

Received February 18, 2022, accepted March 15, 2022, date of publication March 23, 2022, date of current version March 30, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3161661

Vehicles Trajectory Prediction Using Recurrent VAE Network

MIGUEL ÁNGEL DE MIGUEL¹, (Graduate Student Member, IEEE), JOSÉ MARÍA ARMINGOL, AND FERNANDO GARCÍA¹, (Member, IEEE)

Systems Engineering and Automation Department, Universidad Carlos III de Madrid, 28911 Leganés, Spain

Corresponding author: Miguel Ángel de Miguel (mimiguel@ing.uc3m.es)

This work has been supported by the Madrid Government (Comunidad de Madrid) under the Multiannual Agreement with Universidad Carlos III de Madrid (UC3M) in the line of “Fostering Young Doctors Research” (PEVAUTO-CM-UC3M) within the V PRICIT (5th Regional Programme of Research and Technological Innovation) and by Grant PID2019-104793RB-C31 funded by MCIN/AEI/10.13039/501100011033.

ABSTRACT This paper presents an analysis of the implementation and performance of a deep learning model based on Recurrent layers and Variational Auto Encoder model (VAE) architecture for prediction of future local trajectory and maneuver. The proposed method uses the encoder part of the VAE to represent the vehicle’s surroundings agents behavior in time, taking advantage on the fact that VAE encodes similar situations or states close in the latent space and the generative properties of the VAE decoder, that is used to generate naturalistic driving trajectories. Furthermore, the variance of the predicted trajectory is estimated using the statistical properties of VAE model, increasing it if the input data is noisy or unrealistic and decreasing it if the model is certain about the prediction. The model is trained and evaluated with a public dataset. The results show that the proposed architecture outperforms state of the art methods in trajectory prediction error and provides a variance estimation that depends on input quality.

INDEX TERMS Intelligent vehicles, navigation, intelligent vehicles, land vehicles, prediction algorithms, trajectory prediction.

I. INTRODUCTION

An accurate and reliable trajectory prediction module, capable of predicting the future paths of the surrounding vehicles, is a key aspect for modern autonomous driving vehicles navigation algorithms. Those algorithms, such as path planning or decision making, need as input the future behavior of the surrounding vehicles to work properly. However, an accurate trajectory prediction is still a challenging problem due to the uncertainty on the behavior of vehicles and the dynamic nature of the environment. This issue is aggravated in the last seconds of a long term trajectory where it tends to be highly non-linear and unpredictable as it is much stronger influenced by the intentions of the drivers which may have independent goals or different driving styles [1]–[4].

Latest works on trajectory prediction benefit from the use of deep learning models which are more robust and generalize better. In particular, models such as Generative Adversarial Networks (GANs) [5] or Variational Auto-Encoder

The associate editor coordinating the review of this manuscript and approving it for publication was Venkateshkumar M¹.

(VAE) [6], can be very useful for this purpose thanks to their generative properties.

Above all, VAE models have become increasingly popular and have been used in a variety of works with excellent results typically for noise reduction [7], dimensionality reduction [8], [9] and realistic data generation [10]–[12]. Thus, trajectory forecasting can benefit from all those VAE capacities, as the input data for the prediction can be noisy, it is intended to extract the relevant features of the surroundings in a lower dimensional space and it is necessary to generate a realistic trajectory.

This work proposes a deep learning model based on VAE and Long-Short Term Memory (LSTM) layers that predicts the future maneuver of the vehicles in the surrounding, but also generates the future path that the vehicle will follow. This approach makes a deep study of the latent space generated by the VAE to prove that, on the one hand, close n-dimensional points correspond to similar situations, and on the other hand, far away ones to very different ones. It is also proved that all vehicles interactions and states over time are represented on this space.

Furthermore, the generator of the VAE is used to generate naturalistic trajectories, which are similar to the ones of the training set, generating not only the lane change maneuver detection, but also how it will be performed (acceleration, speed or how fast to leave and merge the lane).

This work is focused on highway scenarios as the dataset used provides highway data, but its flexibility makes it applicable to many different highway scenarios with different number of lanes or road geometry.

The proposed model is evaluated with state of the art methods in the same dataset, outperforming them in trajectory error.

The contributions of this paper are:

- VAE encoder is proven to generate a better latent representation that improves trajectory error.
- VAE decoder with LSTM is utilized to produce naturalistic paths.
- Uncertainty of the trajectory is estimated.
- Theoretical analysis of the performance of VAE applied to trajectory prediction is performed.

The remainder of this paper is organized as follows. In section II, relevant works are discussed. Section III motivates the dataset chosen, and describes the data pre-processing. Section IV describes the proposed architecture. Finally section V presents and discuss the results and section VI concludes the paper.

II. RELATED WORK

According to [13], trajectory prediction algorithms can be classified into three main types: based on a physics model, on a maneuver model or on an interaction model. Classic models rely on the definition of a physic model of the vehicle. This model can include either the geometry of the vehicle (kinematic model), or it can also consider the forces that affect its motion (dynamic model). This kind of model is commonly used for collision detection, as the time horizon is not very large. Physics-based models main drawback is the high dependence on the initial conditions, usually estimated with noisy sensors, and the limited information of the environment (road geometry and surrounding vehicles). Some examples of this kind of models can be found on [14], [15].

In the maneuver based models, the vehicle is considered independent from the others on the road, and the model has to predict, from a collection of different possible trajectories, which one the vehicle is more likely to perform. Although this kind of model can consider the surrounding obstacles or road geometry in order to discard a part of the maneuver set, not considering the interactions with other vehicles can lead to an undesired behavior. A representative example on this is [16], where the predicted path is sampled from a set of real trajectories according to the current situation.

The interaction aware models considers that vehicles motion is influenced by other vehicles in the scene. This kind of models rely on detecting and identifying the interaction precisely, but correctly modeling or considering those

interactions can be challenging. Dynamic Bayesian Networks have been used in different works [17], [18] to solve this problem, but in the recent years, the high flexibility and performance of machine learning, has lead on an extensive use of these techniques for prediction or forecasting. Latest relevant works in the topic, use deep neural network to predict the future path, where different layers or input representation are the main difference between works. In [19], authors use a 3D tensor that represents the potential field of the traffic scene in different time steps, whereas in [20] and [21] the interactions between vehicles are modeled using an attention layer or an interaction layer respectively. In [3], a convolutional social pooling layer is used to encode the past motion of the neighboring vehicles and a LSTM decoder generates the future trajectory (CS-LSTM).

There can be observed two main characteristics on the deep learning models reviewed: The use of LSTM layers in the models which improves their capacity of handling data time series [3], [22]–[26], and an encoder-decoder architecture, also named as sequence to sequence for LSTM models, which is extensively used in many different deep learning models like [27].

VAE follows the encoder-decoder architecture which is used in most of the reviewed works, but it is based on bayesian statistics. This type of model stands out for its generative capacities and the ability to produce a regular latent space with good properties for generation.

VAE has been used in different Autonomous Vehicles algorithms to generate a database of realistic trajectories: TrajVAE work [11] uses a VAE based model to generate realistic large-scale trajectories (from dozens of seconds to minutes) to create a dataset. However, the trajectories generated by this method are different to the ones studied in this paper, which are dependent on the surrounding vehicles, with up to five seconds prediction, demanding higher precision.

Multi-Vehicle Trajectory Generation (MTG) method [12], uses a VAE to generate trajectories of pairs of vehicles that drive through a road or an intersection. This method succeeds creating realistic trajectories, but it can not be used to predict trajectories of the surrounding vehicles, as this method is oriented to generate reliable testing scenarios, and does not include information of the surrounding vehicles.

Models based on GAN architecture are also known for generating realistic data. This kind of models have been used for predicting the future trajectory of a vehicle proving their utility as can be shown in [28]–[30]. However, a comparison between GANs and VAE models for trajectory generation performed in [11] and [12] shows how GANs can produce a higher quality reconstruction, but tend to lack full support over the data unlike VAE, which is essential in an unpredictable environment like a road [31].

The work presented in [1] generates prediction of vehicles trajectories using a conditional VAE (cVAE) that has as an additional input which is the intention predicted by an additional model that works as a classifier. This framework predicts future trajectories of the vehicles and outperforms

the compared methods. However, it highly relies on a good performance of the intention recognizer classifier that makes the cVAE generate one of three types of maneuver (keep in lane, merge left/right lane). In addition, the dataset used in this work, does not provide classes for the maneuvers, so they need to be added manually and the criteria for deciding if a trajectory belongs to a class is not covered. In contrast, the proposed method does not depend on a classifier and the number of classes is not limited to three, instead the model can learn any trajectory type.

The aim of this work is to apply a simple VAE model based on LSTM recurrent layers to generate a realistic future local trajectory. The proposed method considers vehicles interactions, as it includes time-distributed data of the surrounding vehicles within the date used to generate the path. These interactions are implicit in the model i.e. it does not analyze interactions independently.

III. DATASET AND DATA PREPROCESSING

To train the model, a driving dataset that includes vehicles trajectories in an highway is required. One of the most used dataset on this topic is the one provided by the Next Generation Simulation (NGSIM) project [32], which includes several hours of driving data. However, as studied in [33], this dataset has some problems in the data which can not be corrected throw filtering, such as wrong trajectories that collide with near ones or unrealistically large magnitudes.

Recently released highway drone (highD) dataset [34], has overcome most of the problems mentioned of NGSIM dataset, giving not only more accurate data, but also a larger collection of samples (highD dataset provides a total driven distance 9 times higher than NGSIM).

In addition, most of the latest state of the art works in the topic of trajectory prediction use highD dataset for both, training and evaluating the results [4], [19]–[21].

Thus, for this work, highD dataset is used.

A. PREPROCESSING AND NORMALIZATION

HighD Dataset includes dynamic information like position, speed and acceleration for every vehicle in the road at a timestamp. Combining and low-pass filtering that information, surrounding vehicles can be extracted, defining eight zones for every vehicle where possibly other vehicle can be driving as shown in Fig. 1. For every zone, a metric m_i is generated to represent the distance to the nearest vehicle in that zone. The expression of the metric for all zones except for zones 2 and 7 is given by

$$m_i = \begin{cases} \sqrt[3]{d_{max} - d_i}, & \text{if } d_i < d_{max} \\ 0, & \text{otherwise} \end{cases}, \quad (1)$$

where d_{max} is the value of the maximum distance to consider (set experimentally to a value of 200m) and real distance d_i is transformed to a metric m_i (that is input to the network) to better represent the absence of a vehicle in that zone and to evenly represent different distance values. For zones 2 and 7,

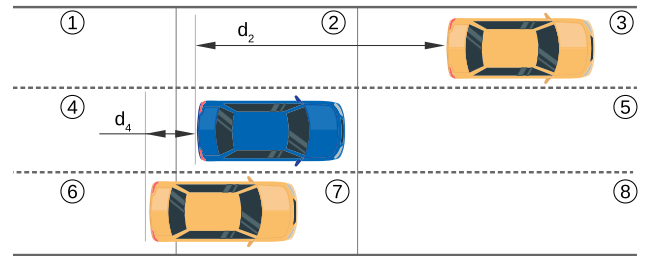


FIGURE 1. Zones representation around ego-vehicle (blue). Two other vehicles (yellow) are placed to give an example of how d_i is obtained.

m_i is set to 1 if a vehicle or part of a vehicle is inside the zone and 0 otherwise.

The rest of the values are normalized to fit a normal distribution, but position for coordinates is done in the following way. For the longitudinal coordinate, the first value is subtracted for all sequence to make them start on 0 and all the values are min-max normalized. Lateral coordinate in every road is transformed to fit a common one where lanes centers are in 0.25, 0.75 and 1.25 meters.

B. MANEUVER LABELING

A maneuver label is generated for the dataset, where every trajectory is classified into three categories: keeping lane and left/right lane change if at any moment of the trajectory at least one time step is close in time to a lane change maneuver. This label will not be utilized in the training as it provides future data leakage, but will be used to perform a deeper analysis on the proposed model in section IV-D.

C. DATA SPLIT

The dataset consists on 60 recording sequences. In order to properly evaluate the performance of the proposed model, the dataset is split into 45 sequences of train data and 15 of test data (75-25%) to make sure that the performance is evaluated in never seen sequences. In addition, 30% of train data is used for validation and hyperparameters tuning.

IV. PROPOSED MODEL ARCHITECTURE

This section describes the proposed model architecture: First, the VAE models used are described, followed by the model used to predict the future trajectories and a description of how the full model is trained. Finally, an analysis of the latent space generated by the encoder and the performance of the VAE itself is made.

A. VAE

A VAE model [6], is very similar in shape to a standard Auto-Encoder (AE), which uses an encoder-decoder architecture to reconstruct the input data but with some differences. The VAE encoder, instead of mapping the input data into a single multi-dimensional point, it generates a Gaussian distribution over that point defined by its mean and variance. The loss

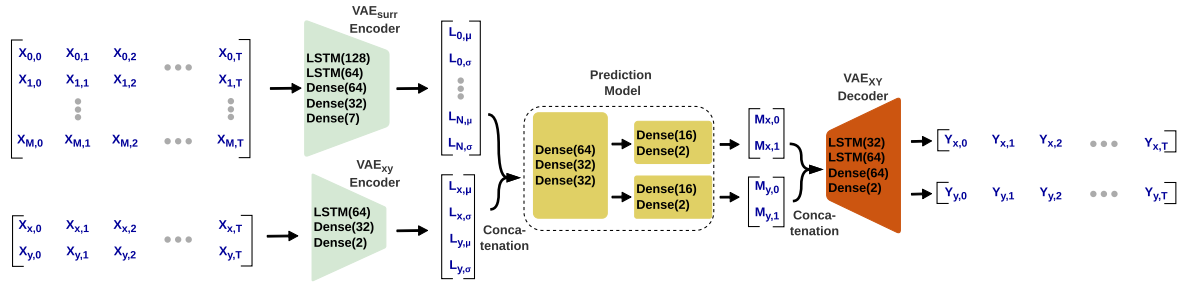


FIGURE 2. Proposed model full architecture. Each box represents a sequential model where the type of layers and the number of neurons are enumerated from top to bottom.

function includes an additional term to force the latent space be similar to a normal distribution [31].

The temporal nature of the problem makes the use of recurrent neural networks convenient for temporal feature extraction and time-series generation. The proposed model architecture for the VAE model uses LSTM layers, widely used for complex sequence learning problems [35], combined with fully-connected and time-distributed fully-connected layers.

The proposed model uses two VAE models, which are trained separately. The first model is trained with M features, corresponding to all the surrounding vehicles data (distances and speeds) for the T previous time stamps. For this model (VAE_{surr}), only the encoder part is used in the final model to generate the latent variables, but the full model is analyzed in section IV-D to prove that all relevant features are represented in the latent space. The second model (VAE_{xy}) is trained only with longitudinal and lateral positions for T time stamps. The decoder part of this model is used to generate naturalistic trajectories in the full model. In addition, the encoder part generates latent variables of the previous trajectory of the vehicle, which are combined with the VAE_{surr} encoder ones to have a better latent description of the environment.

The loss function utilized, combines the Kullback–Leibler (KL) divergence loss D_{KL} , that measures how similar two probabilistic distributions are

$$D_{KL} = -\frac{1}{2} \cdot (1 + \log(\sigma^2) - \mu^2 - \sigma^2) \quad (2)$$

where μ and σ^2 are the predicted mean and variance, and the mean square error (MSE) between the reconstructed sequence \hat{y}_i and the ground truth y_i for T time steps

$$MSE = \frac{1}{T} \cdot \sum_{i=1}^T (y_i - \hat{y}_i)^2 \quad (3)$$

Both losses are combined in a weighted sum

$$loss = \beta \cdot D_{KL} + MSE \quad (4)$$

where β is a weight that can be treated as an hyperparameter tuned experimentally that increases the importance of a close reconstruction (low value) or a regular latent representation (high value) as shown in [36].

B. PREDICTION MODEL

The prediction model takes as an input the encoded situation of the driving scenario for the last T time steps ($L_{n,\mu}$ and $L_{n,\sigma}$) generated by the two encoders of the model, where n is the input latent feature, and predicts a latent representation of the future trajectory of the vehicle $M_{x,m}$ and $M_{y,m}$, where m is the output latent feature.

The first step of this model is to sample the input latent data. VAE encoders generate two values per dimension corresponding to a Gaussian distribution: mean ($L_{n,\mu}$) and variance ($L_{n,\sigma}$). The latent input variable is randomly sampled following that distribution both in the training and in the prediction stages. This sampling procedure allows to improve the robustness of the model compared to using only the mean value and discarding the variance and makes possible to estimate the trajectory mean and variance in the prediction stage using the Monte Carlo sampling method.

The sampled data are fed to a relatively simple model based only on Fully Connected Layers. Considering that the input data have been processed by the encoder and the output data will be processed by the decoder, the complexity of the model and layers does not need to be high.

C. FULL MODEL TRAINING

The full model is trained in two stages:

First, the VAE models are trained. VAE_{surr} model (in charge of encoding the information of the driving scene) is trained with surrounding vehicles data, whereas VAE_{xy} is trained with trajectories from the dataset. The aim of those models is to encode the input data into a lower dimensionality variable and to decode the predicted value into real trajectories. Thus, the input and the output training for each model are the same, using MSE error as a training loss of the models.

After that, the full model is ensembled to train the prediction model. To do this, the previously trained encoders from VAE_{surr} and VAE_{xy} , and the decoder from VAE_{xy} (whose weights are set as non trainable) are put together with the prediction model as shown on Fig. 2. The input data to the ensembled model are the current and past states $X_{m,t}$, where m can be either the feature index or x/y to denote longitudinal or lateral trajectory respectively and t is the time step. The loss

is computed using the MSE between the reference trajectories and the ones generated by the decoder ($Y_{m,t}$). This loss is backpropagated through the decoder to the prediction model in order to update its weights.

D. LATENT ANALYSIS

Number of latent variables is a key parameter that needs to be carefully chosen. That value needs to be sufficiently large to correctly represent all the input data, but also as low as possible so relevant characteristics are extracted from the data. To choose that parameter, a principal component analysis (PCA) is performed, using the LAPACK implementation of the singular value decomposition as shown in [37]. First, a sufficiently large number of dimensions is chosen to train the VAE. After that, the results of the PCA are analyzed, and using the cumulative sum of the explained variance ratio, the number of chosen dimensions is the lower that explains at least 95% of the data which, for the VAE_{surr} model, is 7, and for VAE_{xy} models is 2.

Furthermore, a deep study on the latent variables generated by VAE encoder is performed. This study aims to prove two assumptions: the latent space represents correctly all the input data, and similar driving situations are close together. For the first assumption mean absolute error (MAE) of the reconstructed data by the decoder is analyzed, giving a quantitative result of 0.7m for lateral trajectory and 0.04m for longitudinal trajectory. Furthermore, a visual inspection of several samples of the data, gives a good qualitative result as shown in Fig. 3, where surrounding reconstructed data are similar to original data and captures the evolution of dynamic data (Fig. 3(a) and 3(d)) and sharp distances changes due to other vehicles lane changes (Fig.s 3(b) and 3(c)), meaning that the VAE model gathers all the relevant data in the latent variables.

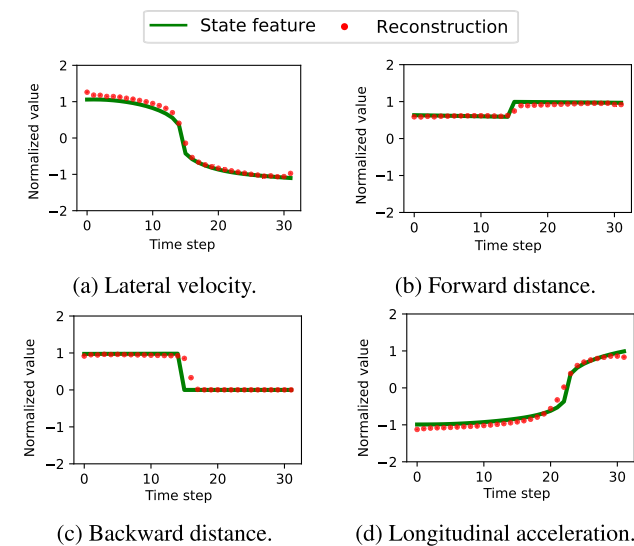


FIGURE 3. VAE_{surr} time reconstruction (red) of some state features (green). X axis represents the time-step, whereas Y axis is the normalized value of the variable.

For the second assumption, the latent variables of only the surrounding data are grouped according to the maneuver that is taking place or will take place soon in the input data. Fig. 4 compares the most representative 2D slices of the latent hyperspace for VAE and normal AE. Encoded maneuvers in latent space for VAE are clearly distinguishable and form different groups (Fig. 4(a)), whereas for normal AE, the different maneuvers are not distinguishable as their corresponding latent variables are placed in the same place (Fig. 4(b)). This analysis evidence that only using information of the surrounding vehicles, it is possible to predict the next maneuver, and VAE is able to distinguish and separate those situations.

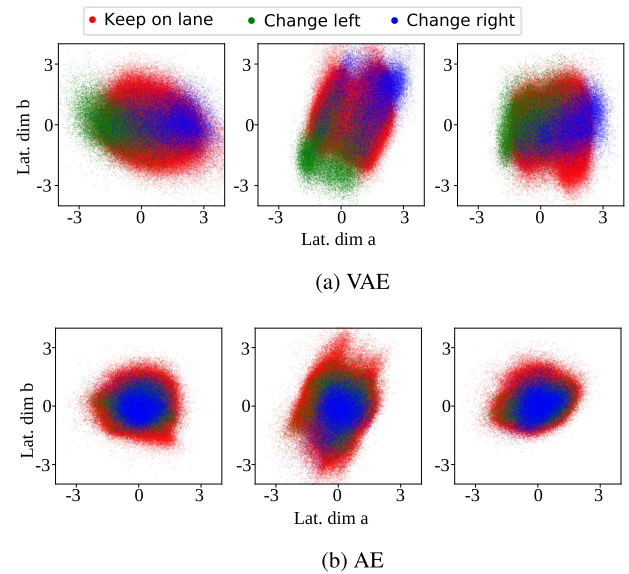


FIGURE 4. Most representative 2D slice of latent hyperspace for VAE and AE lat dim a and lat dim b. Colors denotes future maneuver: keep on lane (red), change left (green) and change right (blue).

In addition, Fig. 5 shows VAE decoder lateral and longitudinal trajectory reconstruction for different combinations of two latent space variables. VAE has learnt a smooth representation of all possible trajectories in the latent space. In this specific case, latent dimension 0 mostly represents the lateral road position, whereas latent dimension 1 denotes the maneuver or lateral displacement. This smooth representation of all the trajectories in the latent space is meant to improve the full model precision, as a small error in the future trajectory in the latent space, would result in a very similar trajectory.

Furthermore, the low dimensional representation of the state, where different maneuvers are grouped, makes easier for the following layers to forecast correct predictions, and reduces the complexity required for the prediction model.

V. RESULTS

The proposed model has been evaluated using sequences from the database to assess its accuracy. This section describes the process of obtaining the necessary metrics to determine the accuracy of the model, compares the

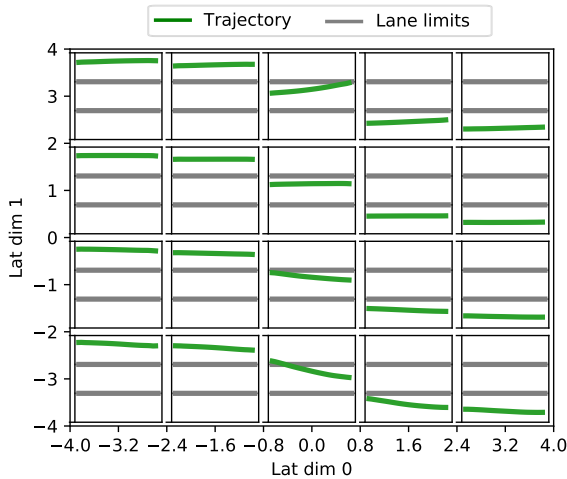


FIGURE 5. VAE decoder output trajectory reconstruction (green line) for different latent dimensions pairs (Lat dim 0, Lat dim 1). Lanes limits denoted by grey lines.

results with state-of-the-art models and evaluates the results obtained.

In addition, a second experiment is performed where the input data quality is degraded by adding noise. The results shows how a low quality input affects both the accuracy and covariance of the model

A. EVALUATION METRICS

Similar works on the topic of trajectory prediction evaluate the performance of their approach using either the root mean square error (RMSE) or the MAE separately for longitudinal and lateral prediction which gives a good intuition of the model performance. For this reason, these two metrics will be calculated to evaluate the proposed model and to compare it with similar works.

B. BASELINE MODELS

Table 1 compares the performance of the proposed method for different time prediction horizons with the following models:

- 1) 3D CNN-LSTM [19]: Interaction aware model based on a 3D spatio-temporal tensor that represents surroundings of the environment. It is compared to, and outperforms a simple LSTM model, and convolutional social pooling LSTM [3].
- 2) Multi-Head Att. [20]: A model based on a multi-head attention layer. It is compared to, and outperforms a linear model, a vanilla LSTM model and an encoder-decoder LSTM model. The original paper only performs tests up to 3s.
- 3) Simple AE: It is a baseline version of the proposed model where the VAE models are trained as a normal AE. This model is used to test whether the use of a VAE model provides the expected improvements over a simple AE.

To ensure a fair comparisons, the model is tested in the test subset of the highD dataset, the same one used in all the selected baseline methods.

C. RESULTS EVALUATION

The data results have been obtained by evaluating the predictions of the proposed model, which are compared with the real trajectories collected in the database.

The input of the model are different sequences that include all the pre-processed data that the model needs. This model have been evaluated and re-configured five times to generate predictions for different time horizons (from 1 second to 5 seconds). The obtained results are compared with the ground truth trajectories that the vehicle have followed in each sequence, computing different metrics that evaluates the model.

The proposed method captures better the environment situation and generates realistic trajectories, meaning that the main improvements are for long-in-time trajectories, whereas for short-in-time trajectories, results show non improvement as it is shown in Table 1.

TABLE 1. Longitudinal and lateral MAE and RMSE for compared methods.

| | Time Hor. (s) | 3D CNN-LSTM | Multi-Head Att. | Simple AE | Proposed method |
|-------|---------------|-------------|-----------------|-----------|------------------|
| | | MAE (m) | RMSE (m) | MAE (m) | MAE/RMSE (m) |
| Long. | 1 | 0.23 | 0.43 | 0.25 | 0.21/0.30 |
| | 2 | 0.59 | 0.47 | 0.5 | 0.37/0.52 |
| | 3 | 1.12 | 0.89 | 0.74 | 0.46/0.68 |
| | 4 | 1.81 | - | 1.07 | 0.70/0.98 |
| | 5 | 2.63 | - | 1.39 | 0.91/1.33 |
| Lat. | 1 | 0.07 | 0.04 | 0.23 | 0.06/0.09 |
| | 2 | 0.16 | 0.06 | 0.23 | 0.13/0.18 |
| | 3 | 0.25 | 0.11 | 0.24 | 0.14/0.20 |
| | 4 | 0.32 | - | 0.27 | 0.20/0.28 |
| | 5 | 0.39 | - | 0.30 | 0.24/0.36 |

Fig. 6(a) gives an example of trajectory forecasting for every vehicle in the scene. Predicted trajectory is close to the ground truth both in lateral and longitudinal axes. Some of the sampled trajectories for a vehicle can be far away from ground truth, but the average of all the samples, results in a close prediction.

One of the main advantages of the method is the possibility of estimating statistics of the variance of the generated trajectory. The second experiment performed evaluates qualitatively how the MSE and the variance of the trajectory evolves for inputs with different amount of noise in the input data. The results show that, for known situations (where little amount or no noise is added), the variance is low (Fig. 6(a)), but for very different ones (where the amount of noise added is higher), not only the prediction error is greater, but the variance increases as it is shown in Fig. 6(b).

In addition, the proposed method is able to run fast enough for real-time applications. Predictions and sampling can be performed in parallel, reducing computational time in contrast to many interaction models. Data of all vehicles can

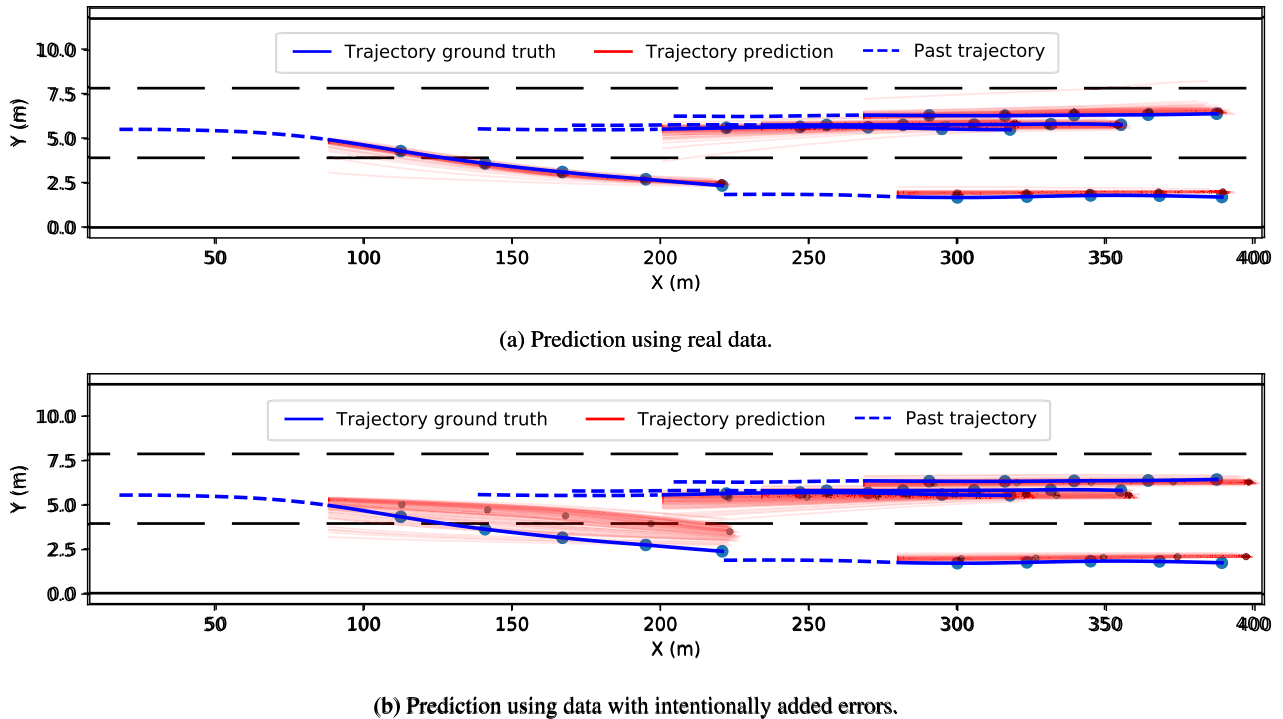


FIGURE 6. Proposed model trajectory prediction (red), ground truth past trajectory (dashed blue) and future ground truth (solid blue with dots separated 1s) for all vehicles in the scene. All the sampled trajectories are plotted with a low alpha value to represent better the confidence of the method and compare both sub-figures confidence. Mean predicted trajectory values are plotted in dark red dots separated 1s.

be concatenated and repeated multiple times to sample with Montecarlo method in parallel using a GPU.

VI. CONCLUSION

In this work, a novel trajectory and probabilistic prediction system, which makes use of the LSTM VAE network, has been evaluated.

The test performed using dataset proves the viability of the approach, and the results provided outperforms state of the art approaches of trajectory prediction. Furthermore, the proposal is able to provide trajectory uncertainty by means of the use of the LSTM VAE network.

Future works will focus in adapting the proposed architecture to different scenarios like city roads or intersections as the proposed method has been trained and tested only in highway roads and to include vehicle sensors raw data (images or LiDAR pointclouds) to have a better description of the environment.

The source code used in this work is publicly available at https://github.com/midemig/traj_pred_vae.

REFERENCES

- [1] X. Feng, Z. Cen, J. Hu, and Y. Zhang, "Vehicle trajectory prediction using intention-based conditional variational autoencoder," in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, Oct. 2019, pp. 3514–3519.
- [2] M. Schreier, V. Willert, and J. Adamy, "Bayesian, maneuver-based, long-term trajectory prediction and criticality assessment for driver assistance systems," in *Proc. 17th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2014, pp. 334–341.
- [3] N. Deo and M. M. Trivedi, "Convolutional social pooling for vehicle trajectory prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2018, pp. 1468–1476.
- [4] K. Messaoud, I. Yahiaoui, A. Verroust-Blondet, and F. Nashashibi, "Non-local social pooling for vehicle trajectory prediction," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2019, pp. 975–980.
- [5] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 27, 2014, pp. 1–9.
- [6] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," in *Proc. 2nd Int. Conf. Learn. Represent. (ICLR)*, Banff, AB, Canada, Apr. 2014, pp. 1–14.
- [7] R. Asaoka, H. Murata, M. Matsuura, Y. Fujino, M. Yanagisawa, and T. Yamashita, "Improving the structure–function relationship in glaucomatous visual fields by using a deep learning–based noise reduction approach," *Ophthalmol. Glaucoma*, vol. 3, no. 3, pp. 210–217, May 2020.
- [8] J. M. Graving and I. D. Couzin, "VAE-SNE: A deep generative model for simultaneous dimensionality reduction and clustering," *BioRxiv*, 2020, doi: 10.1101/2020.07.17.207993.
- [9] M. S. Mahmud, J. Z. Huang, and X. Fu, "Variational autoencoder-based dimensionality reduction for high-dimensional small-sample data classification," *Int. J. Comput. Intell. Appl.*, vol. 19, no. 1, Mar. 2020, Art. no. 2050002.
- [10] Z. Pan, J. Wang, W. Liao, H. Chen, D. Yuan, W. Zhu, X. Fang, and Z. Zhu, "Data-driven EV load profiles generation using a variational auto-encoder," *Energies*, vol. 12, no. 5, p. 849, Mar. 2019.
- [11] X. Chen, J. Xu, R. Zhou, W. Chen, J. Fang, and C. Liu, "TrajVAE: A variational AutoEncoder model for trajectory generation," *Neurocomputing*, vol. 428, pp. 332–339, Mar. 2021.
- [12] W. Ding, W. Wang, and D. Zhao, "A multi-vehicle trajectories generator to simulate vehicle-to-vehicle encountering scenarios," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 4255–4261.
- [13] S. Lefèvre, D. Vasquez, and C. Laugier, "A survey on motion prediction and risk assessment for intelligent vehicles," *ROBOMECH J.*, vol. 1, no. 1, pp. 1–14, 2014.

- [14] A. Polychronopoulos, M. Tsogas, A. J. Amrititis, and L. Andreone, "Sensor fusion for predicting vehicles' path for collision avoidance systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 3, pp. 549–562, Sep. 2007.
- [15] A. Houdenou, P. Bonnifait, V. Cherfaoui, and W. Yao, "Vehicle trajectory prediction based on motion model and maneuver recognition," in *Proc. IEEE/RSI Int. Conf. Intell. Robots Syst.*, Nov. 2013, pp. 4363–4369.
- [16] W. Yao, H. Zhao, P. Bonnifait, and H. Zha, "Lane change trajectory prediction by using recorded human driving data," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2013, pp. 430–436.
- [17] G. Agamennoni, J. I. Nieto, and E. M. Nebot, "A Bayesian approach for driving behavior inference," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2011, pp. 595–600.
- [18] G. Agamennoni, J. I. Nieto, and E. M. Nebot, "Estimation of multivehicle dynamics by considering contextual information," *IEEE Trans. Robot.*, vol. 28, no. 4, pp. 855–870, Aug. 2012.
- [19] M. Krüger, A. S. Novo, T. Nattermann, and T. Bertram, "Interaction-aware trajectory prediction based on a 3D spatio-temporal tensor representation using convolutional-recurrent neural networks," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Oct. 2020, pp. 1122–1127.
- [20] H. Kim, D. Kim, G. Kim, J. Cho, and K. Huh, "Multi-head attention based probabilistic vehicle trajectory prediction," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Oct. 2020, pp. 1720–1725.
- [21] K. Min, H. Kim, J. Park, D. Kim, and K. Huh, "Interaction aware trajectory prediction of surrounding vehicles with interaction network and deep ensemble," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Oct. 2020, pp. 1714–1719.
- [22] F. Alché and A. D. L. Fortelle, "An LSTM network for highway trajectory prediction," in *Proc. IEEE 20th Int. Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2017, pp. 353–359.
- [23] S. H. Park, B. Kim, C. M. Kang, C. C. Chung, and J. W. Choi, "Sequence-to-sequence prediction of vehicle trajectory via LSTM encoder-decoder architecture," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 1672–1678.
- [24] S. Dai, L. Li, and Z. Li, "Modeling vehicle interactions via modified LSTM models for trajectory prediction," *IEEE Access*, vol. 7, pp. 38287–38296, 2019.
- [25] G. Xie, A. Shangguan, R. Fei, W. Ji, W. Ma, and X. Hei, "Motion trajectory prediction based on a CNN-LSTM sequential model," *Sci. China Inf. Sci.*, vol. 63, no. 11, pp. 1–21, Nov. 2020.
- [26] R. W. Liu, J. Nie, S. Garg, Z. Xiong, Y. Zhang, and M. S. Hossain, "Data-driven trajectory quality improvement for promoting intelligent vessel traffic services in 6G-enabled maritime IoT systems," *IEEE Internet Things J.*, vol. 8, no. 7, pp. 5374–5385, Apr. 2021.
- [27] Y. Cai, L. Dai, H. Wang, L. Chen, Y. Li, M. A. Sotelo, and Z. Li, "Pedestrian motion trajectory prediction in intelligent driving from far shot first-person perspective video," *IEEE Trans. Intell. Transp. Syst.*, early access, Jan. 28, 2021, doi: 10.1109/TITS.2021.3052908.
- [28] C. Hegde, S. Dash, and P. Agarwal, "Vehicle trajectory prediction using GAN," in *Proc. 4th Int. Conf. I-SMAC (IoT Social, Mobile, Analytics Cloud) (I-SMAC)*, Oct. 2020, pp. 502–507.
- [29] L. Rossi, A. Ajmar, M. Paolanti, and R. Pierdicca, "Vehicle trajectory prediction and generation using LSTM models and GANs," *PLoS ONE*, vol. 16, no. 7, Jul. 2021, Art. no. e0253868.
- [30] L. Zhao, Y. Liu, A. Y. Al-Dubai, A. Y. Zomaya, G. Min, and A. Hawbani, "A novel generation-adversarial-network-based vehicle trajectory prediction method for intelligent vehicular networks," *IEEE Internet Things J.*, vol. 8, no. 3, pp. 2066–2077, Feb. 2021.
- [31] D. P. Kingma and M. Welling, "An introduction to variational autoencoders," 2019, *arXiv:1906.02691*.
- [32] V. Alexiadis, "Video-based vehicle trajectory data collection," in *Proc. Transp. Res. Board 86th Annu. Meeting*. Princeton, NJ, USA: Citeseer, 2006, pp. 1–18.
- [33] B. Coifman and L. Li, "A critical evaluation of the next generation simulation (NGSIM) vehicle trajectory dataset," *Transp. Res. B, Methodol.*, vol. 105, pp. 362–377, Apr. 2017.
- [34] R. Krajewski, J. Bock, L. Kloeker, and L. Eckstein, "The highD dataset: A drone dataset of naturalistic vehicle trajectories on German highways for validation of highly automated driving systems," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2018, pp. 2118–2125.
- [35] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A search space Odyssey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2222–2232, Oct. 2017.
- [36] C. Yan, S. Wang, J. Yang, T. Xu, and J. Huang, "Re-balancing variational autoencoder loss for molecule sequence generation," in *Proc. 11th ACM Int. Conf. Bioinf., Comput. Biol. Health Informat.*, Sep. 2020, pp. 1–7.
- [37] N. Halko, P.-G. Martinsson, and J. A. Tropp, "Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions," 2009, *arXiv:0909.4061*.



MIGUEL ÁNGEL DE MIGUEL (Graduate Student Member, IEEE) received the B.S. degree in electronics engineering and the M.S. degree in industrial engineering from the University Carlos III de Madrid, in 2015 and 2017, respectively, where he is currently pursuing the Ph.D. degree. He is also an Assistant Lecturer with the University Carlos III de Madrid. In 2013, he began his collaboration in several industrial research projects. His research interests include the areas of localization and path planning with a focus on applications for autonomous vehicles.



JOSÉ MARÍA ARMINGOL received the degree in automation and electronics engineering from the Universidad Politécnica de Madrid, Spain, in 1992, and the Ph.D. degree in industrial engineering from the Universidad Carlos III de Madrid, Spain, in 1997. He is currently a Professor in robotics and automation with the Intelligent Systems Laboratory, Department of Systems Engineering and Automation, Universidad Carlos III de Madrid. His research interests include computer vision, image processing, and real-time systems applied to intelligent transportation systems.



FERNANDO GARCÍA (Member, IEEE) received the Ph.D. degree in electrical, electronic and automatic engineering from the Universidad Carlos III de Madrid, in 2012. He works as an Associate Professor with the Universidad Carlos III de Madrid. His research interests include perception and data fusion, mainly applied to vehicles and robotics. He has been a member of the Board of Governors of the IEEE-ITS Society, since 2017. He was a recipient of IEEE ITS Young Researcher/Engineer Award 2021. He was the Chair of the Spanish Chapter, from 2019 to 2020.

• • •