2019 IEEE Intelligent Transportation Systems Conference (ITSC)
Auckland, NZ, October 27-30, 2019

Cyclist Intent Prediction using 3D LIDAR Sensors for Fully Automated Vehicles

K. Saleh, A. Abobakr, D. Nahavandi, J. Iskander, M. Attia, M. Hossny and S. Nahavandi
Institute for Intelligent Systems Research and Innovation (IISRI)
Deakin University, Australia
Email: k.aboufarw@deakin.edu.au

Abstract— **One of the main barriers against the full deployment of autonomous vehicles in urban traffic environments is the understanding of the intentions and behaviours of the human around them. Moreover, understanding and predicting intentions of vulnerable road users such as cyclists is still one of the most challenging tasks. In this work, we are proposing a novel framework for the task of intent prediction of cyclists via hand signalling from point cloud scans. We utilised our developed data generation pipeline for generating synthetic point cloud scans of cyclists doing a set of hand signals in urban traffic environments. Then, we feed a sequence of the generated point cloud scans to our framework which jointly segments all cyclists instances and predicts their most probable intended actions in an end-to-end fashion. Our proposed framework has achieved superior results with 83% in F_1-Measure score over the testing split of our generated dataset. Additionally, the proposed framework outperformed other compared baseline approaches with more than 39% improvement in F_1-Measure score.**

I. Introduction

Fully automated vehicles have gained significant interest from car manufacturers, research communities and governments worldwide. According to a recent report [1], by 2035 it is expected that more than 25% of newly manufactured vehicles will be fully autonomous. Fully autonomous vehicles (AVs) are promised to help reduce the number of road fatalities happening nowadays due to human drivers errors. That being said, AVs are still confronted by a number of challenges when it comes to understanding the behaviours and the intentions of the human around them, specially the vulnerable road users (VRUs) such as pedestrians and cyclists [2].

Over the past few years, several research studies have investigated techniques to tackle the intent prediction problem of pedestrians from the AVs perspective [3], [4]. On the other hand, the cyclists have not got the same attention despite the fact that they impose more serious problems on AVs according to recent multiple media reports [5]–[7]. Since the intention of VRUs is not an observable quantity, a number of observable cues have been studied in the literature as an indicator of the VRUs intentions. In case of pedestrians, example cues are the head and body pose [8] and the motion dynamics [9]. In case of cyclists, one of the strongest cues is the hand signalling [10]. However, in the literature of intelligent transportation systems, only the motion dynamics of the cyclists were investigated [11], [12]. One of the main reasons why is this the case with cyclists is that identifying their hand signals using vehicle-based sensors is a challenging perception task that needs a large amount of data which is almost non-existent publicly.

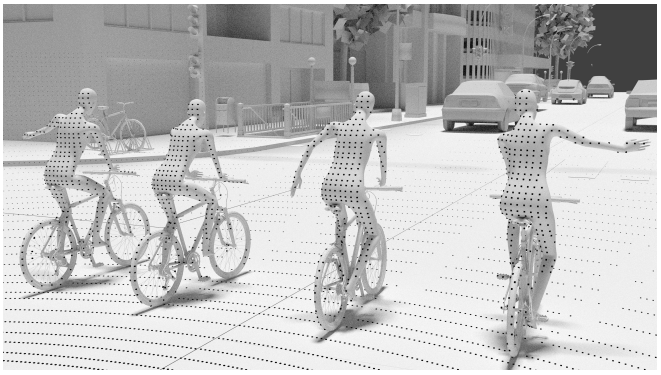


Fig. 1: The proposed method predicts common cyclist intended actions from hand signals via training on synthetic 3D LIDAR point cloud scans (overlaid black points on cyclists). Hand signals from left to right: Left turn (LTRN), no action (NACT), stop (STOP), and right turn (RTRN).

A potential solution for the data scarcity challenge is the utilisation of realistic-simulators [13], [14] to generate synthetic data of various scenes and traffic scenarios involving cyclists. Inside such realistic-simulators, not only a virtually unlimited amounts of data can be generated with different scenarios but also accurate annotations can be easily obtained. Moreover, different types of simulated sensors can be used for capturing and generating that data such as: RGB/depth cameras and 3D LIDAR sensors. In this work, we will utilise one of the famous realistic-simulation frameworks to generate sequences of 3D LIDAR scans of cyclists doing realistic hand signals in urban traffic scenes. Then, given these input 3D LIDAR scans, we will discriminate between three distinctive hand signals that cyclists commonly perform in urban traffic environments to communicate their intentions to other motorists, as shown in Fig. 1. The three hand signals correspond to the following intended actions: left turn, right turn and stop.

The rest of this paper will be organised as follows. In Section II a brief review on the work done in the literature related to the cyclist intent prediction problem will be

978-1-5386-7024-8/19/$31.00 ©2019 IEEE

presented. In Section III, our proposed methodology will be discussed in details. The experiments and its outcome results in comparison to other baseline approaches will be presented in Section IV. Finally, the summary and the future work will be discussed in Section V.

II. Related Work

A. Cyclist Intent and Action Prediction

In [11], Meijer et al. relied on the kinematics of bicycle to predict the intentions of cyclist in a controlled test environment. In return, they retrofitted a bicycle with instruments to measure its kinematics during manoeuvring actions done by volunteered cyclists at an intersection. The measured kinematics signals were as follow: wheel speed, roll angle, steering angle, velocity, acceleration, and peddling frequency. Given these measured signals, they developed a hidden Markov model to predict three intended actions of cyclists, namely going straight, turning right, and stopping. Since in real-traffic scenarios, it is hard to get such measured signals about kinematics of the cyclists, therefore this approach would be challenged when scaled to real-traffic scenarios.

Similarly, in [12], they relied on a vehicle-based stereo camera to record the past trajectory of cyclists for inferring their intended trajectory direction. They introduced two different motion dynamics models; one with a constant velocity Kalman filter (KF) for linear motion prediction. The other one consists of a mixtures of switching KF linear motion models. Each of the switching mixture models is utilised for a unique trajectory direction of cyclists, namely straight, 90°-right bend, 90°-left bend, 45°-right bend and 45°-left bend. Since in this work, we will be relying on 3D LIDAR scans as our input modality for predicting intended actions of cyclists, one of the related work is the recent work done by Benedek et al. in [15]. They were also relying on 3D LIDAR scans for the task of pedestrians' action recognition in surveillance scenarios. They utilised a number of feature descriptors on the point cloud data from a 3D LIDAR sensor to recognise certain activities such as bending and waving. Their proposed activity recognition model was based on a four-layer convolutional neural network (ConvNet) similar to the LeNet architecture [16].

B. Deep Learning for 3D Point Cloud Processing

Recently, deep learning-based methods have been achieving state-of-the-art results in many 2D computer vision tasks such as: image classification, semantic segmentation and object detection. Over the past couple of years, deep learning techniques have been also making some great strides in the 3D computer vision tasks specially the 3D point cloud data processing [17], [18]. One of the pioneer deep learning-based methods for point cloud processing is the PointNet architecture introduced in [17]. Unlike other deep learning methods that converted point cloud data to other intermediate representations, PointNet was the first architecture that worked directly on the raw unordered points of point cloud data. In PointNet, given a scattered and unordered point data, it can effectively learn features directly on point clouds in an end-to-end fashion. PointNet has also demonstrated robust capabilities in different 3D perception tasks such as 3D object classification, 3D part segmentation of objects, and semantic scene segmentation. PointNet relies on training an multi-layer perceptron (MLP) network with shared weights on each individual point from the point cloud data to learn invariant permutations of the point cloud data with the help of a max-pooling symmetric function. Additionally, PointNet also includes a transformation network that learns the transformation matrix of the points to be invariant to point cloud data rotations in the 3D space.

However, the main limitation for the PointNet architecture is that its feature learning network has to be trained and evaluated on the whole input point cloud which contains approximately 100K points. This results into high computational complexity [19]. Recently, this limitation has been addressed efficiently in the VoxelNet [19] architecture. VoxelNet is a generic 3D object detection network that combines feature extraction and 3D bounding box prediction into a single end-to-end trainable architecture. In VoxelNet, the point cloud is first, partitioned into voxels, then voxel features are extracted using a stack of trainable voxel feature encoding (VFE) layers. This feature learning sub-network learns discriminative voxel features via encoding inter-point interactions within a voxel.

TABLE I

Distribution (mean$\pm std$) of the anthropometric measures of the total 16 virtual 3D human manikins used in generating cyclist 3D LIDAR point cloud scans

3D Manikin	Height	Weight	BMI	BSA
Females	158.98±6.73	50.29±9.8	19.81±3.19	1.47±0.15
Males	173.06±7.16	70.9±13.09	23.58±3.59	1.81±0.18

The created models cover variations in gender and anthropometric measures to ensure generalisation to unseen body shapes. The height is reported in centimetres, weight in kilograms, BSA in meter square. Weight and body mass index (BMI) were estimated from body surface area (BSA) via Reading and Freeman's equation [20]–[22].

III. Proposed Methodology

In this section the details of our proposed methodology will be discussed. We will first describe the process we followed for the data-generation of the intended actions of cyclists from their hand signals. Then, we will introduce our novel framework that can simultaneously identify cyclists and their intended actions from a moving 3D LIDAR sensor in urban traffic environments.

A. Cyclist Intent Data Generation Pipeline

The simultaneous identification and recognition of cyclists and their intended actions from 3D LIDAR scans requires large amounts of point clouds with accurate annotations. The use of synthetic data has been successful in overcoming the scarcity of training data limitation and obtaining accurate learning models for several tasks such as pose estimation [23], fall detection [24], [25] and animal detection [26].

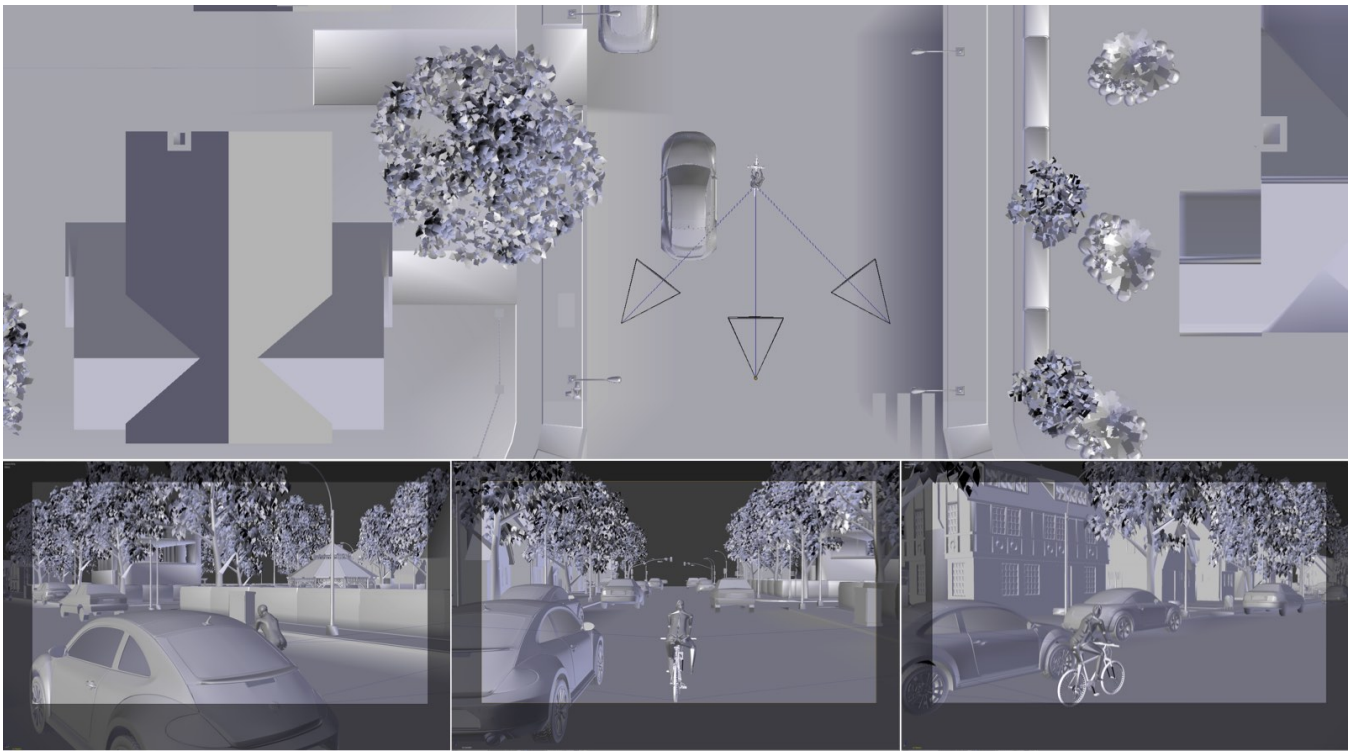


Fig. 2: Cyclist intent prediction data generation pipeline that we implemented in Blensor. We use recorded cycling mocap sequences to animate 3D manikins augmented into virtual scenes. We render the scene point cloud scans using three realistic virtual 3D LIDARs with different camera view angles.

Further, it has been demonstrated that deep learning models trained on synthetic depth images can generalise to real depth images [23], and depth images from a down-sampled and projected LIDAR scans [26]. In this work, we propose a data generation pipeline that integrates motion capture (mocap) technologies with realistic simulators to generate virtually infinite amounts of 3D LIDAR scans. Each generated point cloud scan has a corresponding accurate point-wise labels. The pipeline takes as input recorded mocap sequences of cyclist actions, and re-targets them to animate 3D virtual models of cyclists with different anthropometric measures in different scenes. Finally, a virtual 3D LIDAR point cloud scan is captured for the animated scan. The overall data generation process is shown in Fig. 2.

1) Motion Capture of Cyclist Actions: We used a marker-based mocap system to record cycling activities performed by real humans. The mocap system produces mocap sequences containing trajectories of the markers attached to the body. The mocap data collection was performed using an XSENS mocap system from 4 humans. The subjects were instructed to perform the most common hand signals for cyclists which are; left turn (LTRN), right turn (RTRN), stopping (STOP). We have also recorded sequences for continuous cycling as a no action (NACT) class. The resulting mocap dataset contains 16 mocap sequences for 4 signalling patterns performed by 4 humans.

2) Rendering Cyclist 3D LIDAR Point Clouds: The realistic open source simulator Blensor [13] is used to develop the scenes and render the cyclist signalling actions. Blensor features simulation capabilities using different sensing technologies such 3D LIDARs and depth cameras. We used the mocap sequences to animate virtual 3D human models, generated from MakeHuman software, and performed the rendering using a 3D LIDAR. In this work, we used 16 human manikins with different anthropometric measures as detailed in Table I. Thus, this approach can be replicated to any number of human models allowing generation of virtually infinite amounts of 3D LIDAR point cloud scans. The point clouds of the scene are rendered using three virtual 3D LIDARs following the cyclist with different view angles as shown in Fig. 2.

B. Unified Framework for Cyclists Intent Prediction

Given the generated 3D LIDAR scans from Section III-A, we need at the first stage to identify and localise any cyclists instances in the scene. At the second stage and based on the identified cyclists instances, a prediction about their intended actions need to be provided. Thus, we are proposing a unified framework for accomplishing these two stages jointly in order to predict the cyclists intentions via their hand signals. In the following, we will discuss the aforementioned two stages of our proposed unified framework (shown in Fig. 3).

1) Cyclist Instance Segmentation: In order to identify and localise the instances of cyclists in the generated point cloud data, we will utilise the PointNet architecture [17] for this sub-task of our unified framework. More specifically, we

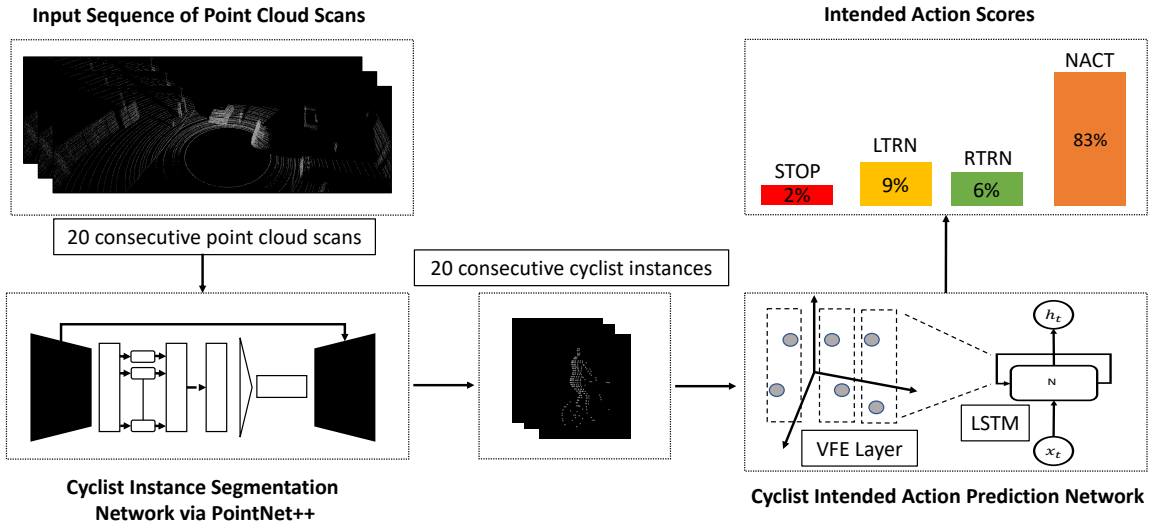


Fig. 3: Proposed unified framework for the cyclists intent prediction task. Given an input sequence of point cloud data from a 3D LIDAR sensor, the cyclist instances are first segmented. Then, they are passed to the intended action prediction network that gives a prediction about the probability scores of four distinctive classes of intended actions.

will be using the PointNet++ architecture [18] which is an efficient extension to the first version of PointNet architecture. PointNet has shown resilient results on a number of benchmarks for 3D semantic scene parsing. The main building blocks of the PointNet architecture are the feature embedding, feature transformation and the max-pooling aggregation blocks. In the feature embedding block, each point from the input point cloud data is embedded via an MLP with shared weights among these points. On the other hand, the feature transformation block is just a matrix multiplication operation to align the points in both the input point cloud space and in the embedding space in order to be robust to geometric transformations. This matrix multiplication operation is calculated between a predicted affine transformation matrix of the points and the original points from either the input point cloud or the embedded points from the feature embedding block. The affine transformation matrix itself is predicted via a small network they refer to it as the T-Net which is series of three convolution layers followed by a max polling layer and two fully connected layers. Then, all the transformed and embedded points are aggregated by a max-pooling layer to obtain a global feature vector of the input point cloud data. Finally, for the instance segmentation task which is our task of interest, the generated global feature vector after the max-pooling operation is concatenated with the local embedding from the feature embedding block for each point. By doing so, a per-point classification can be easily obtained which is then used to output i instances segmentation scores for each point in the input point cloud.

In our case, we have only two instances that each point can be, whether the point belongs to a cyclist instance or a background instance. As we mentioned earlier, we will be utilising the PointNet++ architecture. Unlike the aforementioned first version of PointNet, PointNet++ utilise the property of local neighbourhood of points in most input point cloud data to automatically sample and group local points into clusters. Each clustered groups are then passed through the main PointNet that was described above. These combined operations (sampling, grouping and PointNet) are encompassed into one module called the Set Abstraction (SA) module. In PointNet++, it was found that the repeating of consecutive SA modules leads to learning more hierarchical features with higher dimensional representations. After enough number of SA modules, in our cyclist instance segmentation task, comes an interpolation operation to reconstruct the down-sampled point cloud data back to its original dimensionality with the two predicted labels (background or cyclist).

One of the main advantages for choosing PointNet++ for our cyclist instance segmentation task is that it can effectively learn hierarchical features of points on different scales and the processing can be efficiently done due to the fact that there is no MLP training for each individual point. In order to further accelerate the performance of the original PointNet++ architecture, we adopted the implementation of the Open3D library [27], which can achieve real-time performance for instance segmentation task on point cloud data. The acceleration is accomplished via a GPU-based down-sampling of the dense point cloud data to sparse one during the training and GPU-based interpolation of the predicted sparse labels into the original dense ones during the inference.

2) Cyclist Intended Action Prediction: The proposed intended action prediction model Voxel-LSTM combines the feature learning capabilities of VoxelNet with a long short term memory (LSTM) temporal dynamics modelling module to identify the intended action, as shown in Fig. 3. The input to this stage is a sequence of point cloud scans of cyclist instances segmented using the cyclist instance segmentation step. The feature learning module implements a stack of voxel feature encoding (VFE) layers to learn discriminative

shape information from an input cyclist voxel. The VFE layer encodes the shape of the surface contained in the voxel through aggregating point-wise features extracted using a fully connected network (FCN). Given an input cyclist voxel V, we initially sample a fixed number of points N from V to reduce complexity and avoid potential voxel density imbalance. Then, we augment each voxel point with its relative offset to the local mean of the voxel. This transforms the voxel V into $V_f = \{p_i = [x_i,\ y_i,\ z_i,\ x_i - \bar{x},\ y_i - \bar{y},\ z_i - \bar{z}\ ]^T \in \mathbb{R}^6\}_{i=1}^{N}$, where XYZ are the 3D coordinates of voxel points, and $\overline{XYZ}$ are the local mean coordinates. Each point p_i is then fed the FCN to extract learned point-wise features $f_i \in \mathbb{R}^k$. The FCN encompasses a fully connected layer (FC), a batch normalisation layer (BN) followed by a rectified linear unit (ReLU) activation. The resulting point-wise feature representations are then locally aggregated using a 1D MaxPooling across all voxel features f_i. The locally aggregated features are concatenated with the point-wise features to encode inter-point interactions within the voxel. The temporal dynamics between subsequent cyclist voxel feature frames are modelled using a stack of long short term memory (LSTM) layers. Finally, the Voxel-LSTM model predicts the action using an output FC layer.

IV. Experiments

In this section we will firstly describe the procedure we followed for training the two-stages of our proposed unified framework for cyclist intent prediction. Then, the results of our experiments and the performance of our framework will be discussed.

A. Training Cyclist Instance Segmentation Network

As described earlier in Section III-B.1, we utilised the PointNet++ architecture for our cyclist instance segmentation task. Similar to the PointNet++ architecture, during the training phase we fed our cyclist instance segmentation network with fixed-size cropped boxes from the full generated point cloud scan. In our experiments, we fed the network with cropped boxes of size +/-30 meters (front/behind the 3D LIDAR) by +/-10 meters (right/left to the 3D LIDAR). In total, we trained our cyclist instance segmentation network with total 64 point cloud scans each sampled with the fixed-size boxes mentioned above which resulted in a total of 2400 training samples. We trained our network for 500 epochs using the Adam optimiser with a learning rate of 0.001 and batch size of 16 samples on a Nvidia Titan X GPU. In the original PointNet++, they found three consecutive modules of SA was enough to learn higher representations of the point cloud data. Thus, in our PointNet++-based instance cyclist segmentation network we used as well three modules of SA. In Fig. 4, a sample prediction from our cyclist instance segmentation network is shown. As it can be noticed, each point from the input point cloud scan is labelled either with blue "cyclist" or green "background" colours.

B. Training Cyclist Intent Action Prediction Network

The Voxel-LSTM model composes a feature learning network and temporal dynamics stage. We used a stack of two VFE layers; VFE-1(6, 32) and VFE-2(32, 128). This stack transforms randomly sampled $N = 280$ cyclist voxel frame points to a feature vector of length 128 features. The temporal dynamics modelling module has two stacked LSTM layers with hidden states of size 100 units. They process the input sequence of cyclist voxel features of length $T = 20$ and pass the activations to the final FC layer of 4 classes to predict the intended action.

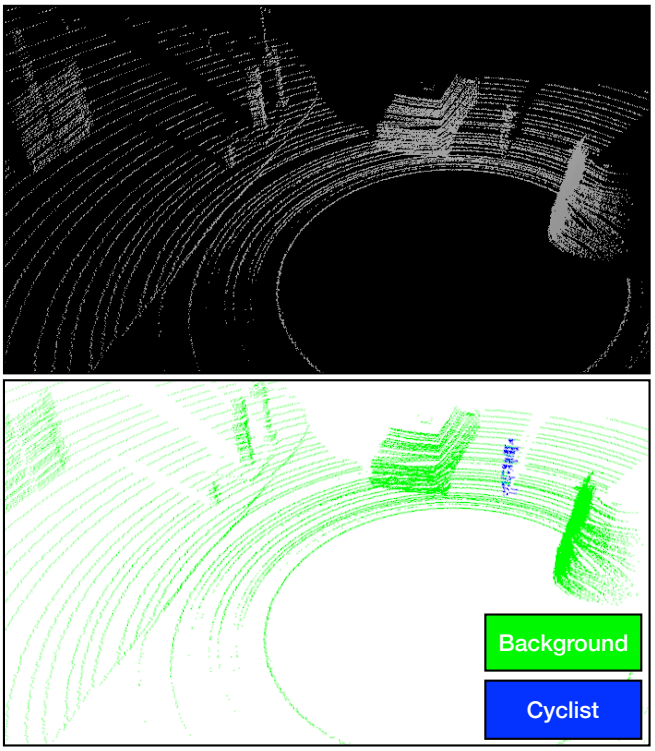
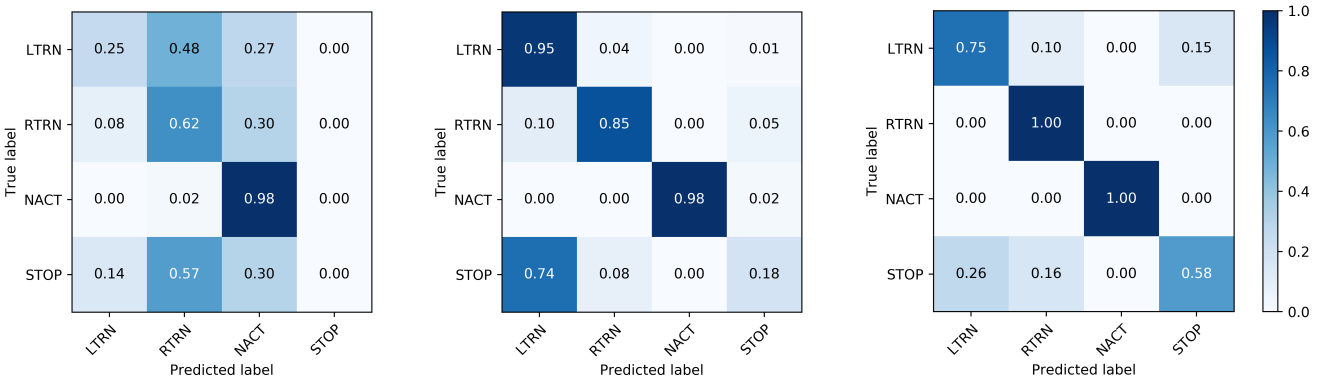


Fig. 4: Sample prediction from our cyclist instance segmentation network on an input point cloud scan (top image). In the predicted mask (down image), each point is labelled either with blue "cyclist" or green "background".

We generated a total of 1600 actions uniformly distributed among the 4 classes. Each action contains 25 synthetic 3D LIDAR point cloud scans covering the whole duration of the hand signal. The actions are depicted from 4 virtual scenes for 16 manikins that were animated using mocap sequences recorded from 4 real subjects. We segmented the cyclist from the generated point clouds using the PointNet++ model. Then, we randomly sampled 10 sequences of length $T = 20$ scans from each action to construct the training dataset. The dataset has also been filtered to remove the highly sparse cyclist voxels that contain less than 150 points. We split the dataset via leaving one scene and one subject for validation and the remaining data for training. This results into a training split of 6760 sequences and a validation split of 860 sequences. We trained the Voxel-LSTM model for 100 epochs using the Adam optimiser with an initial learning rate of 0.01, batch size of 16 sequences and weight decay of 0.0005, on a Nvidia Titan X GPU. The training objective function is minimising the cross entropy loss.



a) FPFH-RF b) Voxel-MLP c) Voxel-LSTM

Fig. 5: The performance of the proposed Voxel-LSTM model against other baselines.

C. Results and Discussion

In order to evaluate the performance of the proposed framework for the cyclist intent prediction task, the following evaluation metrics will be used. The first evaluation metric is the precision score, which measures how good is a given classifier at not-labelling a positive sample as a negative one. Accordingly, the precision is calculated using the formula; $tp/(tp+fp)$, where tp is the count of the predicted true positive samples, and fp represents the count of the predicted false positive samples. The second evaluation metric is the recall score, which measures the capability of a given classifier to identify the true positive samples. The recall is calculated according to the formula; $tp/(tp+fn)$, where fn is the count of predicted false positive samples. The last evaluation metric is the F_1-Measure score which is a weighted average between the precision and the recall scores and is calculated using the formula; $2\times(precision\times recall)/(precision+recall)$.

In Table II, we report the performance of our Voxel-LSTM model using the three aforementioned evaluation metrics. Additionally, we further compare the performance of our proposed model against other two baseline approaches. The first baseline approach, is an adapted version similar to the ones proposed in [28], [29]. In this approach, it relies on two separate stages, the first stage is hand-crafting a set of features from the input sequence of 3D point cloud scans, usually named feature descriptor stage. Then, the extracted features are passed to a classification stage. In our adapted implementation (we refer to this approach as FPFH-RF) we used the fast point feature histograms (FPFH) descriptor [30], which is one the most commonly used feature descriptors for 3D point cloud data and was used in [28], [29]. For the classification stage we used a random forest (RF) classifier [31]. The hyper-parameter we used in our experiments for the FPFH are 0.25 for the radius and 50 for the number of the nearest-neighbours points for its KDTree search algorithm. For the RF classifier, we used 50 for the number of trees in the forest and each tree has a maximum depth of 10.

The second baseline approach we compared our proposed model against is a similar one (we refer to it as Voxel-MLP) to our proposed model, however instead of using two LSTM layers, we used an MLP. MLP is simply a feed-forward neural network with multiple hidden layers and it has a non-linear activation function. In our experiments, the hyper-parameters we used for the Voxel-MLP model, is 2 for the number of hidden layers and each one has 100 neurons. Additionally, we used the ReLU as the activation function for the MLP network.

TABLE II

Comparison between our proposed framework for cyclist intent prediction and two different baselines approaches over the testing split of our generated dataset. Higher is better.

Model	Precision	Recall	F_1-Measure
FPFH-RF	0.41	0.52	0.43
Voxel-MLP (ours)	0.78	0.76	0.73
Voxel-LSTM (ours)	**0.83**	**0.83**	**0.82**

The proposed Voxel-LSTM model outperforms the FPFH-RF and Voxel-MLP baselines.

As it can be noticed from Table II, our proposed model Voxel-LSTM has outperformed the other compared approaches with F_1-Measure score of 0.82 and more than 0.39 improvement over the FPFH-RF approach which is commonly utilised for such tasks in the literature. This improvement demonstrates the advantages of the utilisation of the end-to-end learning technique of our proposed model Voxel-LSTM, where the need for hand-crafting set of features is not requires and as a result it leads to more robust classification capabilities in comparison to the FPFH-RF approach. Additionally, the benefits of using the LSTM architecture for modelling the temporal dependency between the learned features from the point cloud data is prevalent by the increased performance of the Voxel-LSTM in comparison to the Voxel-MLP which is missing this capability.

Additionally, in Fig. 5, we give more visualised analysis of the reported scores in Table II using confusion matrices. As it can be shown, our Voxel-LSTM model is showing

resilient results in identifying RTRN, NACT and LTRN intended actions in comparison to the other baselines. It is still however challenged when it comes to the STOP action, but this is justifiable since the stop action is pretty similar to the LTRN as it shown in Fig. 1. That being said, this confusion between the LTRN and STOP actions is still minimal in comparison to the other two baseline approaches.

V. Conclusion and Future Work

In this work we have proposed a novel framework for the task of cyclists intent prediction via hand signals from point cloud data. In order to overcome the scarcity of data for this task, we introduced a data generation pipeline for point cloud synthesis of traffic scenes involving cyclists making four distinctive hand signals. Given the generated point cloud scans, we firstly segment all the cyclists instances in the scans over a sequence of 20 scans using PointNet++. Then, we introduce another novel Voxel-LSTM model that simultaneously learns representative voxel features and the voxel-temporal dependency of the input sequence of cyclists instances to identify their intended action. Our proposed Voxel-LSTM model has demonstrated resilient results with 83% F_1-Measure score in comparison to other two baseline models on the testing split from the generated point cloud data. In our future work, we will focus on collecting real data from physical 3D LIDAR sensor in order to further evaluate the performance of our proposed framework.

References

[1] B. Philanthropies, "The aspen institute.(2017). taming the autonomous vehicle: A primer for cities."

[2] K. Saleh, M. Hossny, and S. Nahavandi, "Towards trusted autonomous vehicles from vulnerable road users perspective," in *Systems Conference (SysCon), 2017 Annual IEEE International*. IEEE, 2017, pp. 1–7.

[3] B. D. Ziebart, N. Ratliff, G. Gallagher, C. Mertz, K. Peterson, J. A. Bagnell, M. Hebert, A. K. Dey, and S. Srinivasa, "Planning-based prediction for pedestrians," *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2009*, pp. 3931–3936, 2009.

[4] K. Saleh, M. Hossny, and S. Nahavandi, "Intent prediction of vulnerable road users from motion trajectories using stacked lstm network," in *Intelligent Transportation Systems Conference (ITSC), 2017 IEEE International Conference on*. IEEE, 2017.

[5] P. Fairley, "The self-driving car's bicycle problem," *IEEE Spectrum*. [Online]. Available: https://spectrum.ieee.org/cars-that-think/transportation/self-driving/the-selfdriving-cars-bicycle-problem

[6] L. Laker, "Street wars 2035: can cyclists and driverless cars ever co-exist?" *The Guardian*. [Online]. Available: https://www.theguardian.com/cities/2017/jun/14/street-wars-2035-cyclists-driverless-cars-autonomous-vehicles

[7] S. Hanley, "Bicycles and autonomous cars are on a collision course," *CleanTechnica*. [Online]. Available: https://cleantechnica.com/2017/08/21/bicycles-autonomous-cars-collision-course/

[8] K. Saleh, M. Hossny, and S. Nahavandi, "Early intent prediction of vulnerable road users from visual attributes using multi-task learning network," in *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 2017, pp. 3367–3372.

[9] C. G. Keller and D. M. Gavrila, "Will the pedestrian cross? A study on pedestrian path prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 2, pp. 494–506, 2014.

[10] I. Walker, "Signals are informative but slow down responses when drivers meet bicyclists at road junctions," *Accident Analysis & Prevention*, vol. 37, no. 6, pp. 1074–1085, 2005.

[11] R. Meijer, S. de Hair, J. Elfring, and J. Paardekooper, "Predicting the intention of cyclists," in *6th Annual International Cycling Safety Conference, 21-22 September 2017, Davis, California*, 2017, pp. 1–3.

[12] E. A. Pool, J. F. Kooij, and D. M. Gavrila, "Using road topology to improve cyclist path prediction," in *Intelligent Vehicles Symposium (IV), 2017 IEEE*. IEEE, 2017, pp. 289–296.

[13] M. Gschwandtner, R. Kwitt, A. Uhl, and W. Pree, "Blensor: blender sensor simulation toolbox," in *International Symposium on Visual Computing*. Springer, 2011, pp. 199–208.

[14] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in *Proceedings of the 1st Annual Conference on Robot Learning*, 2017, pp. 1–16.

[15] C. Benedek, B. Gálai, B. Nagy, and Z. Jankó, "Lidar-based gait analysis and activity recognition in a 4d surveillance system," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 1, pp. 101–113, 2018.

[16] Y. LeCun, L. Jackel, L. Bottou, A. Brunot, C. Cortes, J. Denker, H. Drucker, I. Guyon, U. Muller, E. Sackinger *et al.*, "Comparison of learning algorithms for handwritten digit recognition," in *International conference on artificial neural networks*, vol. 60. Perth, Australia, 1995, pp. 53–60.

[17] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 652–660.

[18] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," in *Advances in Neural Information Processing Systems*, 2017, pp. 5099–5108.

[19] Y. Zhou and O. Tuzel, "Voxelnet: End-to-end learning for point cloud based 3d object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4490–4499.

[20] B. Reading and B. Freeman, "Simple formula for the surface area of the body and a simple model for anthropometry," *Clinical Anatomy*, vol. 18, pp. 136–130, 2005.

[21] D. Nahavandi, A. Abobakr, H. Haggag, and M. Hossny, "A low cost anthropometric body scanning system using depth cameras," in *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Oct 2018, pp. 3486–3491.

[22] D. Nahavandi, A. Abobakr, H. Haggag, M. Hossny, S. Nahavandi, and D. Filippidis, "A skeleton-free kinect system for body mass index assessment using deep neural networks," in *2017 IEEE International Systems Engineering Symposium (ISSE)*, Oct 2017, pp. 1–6.

[23] A. Abobakr, M. Hossny, and S. Nahavandi, "Body joints regression using deep convolutional neural networks," in *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 2016, pp. 003 281–003 287.

[24] A. Abobakr, M. Hossny, and S. Nahavandi, "A skeleton-free fall detection system from depth images using random decision forest," *IEEE Systems Journal*, vol. 12, no. 3, pp. 2994–3005, Sep. 2018.

[25] A. Abobakr, M. Hossny, H. Abdelkader, and S. Nahavandi, "Rgb-d fall detection via deep residual convolutional lstm networks," in *2018 Digital Image Computing: Techniques and Applications (DICTA)*, Dec 2018, pp. 1–7.

[26] K. Saleh, M. Hossny, and S. Nahavandi, "Kangaroo vehicle collision detection using deep semantic segmentation convolutional neural network," in *2016 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*. IEEE, 2016, pp. 1–7.

[27] Q.-Y. Zhou, J. Park, and V. Koltun, "Open3D: A modern library for 3D data processing," *arXiv:1801.09847*, 2018.

[28] R. B. Rusu, J. Bandouch, F. Meier, I. Essa, and M. Beetz, "Human action recognition using global point feature histograms and action shapes," *Advanced Robotics*, vol. 23, no. 14, pp. 1873–1908, 2009.

[29] M. Khokhlova, C. Migniot, and A. Dipanda, "3d point cloud descriptor for posture recognition." in *VISIGRAPP (5: VISAPP)*, 2018, pp. 161–168.

[30] R. B. Rusu, "Semantic 3d object maps for everyday manipulation in human living environments," Ph.D. dissertation, Computer Science department, Technische Universitaet Muenchen, Germany, October 2009.

[31] T. K. Ho, "Random decision forests," in *Proceedings of 3rd international conference on document analysis and recognition*, vol. 1. IEEE, 1995, pp. 278–282.