motion, we consider spectral representation. The spectrum of pedestrian trajectories decomposes the time series into a combination of different frequencies that can reflect their motion patterns at different frequency scales [38]. Therefore, we propose the spatial-temporal-spectral feature encoding module to reflect the intrinsic motion patterns of pedestrians.

The observed trajectory sequence of the i^{th} pedestrian is denoted as $X^i = [p_1, p_2, \ldots, p_{T_{obs}}]$, and contains spatial information of positions $p = (x, y)$. The temporal information for each dimension can be represented as in Eq. 1.

$$P_x = [x_1, x_2, \ldots, x_{T_{obs}}] \quad (1)$$
$$P_y = [y_1, y_2, \ldots, y_{T_{obs}}]$$

where the sampling time step of the sequence is $t = [1, 2, \ldots, T_{obs}]$. For the time series on each dimension, we apply Discrete Fourier Transform (DFT) to get T_{obs} point spectral information of the trajectory, and obtain the amplitudes and phases of each dimension, as shown in Eq. 2.

$$S_x = (A_x, \Phi_x) = DFT([x_1, x_2, \ldots, x_{T_{obs}}]) \quad (2)$$
$$S_y = (A_y, \Phi_y) = DFT([y_1, y_2, \ldots, y_{T_{obs}}])$$

where the frequency range is $\omega = [\omega_0, \omega_1, \ldots, \omega_{N-1}]$. The k^{th} frequency component is calculated by Eq. 3.

$$\omega_k = k * f_s / N \quad (3)$$

where f_s is the frequency of the input time series signal, N is the number of sample points, that equals to T_{obs}.

The spatial, temporal, and spectral features are concatenated and fed into the embedding layer to obtain the encoded spatial-temporal-spectral feature, as shown in Eq. 4.

$$e = W_{em} \cdot concat(P_x, P_y, S_x, S_y) \quad (4)$$

where W_{em} is the linear embedding weights, $concat(\cdot)$ denotes the concatenate operation. For each i^{th} pedestrian, we get the embedded feature e^i.

C. LSTM Encoder-Decoder Prediction Structure

As we have introduced in Sec. II-A, there are three commonly used structures for pedestrian trajectory prediction, LSTMs, CNNs, and Transformers. These models learn different motion features of pedestrians and perform differently. We use the sequence-to-sequence LSTM encoder-decoder structure in our model to predict pedestrian trajectories and compare its transferability to the other two models.

After obtaining the i^{th} pedestrian's spatial-temporal-spectral embedded feature e^i, we feed it into the LSTM encoder to learn the hidden states of pedestrians, as shown in Eq. 5.

$$h^i_{T_{obs}} = LSTM_{enc}(h^i_0, e^i; W_{enc}) \quad (5)$$

where $h^i_{T_{obs}}$ is the encoded feature of LSTM encoder for the i^{th} pedestrian, h^i_0 is the initial hidden state, e^i is the embedded spatial-temporal-spectral feature. The LSTM encoder is denoted as $LSTM_{enc}(\cdot)$, and the weights learned by the LSTM encoder are W_{enc}.

Traditional LSTM networks that take time series as inputs can store and retrieve information over long time intervals using the input gate, output gate, and forget gate, and store the information in the memory cell. The LSTM networks use hidden states to learn and represent the "motion status" of each pedestrian over time.

In our proposed model, LSTMs are fed with spatial information in temporal order and spectral information in frequential order. The network learns and stores not only important temporal information, but also critical frequency components. Both temporal motion status and spectral moving preferences are learned in this way. The LSTM encoder encodes the input sequence into a fixed-length vector as the latent representation of the time and frequency information. The learned weights are shared across all pedestrian sequences.

Then, we use the LSTM decoder to generate the output sequence. The LSTM decoder takes the predicted position of the previous time step as input. Here we use the mean value of predicted distribution $(\mu_x, \mu_y)_{t-1}$ and hidden states h_{t-1} to generate the LSTM decoder output h_t for time step t, as shown in Eq. 6.

$$h^i_t = LSTM_{dec}(h^i_{t-1}, l^i_t; W_{dec}) \quad (6)$$
$$l^i_t = W_s \cdot (\mu_x, \mu_y)_{t-1} \quad (7)$$

where $t \in [T_{obs} + 1, T_{obs} + 2, \ldots, T_{pred}]$. The LSTM decoder is denoted as $LSTM_{dec}(\cdot)$, and W_{dec} is the LSTM decoder weights. W_s is the linear embedding weights for spatial encoding.

D. Position Estimation and Loss Function

There are two commonly used loss functions for trajectory prediction. L2 loss is used for deterministic prediction, and negative log-likelihood (NLL) loss is used for probabilistic prediction. In the proposed model, we predict the bi-variate Gaussian distribution using NLL loss.

We assume the positions of pedestrians are random variables, and the i^{th} pedestrian's position at time t follows bi-variate Gaussian distribution $\hat{Y}^i_t \sim \mathcal{N}(\mu^i_t, \sigma^i_t, \rho^i_t)$. We generate output sequence distribution for each time step, $(\mu_x, \mu_y, \sigma_x, \sigma_y, \rho)^i_t$, where the mean value of the position is $(\mu_x, \mu_y)^i_t$, the standard deviation is $(\sigma_x, \sigma_y)^i_t$, and ρ^i_t is the correlation coefficient. The probability density function of position (x, y) is shown in Eq. 8.

$$f(x,y) = \frac{1}{2\pi\sigma_x\sigma_y\sqrt{1-\rho^2}} \exp\left[-\frac{1}{2(1-\rho^2)} * \left(\frac{(x-\mu_x)^2}{\sigma_x^2} + \frac{(y-\mu_y)^2}{\sigma_y^2} - \frac{2\rho(x-\mu_x)(y-\mu_y)}{\sigma_x\sigma_y}\right)\right] \quad (8)$$

After we obtain the hidden states from the LSTM decoder, we use output transformation to get the final prediction. The output transformation from hidden states to the distributions is shown in Eq. 9.

$$[\mu_x, \mu_y, ln(\sigma_x), ln(\sigma_y), tanh^{-1}(\rho)]^i_t = o^i_t = W_o \cdot h^i_t \quad (9)$$

(a) Waymo (Source) (b) ETH-univ (c) ETH-hotel (d) UCY-zara1 (e) UCY-zara2 (f) UCY-univ

Fig. 3. Screenshots of the data showing the differences between the source dataset (Waymo Open Dataset) and target datasets (ETH and UCY datasets). These data are collected by various sensors at different locations from different views. For the Waymo data, we present the front camera image from the vehicle's view, but we use the annotated pedestrians from the bird's-eye-view collected by LiDAR.

where $t \in [T_{obs}+1, T_{obs}+2, \ldots, T_{pred}]$. The output of the LSTM decoder is denoted by h_t^i, and W_o are the linear weights for transforming the hidden state to the output. Instead of learning σ_x, σ_y, ρ directly, we learn the logarithm of the standard deviation $ln(\sigma_x), ln(\sigma_y)$, and the arctanh of the correlation coefficient $tanh^{-1}(\rho)$ to avoid negative values.

The NLL loss function is shown below:

$$L_{nll}(W_\theta) = -\sum_{t=T_{obs}+1}^{T_{pred}} \sum_{i=1}^{n} \log\left(f(x_t^i, y_t^i | \mu_x, \mu_y, \sigma_x, \sigma_y, \rho)\right) \tag{10}$$

where W_θ represents the learned network parameters.

IV. Experiments

A. Datasets

The large and abundant Waymo Open Dataset [49] is used as the source dataset. We train the model on the Waymo training set, and evaluate the non-transfer task on the Waymo test set. Five transfer tasks are evaluated, including two ETH target datasets and three UCY target datasets. In this section, we introduce the basic information of these datasets. Fig. 3 shows screenshots of different pedestrian behavior scenarios of the source and target datasets. The basic information of the datasets is shown in Table II.

a) Waymo Open Dataset: Waymo Open Dataset [49] contains 1,150 real-world road scenes collected from the vehicle's view, among which 450 scenes are collected in urban street scenarios. To investigate pedestrian behavior in urban scenarios, we use the 450 urban scenes, including 374 training records and 76 test records with 20 seconds duration. We divide the training records into a training set with 337 records and a validation set with 37 records. The 76 test records are used for non-transfer task evaluation.

The data are collected with high-resolution cameras and LiDARs. We use the annotations from LiDAR data that have a scan range of 75m. The 3-dimensional positions in the real world are pre-processed into 2-dimensional (x, y) positions from the bird's-eye-view, and used as ground truth for training and evaluation.

b) ETH and UCY datasets: ETH [50] and UCY [51] datasets are widely used by existing studies on pedestrian trajectory prediction. These datasets contain five different scenarios at fixed locations from the bird's-eye-view recorded by a camera. We evaluate five senes separately. The positions are pre-processed into real-world 2-dimensional (x, y) positions.

B. Input Alignment and Data Pre-processing

As shown in Table II, the source and target datasets are collected in different scenarios with various sensors, views, and data frequency, and contain different labeled objects. Therefore, to develop a transferable model, we first need to align the input of the datasets.

To avoid the impact of different sensors and calibration parameters, we only use pedestrian trajectories as input instead of using raw data from cameras or LiDARs. We pre-process all data into the same coordinate system and frequency to keep the input information of the model consistent. For the Waymo datasets, the pedestrians' real-world center positions (x, y) are used and pre-processed into the global coordinate with a fixed origin for each record to reduce the influence of the ego-vehicle's movement. For the ETH and UCY datasets, the labels of pedestrians are transformed from the image coordinate (u, v) into real-world center positions (x, y). To avoid the influence of different frequencies, we sample all sequences to 2.5 Hz.

C. Evaluation Metrics

The following two metrics are used to evaluate the prediction performance:

- The Average Displacement Error (ADE): the average distance between the ground truth and the predicted trajectories over all predicted time steps, as shown in Eq. 11.

$$ADE = \frac{\sum_{i \in n} \sum_{t=T_{obs}+1}^{T_{pred}} \|Y_t^i - \hat{Y}_t^i\|_2}{n \times (T_{pred} - T_{obs})} \tag{11}$$

- The Final Displacement Error (FDE): the average distance between the ground truth and the predicted trajectories for the final predicted time step, as shown in Eq. 12.

$$FDE = \frac{\sum_{i \in n} \|Y_t^i - \hat{Y}_t^i\|_2}{n}, t = T_{pred} \tag{12}$$

D. Baselines

To avoid the impact of various sensors and calibration parameters from different datasets, the baseline models we use for comparing the transferability take only pedestrian trajectories as input, and do not consider other input information such as camera images, LiDAR data, or scene map information. The following models are retrained on the Waymo dataset, and compared for the performance and transferability on both the source and target datasets.

TABLE II
BASIC INFORMATION OF SOURCE AND TARGET DATASETS

Dataset	Year	Objects	Sensors	View	Collected Location	Scenarios	Frequency
Waymo	2020	Pedestrians, vehicles, cyclists, signs	Cameras, LiDARs	Vehicle's view, bird's-eye-view	United States	Urban road traffic	10 Hz (LiDAR)
ETH	2009	Pedestrians	Camera	Bird's-eye-view	Switzerland	Urban outdoor	2.5 Hz (Sampled)
UCY	2007	Pedestrians	Camera	Bird's-eye-view	Cyprus	Urban outdoor	2.5 Hz (Sampled)

1) Social LSTM: proposed by Alahi et al. [5] in 2016, considering individual trajectories of target pedestrians and social interactions, using L2 loss.
2) TF Individual: proposed by Giuliari et al. [15] in 2020, considering only individual trajectories of target pedestrians, using L2 loss.
3) Social STGCNN: proposed by Mohamed et al. [7] in 2020, considering individual trajectories of target pedestrians and social interactions, using NLL loss.
4) Social IWSTCNN: proposed by Zhang et al. [12] in 2021, considering individual trajectories of target pedestrians and social interactions, using NLL loss.
5) DMRGCN: proposed by Bae and Jeon [24] in 2021, considering individual trajectories of target pedestrians and social interactions, using NLL loss.
6) Spatial-Temporal-Spectral (STS) LSTM (ours): our proposed model as described in Sec. III, considering only individual trajectories of target pedestrians, using NLL loss.

In addition to the aforementioned baseline models, we also compare our proposed model with SoPhie [19], Social BiGAT [21], and HSTA [18]. Since these models either require other inputs than just trajectories or have not provided publicly available code, we cannot retrain and test their transferability. Therefore, we directly use the evaluation results on the ETH and UCY datasets from the original papers and compare them with the performance of our transferable model. The comparison shows the performance of our transferable model on the ETH and UCY datasets without prior knowledge compared to the models trained directly on those datasets.

E. Experimental Setting and Implementation Details

1) Model transferability: Experiments are designed to evaluate the transferability of the models. As shown in Fig. 1, all models for comparison are trained from scratch on the Waymo training set. For the non-transfer task, we evaluate the model on the Waymo test set. The data of training and test set share the same prior knowledge, and are collected in similar scenarios. For transfer tasks, we evaluate the model on ETH-univ, ETH-hotel, UCY-zara1, UCY-zara2, and UCY-univ datasets without knowing any prior information and without access to these datasets.

2) Model performance evaluation: We evaluate and compare the model transferability of our proposed model with the baselines as described in Sec. IV-D. The probabilistic models that we compare with [7], [12], [24] used the best of 20 samples for evaluation and comparison. To better align and compare the results, we follow this setting and use the best of 20 samples as results. The observation and prediction lengths are set the same for all datasets. We observe 8 time steps (i.e. 3.2 s) and predict 12 time steps (i.e. 4.8 s), following the setting in baseline models [5], [7], [12], [15], [24].

3) Inference speed: We are concerned about the inference speed of the models. We test the inference speed of four competitive models, including our proposed STS LSTM, Social STGCNN, Social IWSTCNN, and DMRGCN. We compare the total inference time, that consists of data processing, graph building, and network prediction time. The inference speed is tested on the Waymo test set.

4) Prediction network structures: To select the most transferable prediction network structure, we evaluate and compare the LSTM, CNN, and Transformer for both non-transfer and transfer tasks. We use the conventional spatial position in time series as inputs with multilayer perceptron (MLP) embeddings, and consider only individual trajectory features.

5) Loss functions: We implemented and evaluated both L2 loss and NLL loss for all three network structures.

6) Input representations: To compare and analyze how feature representation affects transferability, we evaluate input features with different components, including time series features, spectral features, and the proposed spatial-temporal-spectral representation with both features.

7) Implementation details: The Nvidia GeForce RTX 2080 Ti GPU was used for training and evaluation. Our proposed model was trained with the Adam Optimizer, with the batch size setting to 16 for 200 epochs. The learning rate was set to 1e-4. We use the displacement between each step as spatial inputs for calculating temporal and spectral features.

V. RESULTS AND ANALYSIS

A. Quantitative Results

1) Model performance and transferability: The quantitative evaluation results are shown in Table III. For SoPhie [19], Social BiGAT [21], and HSTA [18], we compare the results from the original papers without testing the transferability. The results show that compared with these non-transferable models trained on the ETH and UCY datasets, our transferable model achieves better results without access to the datasets. This demonstrates the potential of using off-the-shelf pre-trained models for practical tasks.

To test and compare the transferability, the models we select for comparison use only pedestrian trajectories as input, without image or map information. Our proposed model and the TF Individual model [15] consider only the information of single individuals' trajectories, while the other models including Social LSTM [5], Social STGCNN [7], Social IWSTCNN [12],

TABLE III
THE ADE/FDE METRICS (IN METERS) OF BASELINE METHODS COMPARED TO OUR PROPOSED STS LSTM MODEL. LOWER IS BETTER. METHODS ABOVE THE DASHED LINE: RESULTS FROM THE ORIGINAL PAPERS WITHOUT TESTING THE TRANSFERABILITY. METHODS BELOW THE DASHED LINE: MODELS RETRAINED ON THE SOURCE DATA, AND THE TRANSFERABILITY IS EVALUATED. METHODS MARKED WITH THE STAR *: USED THE BEST OF 20 SAMPLES. METHODS MARKED WITH SUPERSCRIPT 1: CONSIDERED SOCIAL INTERACTIONS AND SCENE INFORMATION. METHODS MARKED WITH SUPERSCRIPT 2: CONSIDERED SOCIAL INTERACTIONS. **BOLD**: BEST, UNDERLINE: SECOND BEST.

Model (Year)	Waymo test (Source Data)	ETH-univ	ETH-hotel	UCY-zara1	UCY-zara2	UCY-univ	Average (Target Data)
SoPhie*[1] (2019) [19]	– / –	0.700 / 1.430	0.760 / 1.670	0.300 / 0.630	0.380 / 0.780	0.540 / 1.240	0.540 / 1.150
Social BiGAT*[1] (2019) [21]	– / –	0.690 / 1.290	0.490 / 1.010	0.300 / 0.620	0.360 / 0.750	0.550 / 1.320	0.480 / 1.000
HSTA*[2] (2021) [18]	– / –	0.380 / 0.620	0.400 / 0.790	0.340 / 0.710	0.320 / 0.680	0.550 / 1.170	0.400 / 0.790
Social LSTM[2] (2016) [5]	0.393 / 0.841	0.696 / 1.375	0.365 / 0.707	0.448 / 0.972	0.360 / 0.795	0.561 / 1.214	0.486 / 1.013
TF Individual (2020) [15]	0.363 / 0.782	0.647 / 1.287	0.322 / 0.623	0.429 / 0.955	0.324 / 0.720	0.518 / 1.141	0.448 / 0.945
Social STGCNN*[2] (2020) [7]	0.335 / 0.550	0.418 / 0.681	0.253 / 0.360	0.352 / 0.610	0.303 / 0.515	0.401 / 0.753	0.346 / 0.584
Social IWSTCNN*[2] (2021) [12]	0.328 / 0.538	0.405 / **0.644**	0.209 / 0.285	0.340 / 0.581	0.292 / 0.506	0.412 / 0.775	0.332 / 0.558
DMRGCN*[2] (2021) [24]	**0.275 / 0.468**	**0.390** / 0.664	**0.193 / 0.266**	**0.299 / 0.541**	0.252 / **0.451**	**0.365 / 0.680**	**0.300 / 0.520**
STS LSTM* (ours)	0.284 / 0.532	0.457 / 0.813	0.203 / 0.282	0.306 / 0.567	**0.243** / 0.476	0.373 / 0.700	0.316 / 0.568

TABLE IV
INFERENCE SPEED COMPARISON (IN MS). **BOLD**: FASTEST.

Model (Year)	Inference Time per sequence
Social STGCNN (2020) [7]	15.81
Social IWSTCNN (2021) [12]	3.38
DMRGCN (2021) [24]	31.13
STS LSTM (ours)	**3.08**

and DMRGCN [24] consider also social interactions between pedestrians.

Compared with the baseline models, our proposed STS LSTM model gets the second-best performance on the non-transfer task on the Waymo test set. The ADE and FDE achieve 0.284 m and 0.532 m respectively, slightly worse than the previous best SOTA method DMRGCN. The ADE of our model is only less than 1 cm behind DMRGCN, while our proposed model has a much simpler structure. The proposed STS LSTM does not calculate the complex spatial and temporal interaction with other pedestrians and hence, it does not require much computational resource. Compared with Social STGCNN and Social IWSTCNN which consider social interactions, our proposed STS LSTM achieves better performance. This shows the effectiveness of the proposed spatial-temporal-spectral feature encoding module. Using the proposed feature encoding module, our model can learn the general motion pattern of pedestrians and obtain accurate predictions even without considering the status of their neighboring pedestrians.

When transferring to other target datasets, although DMRGCN achieves the best average results on target datasets, it has the most complicated network structure that considers both spatial and temporal interactions. On four out of five datasets, including ETH-hotel, UCY-zara1, UCY-zara2, and UCY-univ datasets, the ADE of our model is only marginally worse than DMRGCN with a difference of less than 1 cm. Our proposed model outperforms the other baseline methods on these four datasets using a simple LSTM encoding-decoding structure with spectral domain information. Another competitive model is the Social IWSTCNN, which performs better on the ETH-univ dataset, while it also uses a more complicated structure than ours that considers social interactions between pedestrians within crowds. This indicates better transferability of the proposed model to other unseen cases compared to existing methods.

We compare the prediction errors of the proposed STS LSTM model on the source and target datasets. The performance of the model is similar or slightly decreased on the ETH-hotel, UCY-zara1, and UCY-zara2 datasets, while the performance drops sharply on the ETH-univ and UCY-univ datasets. To better understand model transferability, we look into the property of the source and target datasets. The source Waymo dataset is collected in urban street scenarios with densely populated pedestrians. The three target datasets: ETH-hotel, UCY-zara1, and UCY-zara2 datasets are collected in similar crowded urban street scenarios, so they have similar performance to the source data and good transferability. The ETH-univ and UCY-univ datasets are collected in university scenarios. The pedestrian moving patterns are very different from those in the urban street scenarios. So the performance is worse compared to the source data. Therefore, these datasets of pedestrians with different motion patterns are harder for the models to transfer.

2) Inference speed: To investigate the potential of using prediction models in practical applications, we are also interested in the computational performance of the models. The inference speed of four competitive models, including Social STGCN, Social IWSTCNN, DMRGCN, and our proposed STS LSTM model are compared. The total inference time per sequence is listed in Table IV. Our proposed method achieves the best inference time of 3.08 ms per sequence. Although DMRGCN has less prediction error, the complicated structure, data processing, and graph building lead to the longest total prediction time of 31.13 ms, which is 10 times slower than our proposed model. The simpler structure design and less computational requirement of our proposed model demonstrate the potential of applying the model in practice.

B. Ablation Study

1) Prediction network structures and loss functions: To find a prediction network structure with good transferability, we compare three commonly used network structures, including LSTMs, CNNs, and Transformers, with two different loss

TABLE V
THE ADE/FDE METRICS (IN METERS) FOR LSTM, CNN, AND TRANSFORMER PREDICTION STRUCTURES USING L2 AND NLL LOSS. FOR MODELS USING NLL LOSS THAT OUTPUT DISTRIBUTIONS, WE COMPARE THE BEST OF 20 SAMPLES. LOWER IS BETTER. **BOLD**: BEST.

Model	Loss Type	Waymo test set (Source Data)	ETH-univ	ETH-hotel	UCY-zara1	UCY-zara2	UCY-univ	Average (Target Data)
CNN	L2	0.512 / 1.022	0.720 / 1.404	0.380 / 0.712	0.507 / 1.049	0.414 / 0.855	0.595 / 1.251	0.523 / 1.054
LSTM	L2	0.380 / 0.812	0.708 / 1.392	0.356 / 0.660	0.453 / 1.001	0.341 / 0.755	0.523 / 1.140	0.476 / 0.990
Transformer	L2	0.363 / 0.782	0.647 / 1.287	0.322 / 0.623	0.429 / 0.955	0.324 / 0.720	0.518 / 1.141	0.448 / 0.945
CNN	NLL	0.334 / 0.571	0.488 / 0.837	**0.204 / 0.301**	0.369 / 0.639	0.282 / 0.504	0.422 / 0.788	0.353 / 0.614
LSTM	NLL	**0.291 / 0.537**	**0.457 / 0.790**	0.233 / 0.374	**0.304 / 0.547**	**0.247 / 0.468**	**0.374 / 0.708**	**0.323 / 0.577**
Transformer	NLL	0.328 / 0.756	0.577 / 1.284	0.238 / 0.479	0.428 / 1.055	0.304 / 0.730	0.457 / 1.045	0.401 / 0.919

TABLE VI
THE ADE/FDE METRICS (IN METERS) FOR DIFFERENT INPUT REPRESENTATIONS USING THE LSTM STRUCTURE WITH NLL LOSS. WE COMPARE THE BEST OF 20 SAMPLES. LOWER IS BETTER. **BOLD**: BEST.

Model	Waymo test set (Source Data)	ETH-univ	ETH-hotel	UCY-zara1	UCY-zara2	UCY-univ	Average (Target Data)
Temporal LSTM	0.291 / 0.537	0.457 / **0.790**	0.233 / 0.374	**0.304 / 0.547**	0.247 / **0.468**	0.374 / 0.708	0.323 / 0.577
Spectral LSTM	0.286 / 0.546	**0.454** / 0.800	0.222 / 0.343	0.305 / 0.571	0.244 / 0.479	0.373 / 0.714	0.320 / 0.581
STS LSTM	**0.284 / 0.532**	0.457 / 0.813	**0.203 / 0.282**	0.306 / 0.567	**0.243** / 0.476	**0.373 / 0.700**	**0.316 / 0.568**

functions including L2 and NLL loss. For each model, the spatial positions of individual pedestrian trajectories in the temporal order are used as input features. The evaluation results are shown in Table V.

For models using L2 loss, the Transformer performs best and achieves the least errors on both source data and most target datasets. This is consistent with Giuliari et al.'s conclusion [15], that the Transformer predictor outperforms other individual LSTM-based approaches due to its attention mechanism. The CNN performs worst on both source and target datasets when using L2 loss. This may be because CNNs are not designed for capturing the dependencies in time series data, and they may fail to learn pedestrian motion patterns when it is used without considering the randomness of pedestrian motion. The LSTM model achieves relatively competitive but slightly worse results compared with the Transformer on both the source and target datasets. Without the help of the attention mechanism, LSTMs can only process input sequences one step at a time, so they may fail to capture more complex relationships between different time steps.

Compared to the deterministic models using L2 loss, the errors of the probabilistic models using NLL loss are much lower. This is because the probabilistic models evaluating the best of 20 samples are comparing the "upper-bound" of the performance that they can achieve. Another reason is that the prediction using L2 loss provides a deterministic "average" result and loses the randomness of pedestrian behavior, while the distribution prediction with NLL loss provides sample-based results and preserves the randomness of human motion. Therefore, to keep the random nature of pedestrian motion and obtain better prediction results, we choose NLL loss for regression and perform sample-based prediction.

For models using NLL loss, the LSTM model performs the best on the source and target datasets, while the Transformer does not perform well. The CNN and LSTM are competitive on different prediction tasks. The LSTM model performs better on the source dataset and four target datasets, including ETH-univ, UCY-zara1, UCY-zara2, and UCY-univ datasets, while the CNN model performs better on the ETH-hotel dataset. The four datasets with better performance of the LSTM model are denser compared to the ETH-hotel dataset, whose statistics are shown in Table I. Since the models are trained on Waymo data which is also dense, the results show that the LSTM model with NLL loss has better transferability. Therefore, we choose the LSTM model as the prediction structure with NLL loss as the loss function for more accurate predictions of source data and better transferability to target data.

2) Input representations: We compare the performance and transferability of different input representations, including a) only temporal representation, b) only spectral representation, and c) our proposed spatial-temporal-spectral representation. The LSTM encoder-decoder is used as the prediction structure with NLL loss. The evaluation results are presented in Table VI.

The evaluation results show that using only temporal representation and only spectral representation achieves similar prediction performance. This is because we obtain the spectrum of the pedestrian motion using the Discrete Fourier Transform which is an invertible operation, thus, both kinds of input features contain the same invertible information. When using temporal information as input features, the LSTM encoder learns the most important temporal information about pedestrian positions. When using the spectrum of pedestrian motion as input features, the LSTM encoder learns the most important spectrum information, i.e. the most important frequency components. The model gets better prediction results on the source dataset and has the best transferability on more target datasets when using both temporal and spectral information as input features. This is because when combining the time series and spectral information, the LSTM encoder extracts the most important motion features in the spatial-temporal domain from time series input, while also learning the most important frequency features in the spectral domain.

C. Visualization and Analysis

In this section, we visualize the inputs and outputs of the proposed model to better understand the data and model transferability.

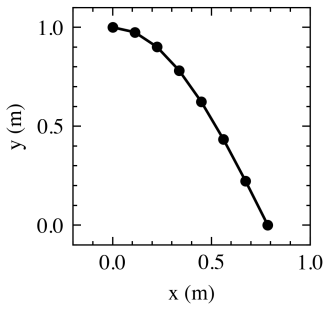


Fig. 4. An example of a non-linear trajectory in x-y coordinates from the bird's-eye-view.

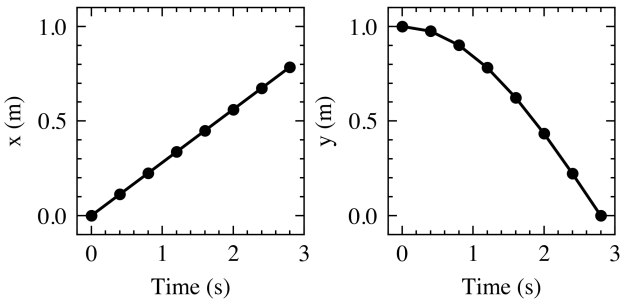


Fig. 5. The time series of the example in x and y directions.

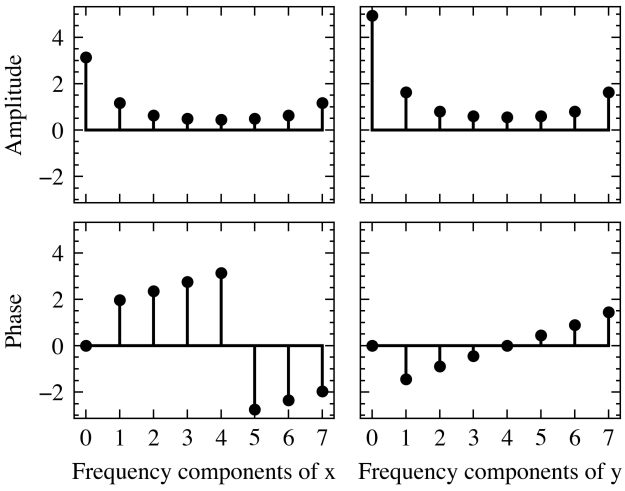


Fig. 6. The spectral sequences of the example in x and y directions.

1) Input representations: We provide a simplified example to show how spectral features improve feature representation. A non-linear trajectory is shown in Fig. 4 as an example. The data is represented in x-y coordinates from the bird's-eye-view. Eight time steps are observed at a frequency of 2.5 Hz. Traditional prediction networks use time series of x and y directions as input features, as shown in Fig. 5. The time series

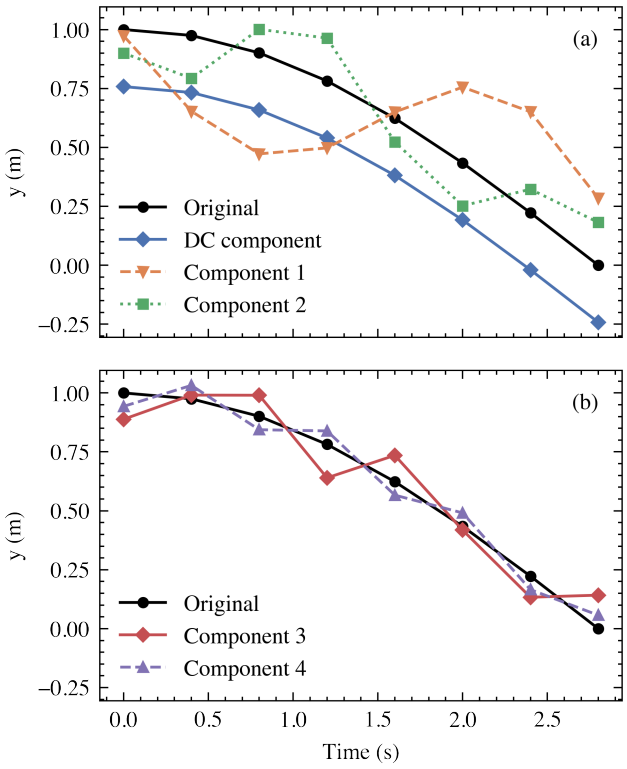
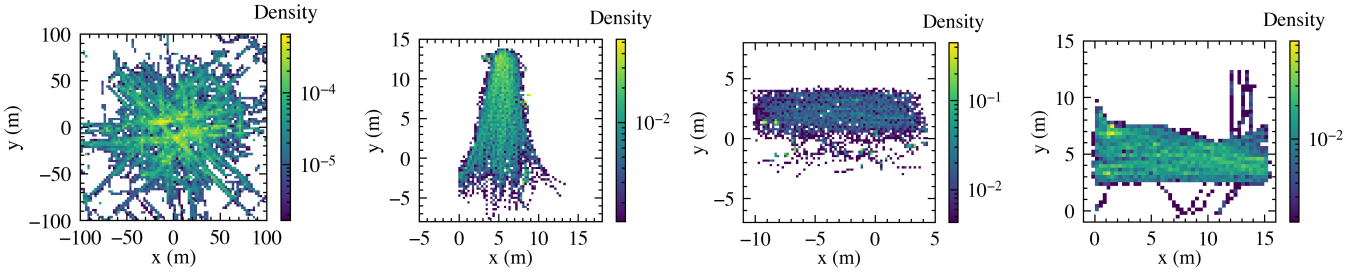


Fig. 7. The effects of different frequency components for the time series in the y direction. (a) the impacts of low-frequency components. (b) the impacts of high-frequency components.

in the x direction is linear, which is easy to predict, while the sequence in the y direction is non-linear, which is hard for the network to capture the pattern. To extract the general motion pattern of pedestrians, we add spectral domain information as input features. The spectral representation of the x and y directions is shown in Fig. 6. The spectrum provides additional information about different frequency components that can represent motion patterns at different scales.

To demonstrate how the low-frequency and high-frequency components in the spectral domain impact the time series, we take the non-linear sequence in the y direction as an example. Fig. 7 shows how the time series changes with different frequency components. When modifying the direct-current (DC) component, we get a sequence in the same shape but with a different average value. The DC component represents the average value of the time series signal. It represents the constant property of a pedestrian's movement. As shown in Fig. 7 (a), when modifying low-frequency (LF) components, including components 1 and 2, the coarse-level shape of the sequence changes. This indicates that the LF components reflect the macroscopic moving trends such as the pedestrian's intention and destination. As shown in Fig. 7 (b), when modifying the high-frequency (HF) components, including components 3 and 4, the time series is "fine-tuned" with fine-level changes. This indicates that the HF components reflect the microscopic moving trend such as the pedestrian's adjustment to the interaction with other road users and their moving preferences. This implies that using displacements and



(a) Waymo (b) ETH-univ (c) ETH-hotel (d) UCY-zara1

Fig. 8. Distributions of pedestrian trajectories (positions) for the source dataset (Waymo) and three target datasets (ETH-univ, ETH-hotel, and UCY-zara1). Density is presented in log scales. Displayed in white when no pedestrians have passed.

their spectrum as inputs allows the model to learn not only the temporal features but also pedestrians' moving trends in different scales.

2) Data distribution: The 2D distributions of pedestrian trajectories for the source dataset (Waymo) and three target datasets (ETH-univ, ETH-hotel, and UCY-zara1) are shown as log scale heatmaps in Fig. 8. The distributions vary greatly in different scenarios. the source data is collected on a moving vehicle, while three target datasets are collected from fixed scenarios.

To represent the motion states of pedestrians in the temporal domain, the displacements between adjacent frames (equivalent to velocities) on the x-axis and y-axis in time series are used as inputs. Distributions of the displacement on the x-axis and y-axis are shown in the violin plot in Fig. 9.

The peaks of the x and y velocities are close to zero, indicating that most pedestrians walk along the streets in the direction of the x-axis or y-axis. There are two secondary peaks in the Waymo dataset on both the x-axis and y-axis at about 0.4 m, indicating that there are also many people walking along the streets in diagonal directions. In the ETH-univ dataset, the velocities on the y-axis are larger and more widely distributed than that of the Waymo dataset, and the velocities on the x-axis are more concentrated at zero, indicating that most pedestrians walk along the y-axis direction passing through the gate. The ETH-hotel dataset has a distribution of velocities concentrated at zero on the y-axis, indicating that most pedestrians walk along the street in the direction of the x-axis, and there are also some people standing static waiting to get on the tram or cross the road. The distribution of the UCY-zara1 dataset has a peak near zero on the y-axis, indicating that most pedestrians walk along the x-axis passing the corner of the store.

Distributions of the amplitude of four frequency components in the x-axis and y-axis directions are shown in Fig. 10. The DC component that represents the static property is not plotted in the figure. The other frequency components represent the moving property. As the spectrum is symmetrical, we only plot the first four frequency components.

The frequency components shown in Fig. 10 provide additional dimensions for representing input data, extending beyond the sole utilization of displacement in the x and y axes as shown in Fig. 9. This implies that adding spectral information can supplement the representation and improve transferability.

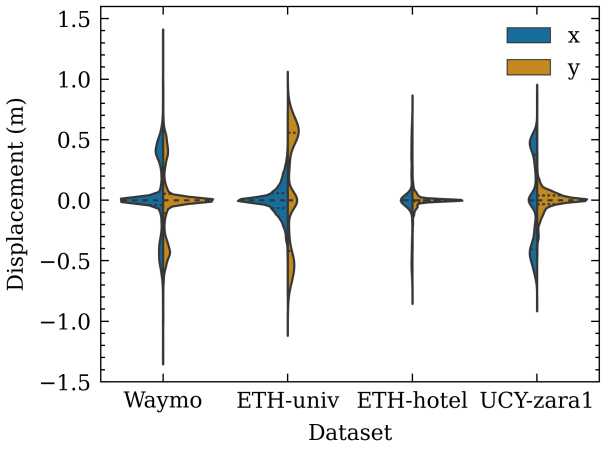


Fig. 9. Violin plots of the displacement (velocity) distributions for the source dataset (Waymo) and three target datasets (ETH-univ, ETH-hotel, UCY-zara1). Violin plots are shown with quartiles representing the 25th (top line), 50th (middle line), and 75th (bottom line) percentiles of the distribution.

As shown in Fig. 10, the distribution of frequency components of the ETH-univ dataset is greatly different from the Waymo dataset. This is a possible reason for the drop in the accuracy of transfer tasks on the ETH-univ dataset.

3) Examples of prediction results on target datasets: Fig. 11 presents qualitative examples of prediction results on three target datasets. As shown in the figure, even without any prior knowledge and without access to the target datasets, our proposed model can predict position distributions that are close to the ground truth. Our model predicts future position distributions, and we plot 99.7% confidence interval here. For the ETH-univ and the UCY-zara scenarios, the prediction shows small uncertainty. This indicates our model is able to deal with moving cases where pedestrians are walking. For the ETH-hotel scenario, the uncertainty is larger, especially for the static pedestrians standing by the roadside. This implies that our model tends to predict the motion of stationary pedestrians with large uncertainty to deal with randomness, indicating that those stationary pedestrians may move at any time.

4) Limitations: Since our model only explicitly considers the individual trajectories, it struggles to handle the cases related to pedestrians' intentions and complex interactions.

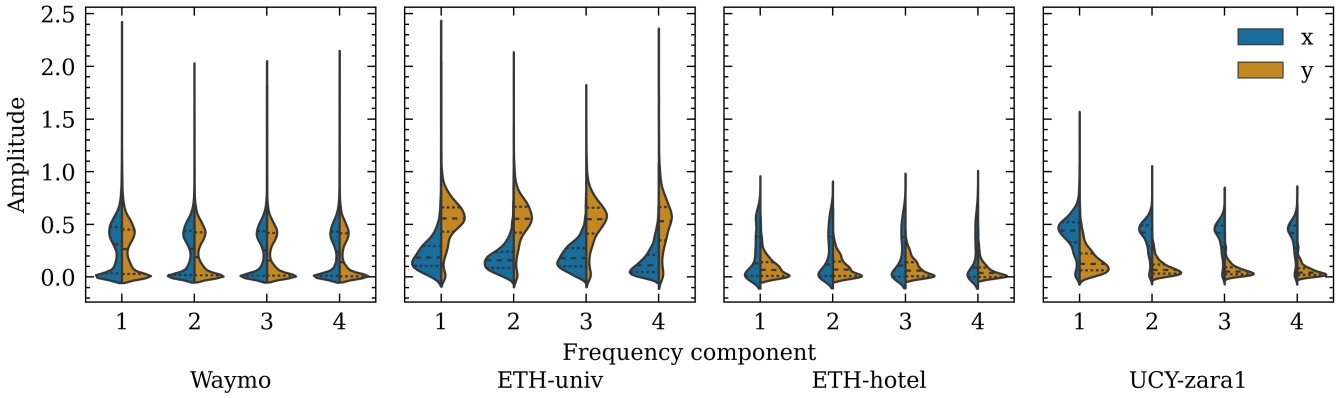
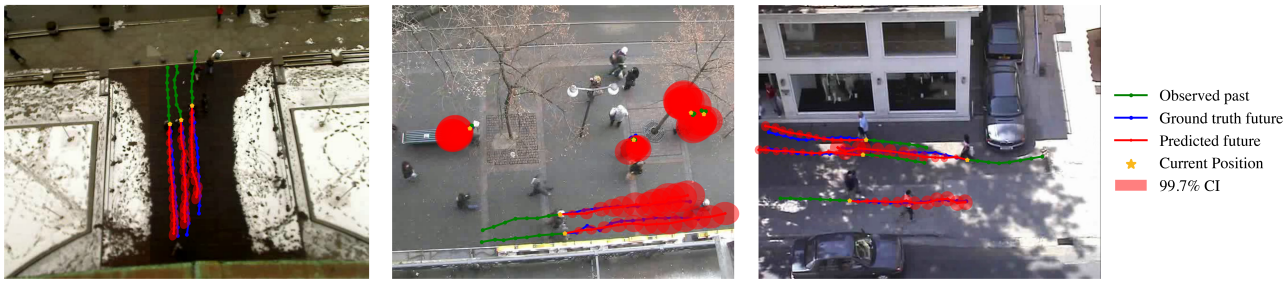


Fig. 10. Violin plots of the amplitude distributions of four frequency components for the source dataset (Waymo) and three target datasets (ETH-univ, ETH-hotel, UCY-zara1). Violin plots are shown with quartiles representing the 25th (top line), 50th (middle line), and 75th (bottom line) percentiles of the distribution.



(a) ETH-univ (b) ETH-hotel (c) UCY-zara

Fig. 11. Examples of prediction results on three target datasets, including the ETH-univ, ETH-hotel, and UCY-zara dataset, with 99.7% confidence interval.

Fig. 12 shows examples of failed predictions. The failure occurs when pedestrians move suddenly or accelerate at low speeds. Sudden movements of pedestrians are usually related to their intentions. For example, a) the pedestrians in Fig. 12c continue walking after looking at merchandise in the shop window, and b) the pedestrians are hesitating to enter the gate as in Fig. 12a. Besides, when there are other pedestrians or objects that pedestrians interact with, the proposed model does not perform well, such as the cases in Fig. 12a and Fig. 12b. This is because our proposed model only considers individual movements without social interactions. These failures are due to the lack of information about the intentions and interactions of pedestrians. Since our proposed method is a simple base structure for transfer learning, it can be combined with other intention-based or interaction-based models to improve domain-specific performance.

VI. Conclusions

In this paper, we focus on developing a transferable model for pedestrian trajectory prediction. Specifically, the STS LSTM model is proposed that performs well on both source and target datasets with a faster inference speed compared with the SOTA methods. We have evaluated and analyzed the transferability of the proposed model and several popular existing pedestrian trajectory prediction models. Experimental results show that among the three commonly used neural network structures including LSTMs, CNNs, and Transformers, and two commonly used loss functions including L2 loss and negative log-likelihood loss, the best prediction results are achieved by using LSTMs with NLL loss. By comparing using only time series, only spectrum, and our proposed spatial-temporal-spectral representation as input features, our proposed representation can effectively represent the pedestrian motion pattern and achieves the best transferability. In future work, our proposed model as a transferable base model can be combined with other intention-based models or interaction-based models to improve performance.

Acknowledgments

This research is funded by the European research project "SHAPE-IT - Supporting the Interaction of Humans and Automated Vehicles: Preparing for the Environment of Tomorrow". This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement 860410. The authors would like to thank Assistant Professor Dr. Yinan Yu and Samuel Scheidegger for their valuable comments and suggestions. Wikimedia user Auguel is acknowledged for providing the icon of the multivariate Gaussian demonstration.

References

[1] WHO, "Global status report on road safety 2018: Summary," World Health Organization," Report, 2018.

(a) ETH-univ (b) ETH-hotel (c) UCY-zara1

Fig. 12. Examples of failure prediction results on three target datasets, including ETH-univ, ETH-hotel, and UCY-zara dataset with 99.7% confidence interval.

[2] A. Rasouli, I. Kotseruba, and J. K. Tsotsos, "Are they going to cross? a benchmark dataset and baseline for pedestrian crosswalk behavior," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) Workshops*, Oct 2017, Conference Proceedings, pp. 206–213.

[3] A. Rasouli, I. Kotseruba, T. Kunic, and J. K. Tsotsos, "Pie: A large-scale dataset and models for pedestrian intention estimation and trajectory prediction," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, Conference Proceedings, pp. 6262–6271.

[4] C. Zhang, A. H. Kalantari, Y. Yang, Z. Ni, G. Markkula, N. Merat, and C. Berger, "Cross or wait? predicting pedestrian interaction outcomes at unsignalized crossings," *arXiv preprint arXiv:2304.08260*, 2023.

[5] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, F.-F. Li, and S. Savarese, "Social lstm: Human trajectory prediction in crowded spaces," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2016, pp. 961–971.

[6] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi, "Social gan: Socially acceptable trajectories with generative adversarial networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2255–2264.

[7] A. Mohamed, K. Qian, M. Elhoseiny, and C. Claudel, "Social-stgcnn: A social spatio-temporal graph convolutional neural network for human trajectory prediction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 14 424–14 432.

[8] C. Zhang and C. Berger, "Pedestrian behavior prediction using deep learning methods for urban scenarios: A review," 2023, accepted in IEEE Transactions on Intelligent Transportation Systems, Manuscript DOI: 10.1109/TITS.2023.3281393.

[9] H. Xue, D. Q. Huynh, and M. Reynolds, "Ss-lstm: A hierarchical lstm model for pedestrian trajectory prediction," in *2018 IEEE Winter Conference on Applications of Computer Vision*. IEEE, 2018, pp. 1186–1194.

[10] T. Fernando, S. Denman, S. Sridharan, and C. Fookes, "Soft+ hardwired attention: An lstm framework for human trajectory prediction and abnormal event detection," *Neural networks*, vol. 108, pp. 466–478, 2018.

[11] R. Chandra, U. Bhattacharya, A. Bera, and D. Manocha, "Traphic: Trajectory prediction in dense and heterogeneous traffic using weighted interactions," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8483–8492.

[12] C. Zhang, C. Berger, and M. Dozza, "Social-iwstcnn: A social interaction-weighted spatio-temporal convolutional neural network for pedestrian trajectory prediction in urban traffic scenarios," in *2021 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2021, pp. 1515–1522.

[13] C. Zhang and C. Berger, "Learning the pedestrian-vehicle interaction for pedestrian trajectory prediction," in *2022 the 8th International Conference on Control, Automation and Robotics (ICCAR)*. IEEE, 2022, pp. 230–236.

[14] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *31st Conference on Neural Information Processing Systems (NIPS)*, 2017, Conference Paper, pp. 1–11.

[15] F. Giuliari, I. Hasan, M. Cristani, and F. Galasso, "Transformer networks for trajectory forecasting," in *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, 2021, pp. 10 335–10 342.

[16] C. Yu, X. Ma, J. Ren, H. Zhao, and S. Yi, "Spatio-temporal graph transformer networks for pedestrian trajectory prediction," in *European Conference on Computer Vision*. Springer, 2020, Conference Proceedings, pp. 507–523.

[17] L. Torrey and J. Shavlik, "Transfer learning," in *Handbook of research on machine learning applications and trends: algorithms, methods, and techniques*. IGI global, 2010, pp. 242–264.

[18] Y. Wu, G. Chen, Z. Li, L. Zhang, L. Xiong, Z. Liu, and A. Knoll, "Hsta: A hierarchical spatio-temporal attention model for trajectory prediction," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 11, pp. 11 295–11 307, 2021.

[19] A. Sadeghian, V. Kosaraju, A. Sadeghian, N. Hirose, H. Rezatofighi, and S. Savarese, "Sophie: An attentive gan for predicting paths compliant to social and physical constraints," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1349–1358.

[20] J. Amirian, J.-B. Hayet, and J. Pettré, "Social ways: Learning multi-modal distributions of pedestrian trajectories with gans," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2019, pp. 2964–2972.

[21] V. Kosaraju, A. Sadeghian, R. Martín-Martín, I. Reid, H. Rezatofighi, and S. Savarese, "Social-bigat: Multimodal trajectory forecasting using bicycle-gan and graph attention networks," in *Advances in Neural Information Processing Systems*, 2019, Conference Proceedings, pp. 137–146.

[22] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," *arXiv preprint arXiv:1803.01271*, 2018.

[23] N. Nikhil and B. Tran Morris, "Convolutional neural network for trajectory prediction," in *Proceedings of the European Conference on Computer Vision Workshops*, 2018, pp. 186–196.

[24] I. Bae and H.-G. Jeon, "Disentangled multi-relational graph convolutional network for pedestrian trajectory prediction," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 2, 2021, pp. 911–919.

[25] Y. Yuan, X. Weng, Y. Ou, and K. Kitani, "Agentformer: Agent-aware transformers for socio-temporal multi-agent forecasting," *arXiv preprint arXiv:2103.14023*, 2021.

[26] A. Syed and B. Morris, "Stgt: Forecasting pedestrian motion using spatio-temporal graph transformer," in *2021 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2021, Conference Proceedings, p. 1553–1558.

[27] G. Habibi, N. Jaipuria, and J. P. How, "Context-aware pedestrian motion prediction in urban intersections," *arXiv preprint arXiv:1806.09453*, 2018.

[28] H. Manh and G. Alaghband, "Scene-lstm: A model for human trajectory prediction," *arXiv preprint arXiv:1808.04018*, 2018.

[29] N. Deo and M. M. Trivedi, "Trajectory forecasts in unknown environments conditioned on grid-based plans," *arXiv preprint arXiv:2001.00735*, 2020.

[30] X. Song, K. Chen, X. Li, J. Sun, B. Hou, Y. Cui, B. Zhang, G. Xiong, and Z. Wang, "Pedestrian trajectory prediction based on deep convolutional lstm network," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 6, pp. 3285–3302, 2020.

[31] K. Saleh, M. Hossny, and S. Nahavandi, "Intent prediction of pedestrians via motion trajectories using stacked recurrent neural networks," *IEEE Transactions on Intelligent Vehicles*, vol. 3, no. 4, pp. 414–424, 2018.

[32] A. Vemula, K. Muelling, and J. Oh, "Social attention: Modeling attention in human crowds," in *2018 IEEE international Conference on Robotics and Automation (ICRA)*. IEEE, 2018, Conference Proceedings, pp. 4601–4607.

[33] Y. Ma, X. Zhu, S. Zhang, R. Yang, W. Wang, and D. Manocha, "Trafficpredict: Trajectory prediction for heterogeneous traffic-agents," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, Conference Proceedings, pp. 6120–6127.

[34] M. Lisotto, P. Coscia, and L. Ballan, "Social and scene-aware trajectory prediction in crowded spaces," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2019, Conference Proceedings, p. 2567–2574.
[35] P. Kothari, S. Kreiss, and A. Alahi, "Human trajectory forecasting in crowds: A deep learning perspective," *arXiv preprint arXiv:2007.03639*, 2020.
[36] Y. Hu, S. Chen, Y. Zhang, and X. Gu, "Collaborative motion prediction via neural motion message passing," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, Conference Proceedings, pp. 6319–6328.
[37] K. Mangalam, H. Girase, S. Agarwal, K.-H. Lee, E. Adeli, J. Malik, and A. Gaidon, "It is not the journey but the destination: Endpoint conditioned trajectory prediction," *arXiv preprint arXiv:2004.02025*, 2020.
[38] C. Wong, B. Xia, Z. Hong, Q. Peng, W. Yuan, Q. Cao, Y. Yang, and X. You, "View vertically: A hierarchical network for trajectory prediction via fourier spectrums," in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXII*. Springer, 2022, pp. 682–700.
[39] Z. Ding, H. Xie, P. Li, and X. Xu, "A structural developmental neural network with information saturation for continual unsupervised learning," *CAAI Transactions on Intelligence Technology*, 2023.
[40] A. Sharif Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "Cnn features off-the-shelf: An astounding baseline for recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2014.
[41] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "Decaf: A deep convolutional activation feature for generic visual recognition," in *Proceedings of the 31st International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, E. P. Xing and T. Jebara, Eds., vol. 32, no. 1. Bejing, China: PMLR, 22–24 Jun 2014, pp. 647–655. [Online]. Available: https://proceedings.mlr.press/v32/donahue14.html
[42] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.
[43] J. Howard and S. Ruder, "Universal language model fine-tuning for text classification," *arXiv preprint arXiv:1801.06146*, 2018.
[44] N. Houlsby, A. Giurgiu, S. Jastrzebski, B. Morrone, Q. De Laroussilhe, A. Gesmundo, M. Attariyan, and S. Gelly, "Parameter-efficient transfer learning for NLP," in *Proceedings of the 36th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri and R. Salakhutdinov, Eds., vol. 97. PMLR, 09–15 Jun 2019, pp. 2790–2799. [Online]. Available: https://proceedings.mlr.press/v97/houlsby19a.html
[45] M. Shen, G. Habibi, and J. P. How, "Transferable pedestrian motion prediction models at intersections," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 4547–4553.
[46] N. Jaipuria, G. Habibi, and J. P. How, "Learning in the curbside coordinate frame for a transferable pedestrian trajectory prediction model," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, pp. 3125–3131.
[47] E. Zhang, S. Pizzi, and N. Masoud, "A learning-based method for predicting heterogeneous traffic agent trajectories: implications for transfer learning," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2021, pp. 1853–1858.
[48] Y. Xu, L. Wang, Y. Wang, and Y. Fu, "Adaptive trajectory prediction via transferable gnn," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 6520–6531.
[49] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine, *et al.*, "Scalability in perception for autonomous driving: Waymo open dataset," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 2446–2454.
[50] S. Pellegrini, A. Ess, K. Schindler, and L. Van Gool, "You'll never walk alone: Modeling social behavior for multi-target tracking," in *2009 IEEE 12th International Conference on Computer Vision*. IEEE, 2009, pp. 261–268.
[51] A. Lerner, Y. Chrysanthou, and D. Lischinski, "Crowds by example," in *Computer graphics forum*, vol. 26. Wiley Online Library, 2007, pp. 655–664.

Chi Zhang received the B.E. and M.E. degrees in control science and engineering from Zhejiang University, China, in 2014 and 2017, respectively. From 2017 to 2020, she was a research and development engineer at the Intelligence Driving Group, Baidu, Beijing, China, in the area of automated driving perception. Since 2020, she has been pursuing the Ph.D. degree with the Department of Computer Science and Engineering, University of Gothenburg, Gothenburg, Sweden. She is a Marie Curie Early Stage Researcher. Her research interests include applying deep learning on pedestrian behavior prediction and learning interactions between vulnerable road users and vehicles.

Zhongjun Ni received the B.Eng. and M.Eng. degrees in chemical engineering and technology from Zhejiang University, China, in 2014 and 2017, respectively. Between 2017 and 2020, he worked as a software engineer in the industry, such as Microsoft. He is currently pursuing his Ph.D. degree with the Department of Science and Technology, Linköping University, Sweden. His research interests include time series analysis, digital twins, and Internet of Things solutions based on Edge-Cloud computing.

Christian Berger Dr. Christian Berger is Full Professor at the Department of Computer Science and Engineering at University of Gothenburg, Sweden and received his Ph.D. degree from RWTH Aachen University, Germany in 2010. He coordinated the project for the vehicle "Caroline", which participated in the world's first urban robot race 2007 DARPA Urban Challenge Final. He co-led the Chalmers Truck Team during the 2016 Grand Cooperative Driving Challenge (GCDC), and is one of the two leading architects behind OpenDLV (Open Driverless Vehicle). His research expertise is on architecting complex and distributed realtime software systems, micro-services for cyber-physical and IoT-systems, continuous integration/deployment/experimentation, and data-driven software engineering.