Robotics and Autonomous Systems 127 (2020) 103477

Contents lists available at ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot

DeepGoal: Learning to drive with driving intention from human control demonstration

Huifang Ma [a], Yue Wang [a,*], Rong Xiong [a,*], Sarath Kodagoda [b], Li Tang [a]

[a] State Key Laboratory of Industrial Control Technology and Institute of Cyber-Systems and Control, Zhejiang University, 38 Zheda Road, Zhejiang 310027, China

[b] Centre for automotive Systems, The University of Technology, Sydney, Australia

ARTICLE INFO

Article history:
Available online 25 February 2020

ABSTRACT

Recent research on automotive driving has developed an efficient end-to-end learning mode that directly maps visual input to control commands. However, it models distinct driving variations in a single network, which increases learning complexity and is less adaptive for modular integration. In this paper, we re-investigate human's driving style and propose to learn an intermediate driving intention region to relax the difficulties in end-to-end approach. The intention region follows both road structure in image and direction towards goal in public route planner, which addresses visual variations only and figures out where to go without conventional precise localization. Then the learned visual intention is projected on vehicle local coordinate and fused with reliable obstacle perception to render a navigation score map that is widely used for motion planning. The core of the proposed system is a weakly-supervised cGAN-LSTM model trained to learn driving intention from human demonstration. The adversarial loss learns from limited demonstration data with one local planned route and enables reasoning of multi-modal behaviors with diverse routes while testing. Comprehensive experiments are conducted with real-world datasets. Results indicate the proposed paradigm can produce more consistent motion commands with human demonstration and shows better reliability and robustness to environment change. Our code is available at https://github.com/HuifangZJU/visual-navigation.

1. Introduction

In recent automotive driving research, deep learning tries a revolutionizing way for vehicle control, which directly maps raw pixels from camera image to steering commands in an end-to-end manner [1–3]. End-to-end(end2end) approach seeks to avoid steps of building an explicit environment model in conventional approach which includes mapping, localization, route planning and motion planning, etc. Instead, it optimizes driving variations for perception, planning and reasoning in a single network to maximize overall control performance [1]. In contrast to conventional approach, data for training end2end networks can be collected with relative ease way, i.e., driving around and recording human demonstration control.

However, end2end approach faces with the problem of learning very complex mapping in a single network, which needs intensive supervision to handle huge driving variations. Moreover, it prevents intermediate fusion of visual information with other range finder sensors that help much to avoid obstacles. This raises security concerns, as a small error in vision can induce severe consequence for driving in high frequency control loop. Despite these drawbacks, current research often focuses on end2end setting because it allows to look into plenty of challenges. Some recent methods incorporate additional route planner as learning input [4–6], such as a routed map or a directional instruction, as shown in the top half part of Fig. 1. The planned route captures longer-term motion rules and helps to choose a correct direction upon reaching a fork. It is beneficial and brings performance promotion. Yet the network still lacks transparency of how the planner acts on various driving variations.

In order to address the challenges in end2end approach, we focus on learning an explainable representation following the manner of how human drives with route planner. Humans may rely on the planned route in public softwares to figure a direction towards goal, then use visual cues like road semantics to reason where to drive. With the goal-directed area in mind, they perform flexible vehicle control in relating to different driving scenarios. The specific control rules may change, e.g., to follow a lane in urban road nets or to mind unexpected obstacles in campuses. However, a goal-directed visual region is always formed based on local road situation, which keeps an overall sense of driving direction for vehicle control. We denote this region as driving intention.

* Corresponding authors.
E-mail addresses: wangyue@iipc.zju.edu.cn (Y. Wang), rxiong@zju.edu.cn (R. Xiong).

https://doi.org/10.1016/j.robot.2020.103477
0921-8890/

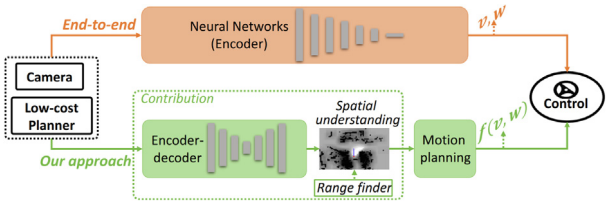


Fig. 1. Comparison of proposed paradigm with end-to-end model for automotive driving.

We think this longer-term driving intention region is effective and informative to improve end2end approach, which addresses only visual variations and solves where to go without conventional precise localization. Therefore, we define the pipeline as shown in the bottom half part of Fig. 1. An encoder–decoder structure is adopted to learn the aforementioned driving intention from image perception and a route planner. For the route planner, we follow the work in [5] and resort to the planned route in public navigation softwares with GPS localization. The driving intention region is then projected onto a local navigation score map with range finder data fused to increase reliability. Such a navigation score map encoded with goal-directed information can be directly used for next motion generation, which is also able to explicitly consider more specified motion variations.

To avoid manual definition and annotation of driving intention on image, we devote to learning from human demonstration. Specifically, human control a vehicle to move and follow a planned route towards goal. Then for each image perception, its traversed area in the near ground satisfies current driving intention and can be projected on image plane as supervision. The challenge lies in that the demonstration driving only covers a single direction for each fork and cross, while routes planned to different directions can be provided during test. The driving intention is valid as long as it holds rationality in regards to both visual observation and local route plan. Thus, we consider the learning task is not pixel-level imitation but structural reasoning, and develop a weakly-supervised model of cGAN-LSTM network with the adoption of adversarial loss function. The network learns to generate a 'fake' driving intention region which is hard to be distinguished from the 'real' region maneuvered by human. Then the generated result is punished as a whole to implicitly learn road semantics, planning intention as well as their correlation. Besides, time continuity is considered with a LSTM unit for performance enhancement. The outline of our method is provided in Fig. 2.

In the experiment, a straightforward motion generation method is implemented in a DWA (Dynamic Window Approach) [7] manner for comparison with end2end approach. The proposed pipeline is validated through real-world datasets including previously unseen scenarios to demonstrate generalization ability. Experiments show our method achieves better performance than state-of-the-art end2end model on both reliability and robustness. To summarize, our main contributions are twofold:

- An innovate learning-based automotive driving system is developed. The system learns from images and low-cost GPS-level route planner to achieve goal-directed driving intention without precise localization. It eases problem complexity for end2end approach and can be efficiently integrated for modular motion planning.
- A weakly supervised and adversarial learning method is developed through learning from demonstration, the core of which is a cGAN-LSTM network trained with limited single-modal demonstration data. The model is enhanced with time continuity and can be generalized to achieve multi-modal behaviors when facing new scenarios.

The remainder of the paper is organized as follows: Section 2 reviews the related works on learning-based approaches for automotive driving. Section 3 illustrates details of the proposed system architecture. Section 4 presents the experimental results, and Section 5 draws a conclusion.

2. Related work

Conventional pipeline of vehicle control includes mapping, localization, path planning and motion planning. The result of localization, i.e. a reference path or a goal in vehicle local coordinate, and the result of path planning, i.e. a grid map with(out) semantics, are fed into motion planning to generate final control command. The system is thus sensitive to environment change and calls for a lot of work to improve performance on separate modules. Specifically for the visual perception, modular approaches can need substantial pixel-level or box-level annotated data for CNN based recognition tasks. And there are multiple sub-components for recognizing driving-relevant objects, such as vehicles, pedestrians and cyclists detection [8–13].

In this section, we give a brief review of two learning-based systems aimed at improving the conventional automotive driving, each following a different system design: end2end approaches and direct perception approaches.

End-to-end approaches. Recently, end2end method learned from human demonstration becomes popular in automotive driving. The intrinsic merit is that the performance of intermediate stages in conventional system architecture may not be aligned with the ultimate goal, namely, the control of the vehicle. With this idea, [1] firstly proved powerful ability of CNN to steer a vehicle directly from vision input. Codevilla, et al. [2] then proposed to learn the driving model to compute motion command via conditional imitation learning, which incorporates high-level command input to consider the repeatability of imitation learning. The work in [4] collected control commands from existing local planner (Dynamic Window Approach [7]) and proposed a two-stage approach to relax prior knowledge for localization. This relies on the path-planning result, as the form of navigation is to learn expected motion commands using a residual neural network. The work in [5] adopted 360-degree surround-view cameras along with planned routes information from commercial maps to learn an end2end driving model. Their work has utilized GPS signals as well as public map to generate steering angles and speeds based on a RNN. As reported in their evaluation, it has unavoidably incorporates human intervention. The work in [14] proposed to estimate a variational network to get a full probability distribution over the possible control commands; however, when combined with specific navigation indicators, they still solve an accurate form of certain control command. Compared with this category, our work relies on similar input without relying on precise geometric transformations while achieves intermediate representation of navigation score map in robot local coordinate. We consider the problem space of end2end control learning is more complicated than ours, as the motion states of vehicle are coupled with the visual understanding.

Direct perception approaches. Another idea still achieves an intermediate representation of environment while goes a step further towards vehicle direct usage. The concept was first proposed in [15] which involves multiple distance regression tasks to surrounding cars and road markings. Al-Qizwini et al. [16] then improved the work in [15] by analyzing different CNNs for the mapping from image to indicators. In [17], a model utilizing public Google Street View and the OpenStreetMap is developed to infer road layout and vehicle relative pose given imagery from onboard cameras. Later, the work in [6] generalized the work in [15]

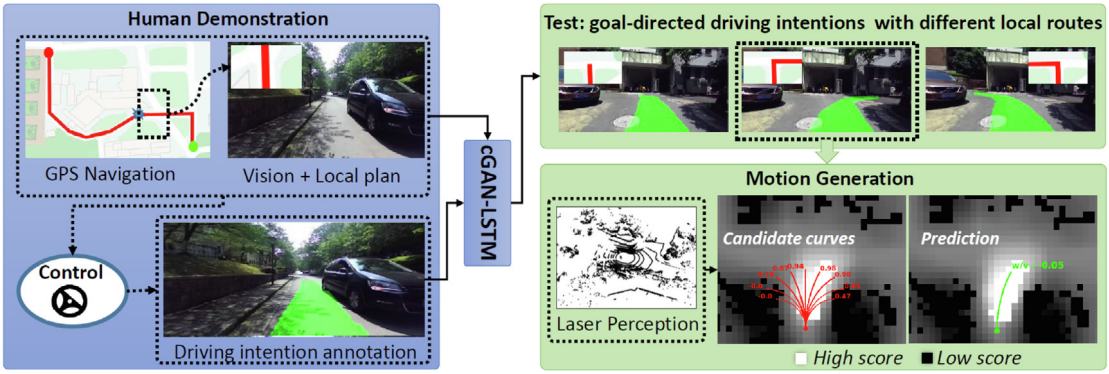


Fig. 2. Outline of our approach. A cGAN-LSTM model is utilized to learn driving intention from human demonstration with local route plan. While testing, the model generates corresponding driving intentions following both planning intentions and road structures. The driving intention is then integrated with concurrently collected laser data and rendered into a navigation score map. Based on this, vehicle motion is generated by scoring candidate driving curves. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

and proposed intermediate affordances to facilitate driving, including both vehicle relative pose to the road and recognition of specific traffic signs. Apart from these newly proposed concepts, the works of traversable region recognition [18–20] also belong to this category. The weakly-supervised drivable area segmentation method proposed in [18] assumes the traversal areas are exactly where vehicle has visited earlier and can be projected to images as annotation. Then Tang et al. [19] applies mapping techniques to extend the trajectory projection from one to many, which broadens the traversable area. Wang et al. [20] proposed an online learning mechanism to deal with the appearance change of traversable region without referring to the massive data. These works have eventually yielded an intermediate representation for vehicle control, while may still need further reasoning for motion control. Furthermore, some of the learned perception results still rely on accurate pose estimation from conventional approaches.

3. Methods

This section introduces the proposed driving paradigm in detail. The system architecture is shown in Fig. 2, blue box shows the procedure of learning from human demonstration and green boxes show the test application. The core of the system is a cGAN-LSTM network, which takes front-view image and local route plan as input. The model learns to generate a goal-directed driving intention region on the road area to indicate supposed future control. When testing in strange scenarios with different local route plans, the model is able to generate corresponding driving intentions towards different directions. For further motion generation, the driving intention is projected on vehicle near ground and integrated with concurrent laser perception to render a navigation score map. We implemented a straightforward method to generate control command by scoring candidate driving curves to test our framework. The specific illustration of each step is provided in the following parts.

3.1. Weakly-supervised driving intention learning

3.1.1. Network design

The intention learning is framed as a structure reasoning process to follow both road situation in image and planning intention in local route plan. Since human demonstration only covers a single direction for each fork and cross while route plans to different goals can be provided during test, the learning is not treated as a pixel-level imitation and regression. Instead, the recent GAN [21] model is adopted with adversarial loss to evaluate the network output as a whole. GAN consists of a generator and a discriminator. The fake output from generator is trained to be similar with the real data so as to cheat the discriminator. Thus, the generator eventually learns to produce an overall reasonable result following the distribution in provided dataset. We implemented a network structure of cGAN-LSTM specifically for driving intention generation, the model structure is provided in Fig. 3.

The network structure has referred to the work in [22], which follows the design of conditional GAN [23] and utilizes a UNet [24] as generator. As implied, UNet is an encoder–decoder structure with skip connections to preserve lower-stage features. We use less layers in our case since the generation task does not need to recover complete textures of the image. Our model treats both the image and the route plan as prior conditions. The two inputs together with the generated driving intention are fed into the discriminator for evaluation. Since the intentions are continuous in both time and space domain during driving, a LSTM (Long Short-term Memory) unit is inserted after the last encoder layer to capture series relation, which at the same time guarantees minimum parameter increase.

Let us consider $k-1$ steps of the former visual perception, the sequential input images are denoted as $I_{[t-k+1,t]}$ and the corresponding local route plans are denoted as $R_{[t-k+1,t]}$, for each time t. A visual driving intention towards goal is expected to learn, denoted as V_t, at current image. Thus, the problem is formulated as $G : \{I_{[t-k+1,t]}, R_{[t-k+1,t]}\} \rightarrow V_t$. Since previous approaches have found it beneficial to mix the GAN objective with a more traditional loss, such as L1 distance [22], the objective function is the sum of two weighted loss for the considered k steps:

$$L = \arg\min_G \max_D \sum_t^{t-k+1} L_{cGAN}(G, D) + \lambda L_{L1}(G) \tag{1}$$

where λ is the weight parameter.

The first item is the standard cGAN objective function:

$$\begin{aligned} L_{cGAN}(G, D) =& E_{I_t, R_t, V_t}[logD(I_t, R_t, V_t)] + \\ & E_{I_t, R_t}[log(1 - D(I_t, R_t, G(I_t, R_t)))] \end{aligned} \tag{2}$$

and the second item is a patch-wise $L1$ distance from generated intention to the provided real driving intention:

$$L_{L1}(G) = E_{I_t, R_t, V_t}[\|V_t - G(I_t, R_t)\|_1] \tag{3}$$

The cGAN-LSTM model ensures generated intention to consider both current road structure in image perception and different intentions in local planned routes. It has implicitly learned their inherent correlation by adversarial training and can be generalized to allow for different driving intentions when confronting with new scenarios.

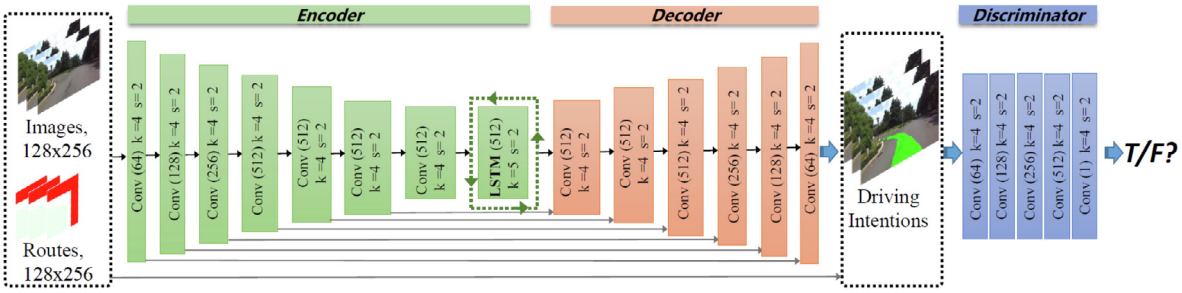


Fig. 3. Model architecture of cGAN-LSTM. Front-view images combined with local route plans are fed into a UNet structure to generate goal-directed driving intentions. The middle of the UNet is inserted with a LSTM-unit to incorporate time continuity. The predictions are then concatenated with input images to go through the discriminator.

3.1.2. Data preparation

To achieve the learning of driving intention, the data of image perception, local route plan, and annotation of driving intention region needs to be provided. More importantly, their correlation needs to be specifically established. This part illustrates how the data are made ready for training network and performing comprehensive experimental evaluations.

Local route plan We devote to follow human's manner to get local route plan, which uses public navigation softwares with GPS signal. For the correlation, one possibility is to enable the communication between vehicle to such an APP during demonstration. Nevertheless, the interface to real-time synchronized view of local route plan is not make public for research usage. Besides, the model performances under different GPS localization errors need to be carefully considered and experimentally evaluated. This may lead to substantial workload for on-line data collection and rendering. Therefore, we developed an off-line route rendering method, which makes use of the spatial alignment between public map and vehicle demonstrated trajectory. The procedure is shown in Fig. 4.

Fig. 4. Procedure to get off-line local route plans. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Given public map data from Baidu Map,[1] the global route R is annotated with a similar manner to that of the navigation softwares, as shown with the red lines in Fig. 4(a). Then, R is discretized to route points R_d, as shown in Fig. 4(b). After human demonstration driving along the global route, the vehicle poses can be obtained as shown with blue dots in Fig. 4(c), denoted as T_r. In the experiment, the vehicle poses are obtained with the conventional localization approach demonstrated in our previous work [19].

The spatial alignment from vehicle poses to the routed public map is now the task of aligning two sets of planar points R_d and T_r. Here, the DTW (dynamic time warping) [25] algorithm is used, which is commonly adopted in the time domain to warping time series data:

$$DTW(T_r, R_d) = min\ \frac{1}{K}\sqrt{\sum_{k=1}^{K} w_k} \tag{4}$$

where K is the warping length, and $w_k = (i, j)_k$ is the warping weight between T_r to R_d.

[1] https://map.baidu.com/.

The step-by-step optimization objective is:

$$\gamma(i, j) = w(T_r(i), R_d(j)) + min\{\gamma(i-1, j-1), \gamma(i-1, j), \gamma(i, j-1)\} \tag{5}$$

where γ is the accumulated series distance. Specifically, a geometric warping criterion is adopted in our scenario, and w is defined as the euclidean distance from projected vehicle pose to the center route point:

$$w(T_r(i), R_d(j)) = \|(T_r(i)), R_d(j)\|_2 \tag{6}$$

The aligned result is shown in Fig. 4(d). By assigning the heading direction of each road section, different local route plans can be cropped under various experiment settings.

Driving intention annotation. The driving intention region is annotated by projecting vehicle traversed area on current image under specified route plan, as shown in Fig. 5.

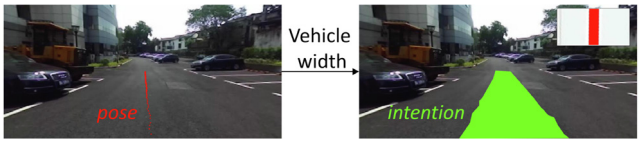


Fig. 5. Driving intention annotation. Left: vehicle projected poses; Right: annotated driving intention region.

For each image, vehicle future poses in the near ground are first projected on the image plane. Then the poses are dilated with vehicle width to indicate current driving intention. Thus the annotation certainly satisfies both planning intention in the route plan and road semantics in image. This idea is similar to the drivable region annotation work in [18]. However, they do not differentiate region directions and further do not learn their relations with the route plans. The other regions on the image are then labeled as *obstacle* and *unknown* [18] utilizing the projection of concurrently collected laser perception.

3.1.3. Training

We consider four time steps to train the cGAN-LSTM network: 0.9 s in the past, 0.6 s in the past, 0.3 s in the past, and the current frame, similar to the experiment setting in [5]. As the straight road sections are much longer than the turning sections, perceptions for straight driving class are down-sampled to around one sixth in order to keep a same quantity with that of turning perceptions.

For parameter optimization, we first train a basic-model of cGAN without LSTM unit, following the common procedure of one gradient descent on D and then one step on G. The basic model network is trained with stochastic gradient descent(SGD) at a learning rate of 0.0002 and a batch size of 12. The momentum parameters are $\beta_1 = 0.5$ and $\beta_2 = 0.999$. The basic model is

trained with 200 epochs. Then the cGAN-LSTM model is fine-tuned based on the pre-trained parameters of the basic model. During the fine-tuning, the encoder part of UNet is fixed to keep a stability of the network. The cGAN-LSTM model is trained additionally around 20 epochs. At inference time, the generator net runs in exactly the same manner as during the training phase.

3.2. Motion generation with driving intention

The driving intention preserves a learned region which is highly adaptive to fuse with other sensors and motion variations to ensure vehicle safety. Considering vehicle usually runs on smooth roads, an assumption is made that the road area in the near front of vehicle can be modeled with a flat plane. Thus, driving intention area can be projected on robot local coordinate with camera calibration parameters and then integrated with concurrent laser perception to render a navigation score map. Based on the local navigation score map, a straightforward motion generation method is implemented by scoring candidate driving curves. The procedure is shown in Fig. 6.

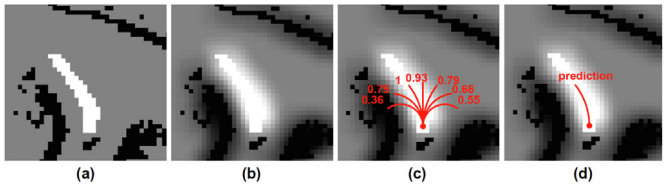


Fig. 6. Motion generation with learned driving intention. (a) Projection of visual driving intention with laser perception integrated; (b) Navigation score map modeled with Gaussian kernel; (c) Candidate driving curves with their scores labeled; (d) Final control command with the highest score.

In Fig. 6(a), the white grids (0.5 m $\times$ 0.5 m) indicate projected driving intention in vehicle local ground. The black grids indicate the obstacle perception from concurrent laser data. To consider the neighboring influence, driving intention grids are assigned with positive Gaussian kernels and obstacle grids are assigned with negative Gaussian kernels, which together form the navigation score map used for motion generation, as shown in Fig. 6(b).

In order to keep the task tractable, motion generate is conducted in an DWA manner, i.e., to produce candidate driving commands and estimate the best one. Following the work in [1, 14], the steering command is presented as driving curvature, denoted as $\frac{1}{r}$, where r is the turning radius in meters. Since vehicle constantly adjusts its control command based on visual perception, it is reasonable to assume that it keeps a uniform motion during a short time clip. Resultantly, the future trajectory can be modeled with a curve in the local navigation score map.

Specifically, the 90° space ahead of vehicle is divided into numbers of directional sections in relating to different requirements of control precision. Then, a same number of candidate driving curves are generated whose curvatures range evenly within $[-0.2, 0.2]$, as shown in Fig. 6(c). It is an example of seven candidate curves. Each curve can be estimated with a score based on the navigation score map. Then the final command is the curve with the highest score, indicating the direction towards goal, as implied in Fig. 6(d). The motion generation is straightforward while efficient to show its usage and be compared with end2end approach. The incorporation of more specified motion variations is the future work.

4. Experiments

This section reports the experiment results of the proposed approach, including experiment set up, performance of driving intention generation, and performance of motion generation.

4.1. Data sets

Experiment data are collected with a real vehicle running in our campus, which have been extensively adopted in conventional automotive driving research [26–28]. The data collection routes are shown in the left of Fig. 7. Blue lines show the training route with a length of 1.2 km. Red lines show the test route with a total length of 4.8 km. The overlap sections of the two routes are basically collected in a bi-directional manner. The vehicle used for data collection is shown in the right side of Fig. 7, which is a four-wheeled mobile vehicle equipped with a ZED stereo Camera, a Velodyne VLP-16 laser scanner and a D-GPS. Only images from the left camera of ZED are used with a resolution of 314 $\times$ 648 pixels. The training data involve 21 times of demonstration driving at different times over three days, covering varying weather/illumination conditions. Each demonstration driving contains ~7000 frames of observations. And the test data contain ~25 000 frames of observations.

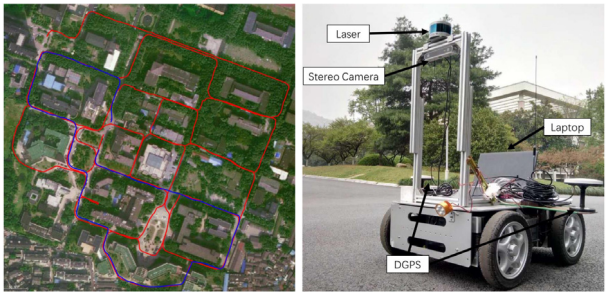


Fig. 7. Data collection routes and experiment vehicle. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

4.2. Intention generation result

The first part presents the result of driving intention generation.

Evaluation Metrics. There are three criteria used for evaluation on intention generation: IoU, Δyaw, and $E(l_2)$. IoU is the intersection over union between predicted driving intention and human demonstration. It is a widely used metric for visual area prediction tasks, which shows both the correlation and the relative scale to ground truth. Δyaw is the angle difference of driving directions between prediction and human demonstration in image plane. Since the presentation of driving intention region is a novel contribution in our work, this metric is first proposed in this work to evaluate the direction difference. The calculation illustration is provided in Fig. 8.

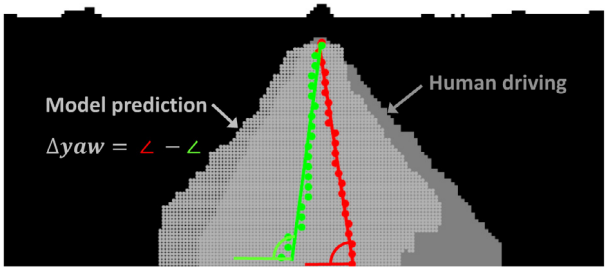


Fig. 8. Illustration of Δyaw. The dark gray grids show the future control of human demonstration and the light gray grids imply model prediction. Red dots and green dots show the center points of human driving and model prediction respectively, which have been both down-sampled for better visualization. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 1
Performance of driving intention generation.

Model	IoU%	Δyaw	$E(l_2)$
cgan_basic	62.01	13.63	0.557
cgan_lstm	78.47	8.46	0.171

$E(l_2)$[29] is the averaged l_2 distance from prediction to ground truth and has been commonly used for long-narrow shape prediction tasks [29–31]. Specifically in our case, the pixel-level center points of driving intention and human demonstration are projected in the ground plane. Then for each pair of points that have the same row in the image plane, their distance on the ground plane is calculated and averaged to estimate $E(l_2)$.

4.2.1. Network performance

We test the visual result with two models of *cgan_basic* and *cgan_lstm*. The *cgan_basic* model does not include a LSTM unit. The result is presented in Table 1.

From the table we can see, *cgan_lstm* outperforms *cgan_basic* for all the three criteria. Thus, the LSTM unit has shown its effectiveness to shape driving intentions with time continuity considered. The model of *cgan_lstm* has achieved an IoU of 78.01% which means the prediction has covered at least 78.01% area of human demonstration and the prediction scale is between 0.78× to 1.3× that of demonstration area. Δyaw is around 9 degrees, which implies a similar heading direction to the ground truth (Fig. 8 shows an example of $\Delta yaw = 11°$). The overall performance of $E(l_2)$ is 0.171 m, and the image-level prediction accuracies under different distance thresholds are shown in Fig. 9.

Fig. 9 shows the accuracies of *cgan_basic* and *cgan_lstm* respectively. The model of *cgan_lstm* performs much better than *cgan_basic* and gets over 90% accuracy with a threshold of 0.3m. The work in [29] has focused on a threshold of 1m accuracy for trajectory prediction and the work in [30] get an overall $E(l_2)$ of 0.77m. Therefore, our result is a relatively favorable value.

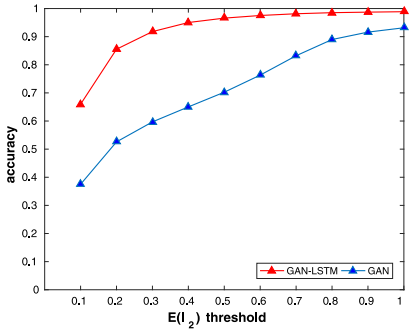
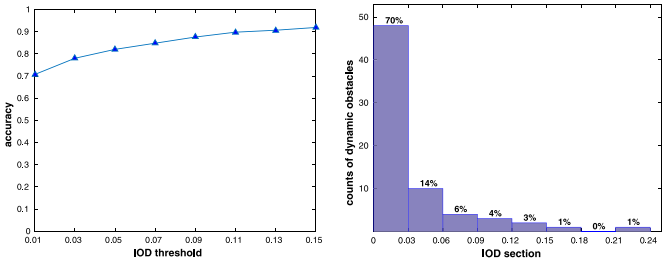


Fig. 9. Intention generation accuracies under different thresholds.

Fig. 10 has provided some visual results of driving intention predictions. In general, turning classes have more shape variations to that of human demonstration while still keep a similar direction to human control. Besides, an obstacle avoidance behavior is observed in the result, as shown with the yellow boxes. Some generated intentions have slightly adjusted there shapes to avoid dynamic obstacles without explicitly learning it. For further quantitative study, we have manually annotated 72 dynamic objects in the dataset and calculated the percentage of driving intention falling inside the annotated bounding boxes as evaluation criterion, which is denoted as IoD (Intersection over Dynamic objects). The result is shown in Fig. 11.

Fig. 11(a) is the image-level accuracies. The correct prediction is evaluated by whether its IoD is less than a pre-defined threshold. It can be seen that around 80% images can correctly avoid current obstacles with a threshold of 0.03 and more than 90% images have an IOD below 0.15. Fig. 11(a) presents the percentage of correctly avoided obstacles in object-level. Among the 72 objects, around 70% are successfully avoided with a mean IOD below 0.03 and around 90% are avoided with a threshold of 0.09. Nevertheless, there are still failed cases with more than 20% coverage on dynamic obstacles. This is also one of the motivations behind this work which has separated the vision variation with that of motion control. It is efficient to deploy recognition tasks on informative visual input while may be hard to get 100% accuracy. Thus, the motion planning module can be further fused with other concrete distance perceptions to ensure vehicle safety. More results on motion planning are presented in Section 4.3.



(a) Image accuracies (b) Object accuracies

Fig. 11. Performance on dynamic object avoidance.

4.2.2. Robustness to localization errors

The method assumes to utilize public navigation softwares with GPS signal. As the GPS signal commonly provides rough localization results, this section discusses the model performance under different localization errors. In order to achieve it, random

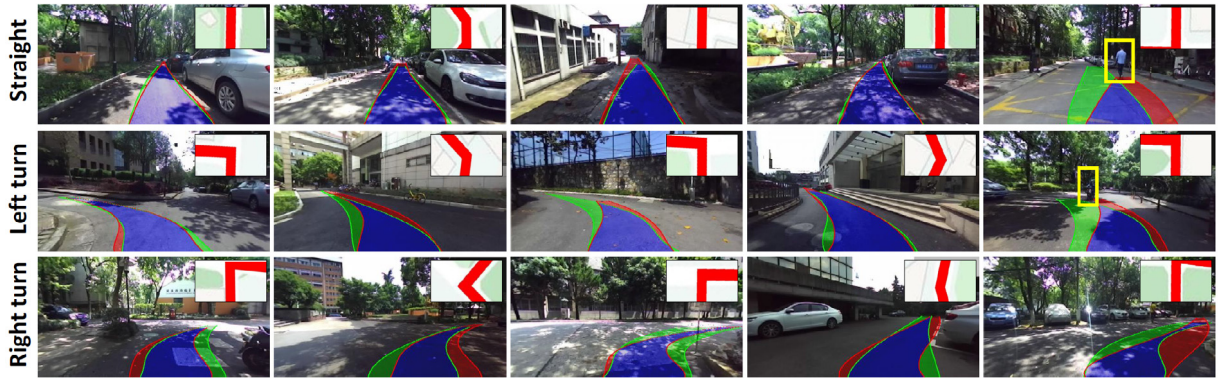


Fig. 10. Visual results from cGAN-LSTM. Green, red, and blue colors represent prediction, human demonstration and their intersection respectively. The local route plans are shown in the top right corner of the image. The intention boundaries are fitted with splines in these figures. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

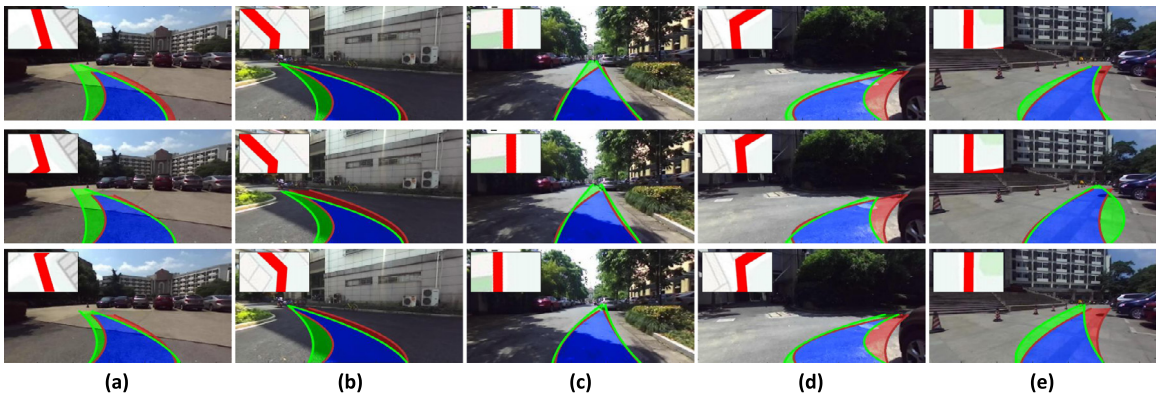


Fig. 12. Robustness to localization errors. From top to bottom: minor (0 m∼1 m), moderate (1 m∼2.5 m) and hard (2.5 m∼5 m) errors. Green, red and blue colors represent the prediction, human demonstration and their intersection respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 2
Robustness to localization errors.

Model	minor(0 m∼1 m)			moderate(1 m∼2.5 m)			hard(2.5 m∼5 m)		
	IoU	$E(l_2)$	Δy	IoU	$E(l_2)$	Δy	IoU	$E(l_2)$	Δy
basic	61.94	0.559	13.64	61.96	0.560	13.62	61.93	0.557	13.61
lstm	76.86	0.174	9.51	76.52	0.180	9.88	76.37	0.179	10.08

Abbreviations: basic(*cgan_basic*), lstm(*cgan_lstm*), Δy(Δyaw).

offsets on both horizontal and vertical direction are added when rendering local route plans in Section 3.1.2. The random offsets go into three levels: easy, moderate, and hard, with each level corresponds to a localization error of 0 m∼1 m, 1 m∼2.5 m, and 2.5 m∼5 m respectively. The result is presented in Table 2

As can be seen in the table, the two models have basically achieved stable performances given different levels of route offsets. To compare with the result from center-view routes in Table 1, only Δyaw of *cgan_lstm* has a slight increase with the growing of localization errors, while the other two criteria have remained in similar values to the previous results. Thus the model has shown robustness to potential localization errors. Some visual result from *cgan_lstm* with different level of route offsets are shown in Fig. 12.

4.2.3. *Discussion*: *multi-modal behaviors*

Annotation from human demonstration only validates a single driving intention with pre-defined local route plan, while multi-modal behaviors are presented when approaching intersections and open areas. For a further qualitative evaluation, three fake route plans are made to test the model, which are intuitively viewed as {go-straight, turn-right, turn-left}. These route plans are used to generate driving intentions with each of the test images. Fig. 13 has provided some direct results from network output.

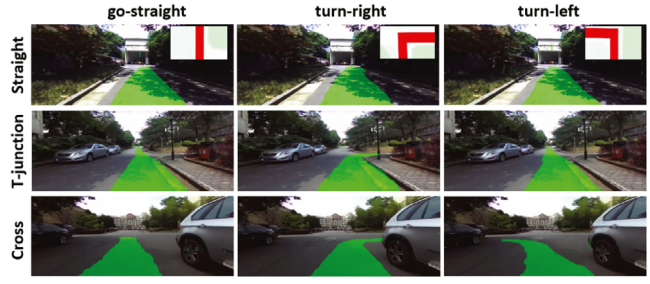


Fig. 13. Multi-modal driving behaviors with different road types. From top to bottom: one-way straight road, T-junction road, and cross area. Green color shows the original output from the network. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

The three rows show the road types of straight, T-junction and intersection respectively. The local route plans are shown in the top right conner on the first row. For the first two classes, there are route plans that may not be allowed on current road situations. In this case, the model can still generate reasonable intentions following specific road structures. While for the last class, where all routes can be performed, the model generates different driving intentions accordingly. Here, we did not use a spline to fit the intention boundary, which better shows the directional differences on network outputs for different routes.

Fig. 14 specifically presents a group of images when facing a moving car with different route plans. The result is visually compared with the model of *pix2pix* [22] which does not include local routes for intention generation. When there are dynamic obstacles appeared in front of the vehicle, the proposed model can generate intention area to fit both planning intention and the dynamic obstacle.

4.3. *Motion generation result*

In order to show the effectiveness of proposed driving intention on motion generation, a straightforward motion planning is performed by scoring candidate driving curves. For comparison to end2end approach, we implement the network structure in [5], which generates direct control commands also with public route plans. Their driving commands are presented as velocity and angular speed. We calculate driving curvature as $c_t = \frac{w_t}{v_t}$, for each time t referring to the work in [14]. As for the 'ground truth' from human demonstration, the actual trajectory curvature is calculated for comparison.

Evaluation Metrics. To quantitatively evaluate the system performance, the motion prediction accuracies are calculated for different control precisions. The definition of true positive is illustrated in Fig. 15.

The black dials indicate different motion resolutions, for which the proposed method generates a same number of candidate motion commands, as shown with the gray curves. Green grid indicates final model prediction with highest score in navigation score map, and red grid indicates ground truth from human demonstration. Then the prediction accuracy is measured

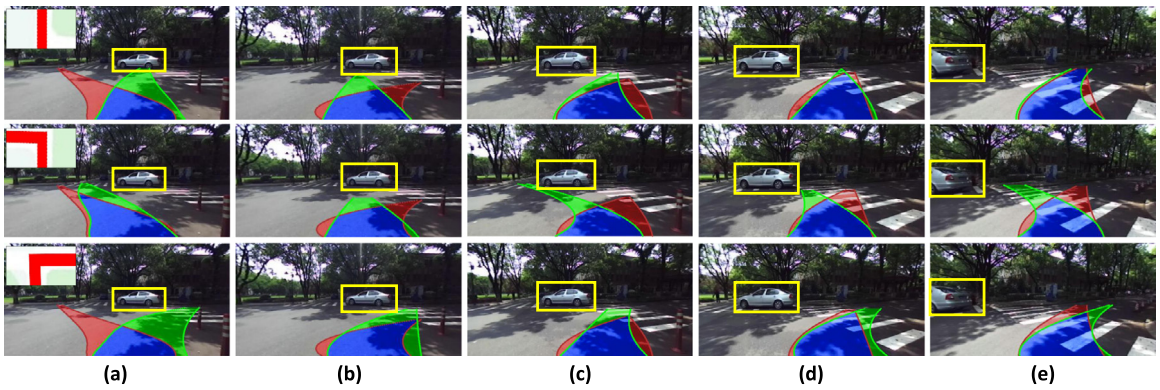


Fig. 14. Multi-modal driving behaviors when facing a moving car. Green, red and blue show the results from cGAN-LSTM, pix2pix, and their intersection respectively. The intention boundaries are fitted with splines in these figures. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

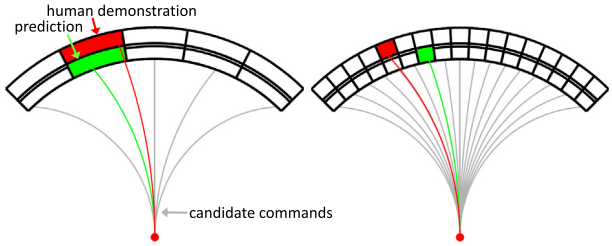


Fig. 15. Evaluation criterion for motion generation. Gray curves show the candidate driving commands in relation to the control precision. Green curve indicates the command with a highest score in the navigation score map. And red color shows the ground truth maneuvered by human demonstration. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

with the grid distance Δg from model prediction to human demonstration:

$$accuracy = \frac{\sum_{t=0}^{T} [|pred_t - human_t| \leq \Delta g]}{T}$$

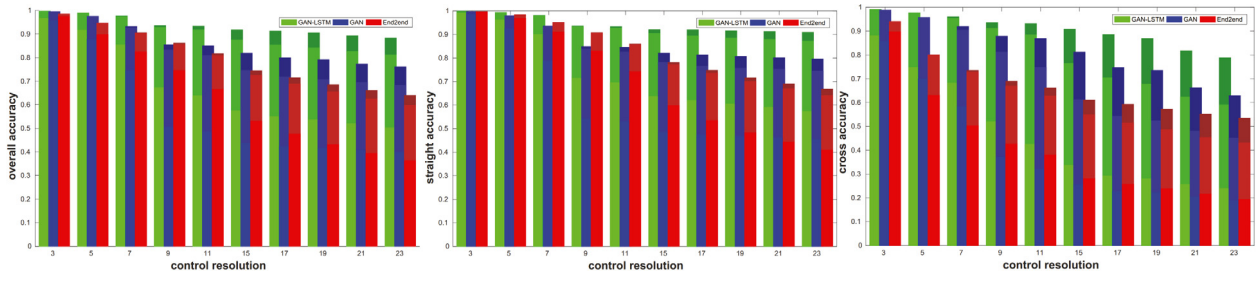
where T is the total steps of test data. $[\cdot]$ equals to 1 if the formulation inside is true and otherwise equals to 0. For comprehensive evaluation, we provide quantitative results under three settings of $\Delta g = \{0, 1, 2\}$ with control resolutions range from 3 to 23. The resolution of 3 means the prediction only distinguishes the directions from right, left to straight. And a resolution of 23 represents a control precision of less than 4°.

4.3.1. Model performance

The two models of *cgan_basic* and *cgan_lstm* are both evaluated to compare with the end2end [5] method. The results with different motion resolutions are shown in Fig. 16.

The three figures present accuracy on complete test route, long straight test route, and turning route respectively. For an overall accuracy measurement, *cgan_lstm* achieves better performance than end2end method under most settings, and shows better robustness to different control resolutions. When the control resolution goes up, *cgan_lstm* also generates more candidate curves for a careful search to better fit for the driving intention. This shows the visual driving intention is adaptive to different motion requirements regardless of specific control precision in the demonstration driving, as the intention region has incorporated demonstration control to local road reference instead of numerical imitation. Some visual results from the proposed system is shown in Fig. 17. The last case in the first row specifically shows a control example for dynamic object avoidance, which has planned a curve different to the heading direction of intention region to avoid the detected obstacle. In contrast, end2end approach learns a numerical mapping from visual input to motion output regardless of different control resolutions, thus it gradually decreases when the evaluation gets more strict.

For the performance on separate straight class and turning classes, advantage of the proposed model is more significant for the turning road sections which are more challenging due to the complex road situations. A more intuitive comparison of error rates along test route is provided in Fig. 18. As the figure shows, end2end method has more errors when vehicle is approaching turning road sections. We consider the reason is the numerical differences of demonstration control when facing a same turning class. A big turn requires a small driving curvature and a small turn may require a big driving curvature. Besides, human may change the driving commands during turning for obstacle avoidance. This also increases the intra-class variation for similar perceptions in the turning classes. Thus, it may be complex in



(a) Overall accuracy (b) Straight road accuracy (c) Turning roa accuracy

Fig. 16. Motion prediction performances. Green, blue, and red represent the performances of *cgan_lstm*, *cgan_basic* and end2end respectively. For each individual model result, color from light to dark represent a grid distance threshold of 0,1,2 for Δg. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

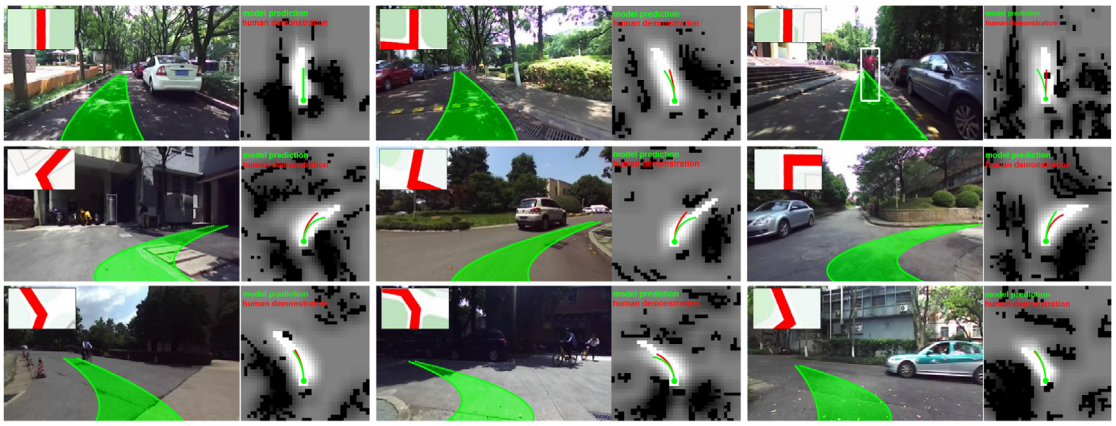
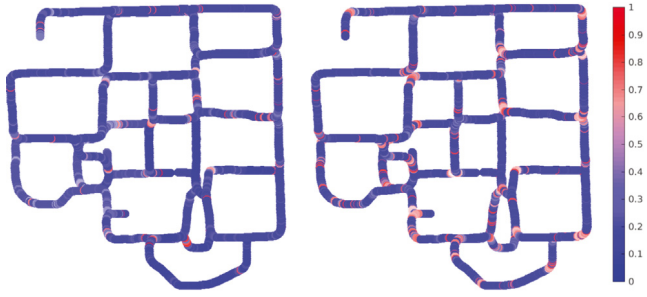
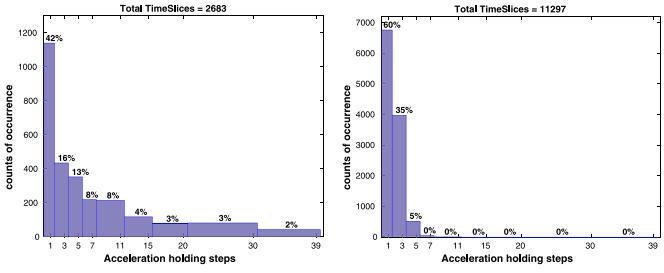


Fig. 17. Motion generation result compared with human demonstration. For each case, top figure shows the visual result and bottom figure shows the generated motion in navigation score map. Green and red colors represent the model prediction and human demonstration respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



(a) cGAN-LSTM (b) End-to-end

Fig. 18. Error rate densities along the test route. The results come from a control resolution of 7-grids. The blue color indicates a better result. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



(a) cGAN-LSTM (b) End-to-end

Fig. 19. Evaluation of acceleration holding steps for different models.

end2end training to consider both variations in perception and planning. In contrast, for the long straight road sections, most commands stay near-linear and unchanged for a long time, which can be efficiently learned by both models.

Besides the evaluation on discrete control commands, two metrics are provided to show the smoothness of generated motion commands. The first one is the standard deviation of command difference between model prediction with human demonstration. We calculate it on the complete test route. The proposed method gets 0.0244 and the end2end method gets 0.0311, which shows the proposed method has less fluctuations around the ground truth trajectory. The second one is the count of acceleration time slices along the test route. During vehicle control, less zero crossings of accelerating values indicate less fluctuated velocity directions, i.e. a better smoothness. Since the data is collected with a fixed rate of 10 Hz, the acceleration values can be efficiently estimated between consecutive frames. Then a time slice can be counted as the acceleration holding steps between two zero crossings. The results of different models are shown in Fig. 19.

Fig. 19(a) is the result of the proposed method. It decomposes test route to 2683 time slices, among which the single-step changes take a share of 42%. Fig. 19(b) is the result of end-to-end method, which decomposes the test route to 11297 time slices. There are 60% single-step changes and most time slices are less that three steps. Therefore, the proposed method can provide more smoothed driving motion than the end-to-end manner.

4.3.2. *Discussion*: *robustness to time delay*

For the design of end2end approach, motion generation is correlated with environment understanding. This makes the system has a strong reliance on real-time vision prediction. However, the vision processing alone is prone to be disturbed and the GPS signal for local route plan can be lost due to occlusions. Thus, it can be a critical ability to generate valid motion command when visual result is delayed. In this section, we investigate the motion generation ability when there are different levels of time delays for visual prediction. The results are shown in Figs. 20 and 21 for end2end method and the proposed method respectively. The two figures show sequential prediction curvatures along a section of test route. The discrete dots represent actual data points, which are fitted with smooth lines to indicate variation trends. In Fig. 20, since end2end approach outputs direct driving commands without intermediate knowledge retainment, it needs to keep the former motion command during time delay without human intervention. This is reflected on the extended dashed lines for generated motion. Nevertheless, for the proposed method in Fig. 21, there are new commands generated during time delay. And the result has shown a similar level of command dispersion with that of no time delay. Since the learned intention has a certain area on the ground plane, visual result in one frame can be efficiently integrated with following obstacle perceptions and be used to render a new navigation score map for motion generation in multiple steps.

Compared with the real-time visual prediction, most prediction errors for *cgan_lstm* appear as big curvatures. It is caused by planning circle actions for vehicle when it has passed the retained intention area. In this case, the area around vehicle position has the biggest score and the model tends to give a largest curvature

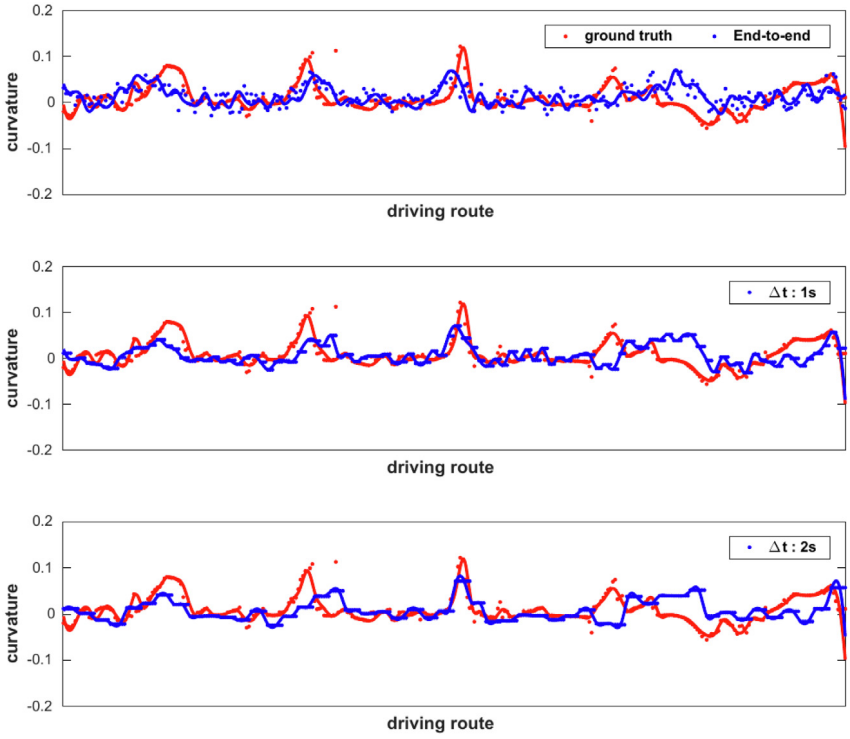


Fig. 20. Comparison of end2end method with human demonstration when visual prediction is delayed. From top to bottom: no time delay, 1 s delay, and 2 s delay. Blue color indicates the model prediction and red color denotes human demonstration. The dots denote actual data points, which are fitted to a smooth line to show the prediction tendency. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

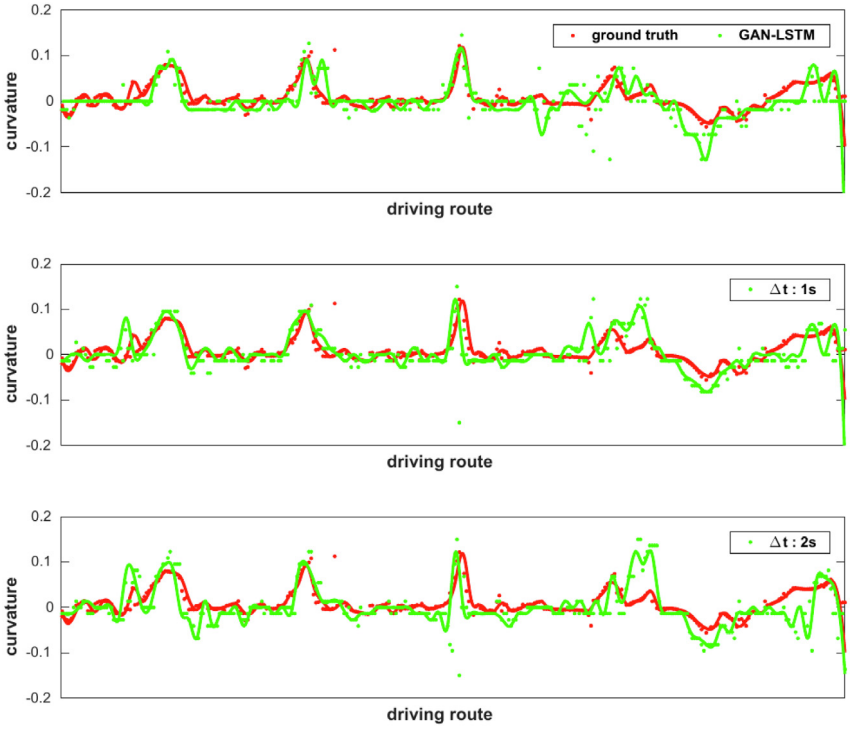


Fig. 21. Comparison of proposed method with human demonstration when visual prediction is delayed. From top to bottom: no time delay, 1 s delay, and 2 s delay. Green color indicates the model prediction and red color denotes human demonstration. The dots denote actual data points, which are fitted to a smooth line to show the prediction tendency. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

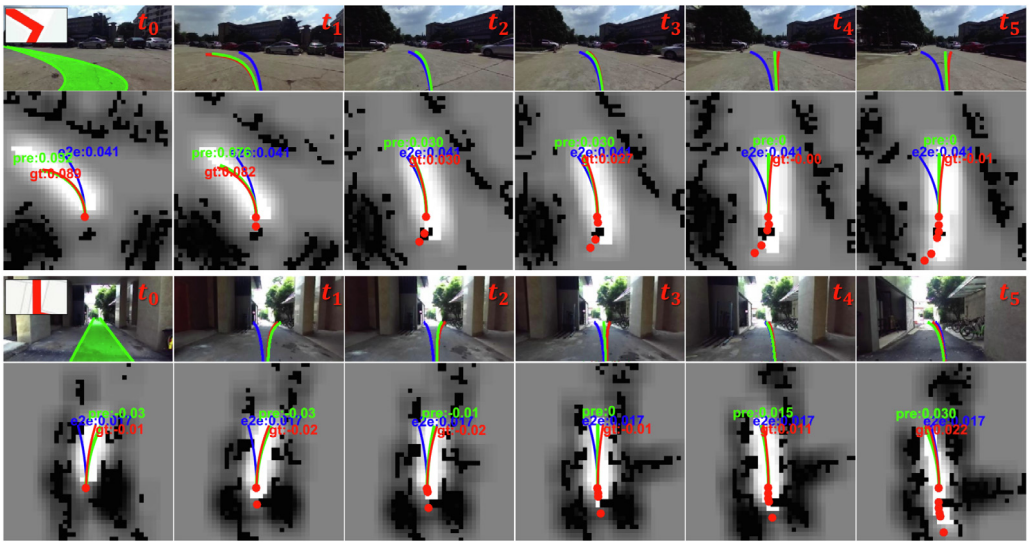


Fig. 22. Robustness to time delay. The time difference between consecutive frames is 0.3s. Green, blue, and red colors indicate motion generation results of cGAN-LSTM, end2end, and human demonstration respectively. The red points in the navigation score map indicates vehicle previous poses. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

to stay nearby. Therefore, the longest time that the proposed method can handle depends on the valid area from driving intention projection as well as the driving speed of vehicle. Some visual results for motion generation with a delay of 1.8 s are shown in Fig. 22.

For the proposed approach, it generates a visual driving intention on the first frame according to local route plan and renders a navigation score map with laser perception integrated. Then for the following frames, the navigation score maps are integrated from the driving intention in the first frame and the laser perceptions in the subsequent frames. To indicate vehicle movements inside the retained driving intention area, vehicle previous poses are also plotted with discrete red dots as shown in Fig. 22. The transformation of consecutive local coordinates can be approximately estimated by laser map registration or the inference from motion model.

In summary, the proposed method separates variations of visual understanding with motion generation. Compared with end2end approach, it demonstrates a better prediction performance.

5. Conclusions

In this paper, an innovative learning model is developed for automotive driving research with an intermediate representation of driving intention. It is learned from image perception and publicly available route planner to indicate the following driving area. The intention area is adaptive for modular fusion and can be efficiently encoded into a navigation score map for motion generation. In this way, variations of motion planning are separated with those of visual understanding, which incorporates modular flexibility and reliability compared with end2end approach. Nevertheless, the learning input of the method is the same as end2end approach and does not rely on precise localization result. The key of the system is a cGAN network inserted with LSTM unit to learn from human demonstration. The adversarial loss enables a weakly-supervised training manner that leverages single-modal demonstration data to achieve generalization on multi-modal behaviors in strange scenarios. Experimental results indicate the proposed method outperforms end2end approach in both accuracy and robustness. Our future work will consider an indoor visual navigation framework without precise localization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported in part by the National Key R&D Program of China (2018YFB1309300), and in part by the National Natural Science Foundation of China (U1609210, 61903332).

References

[1] M. Bojarski, D.D. Testa, D. Dworakowski, B. Firner, K. Zieba, End to end learning for self-driving cars, 2016.
[2] F. Codevilla, M. Müller, A. Dosovitskiy, A. López, V. Koltun, End-to-end driving via conditional imitation learning, 2017.
[3] H. Xu, G. Yang, F. Yu, T. Darrell, End-to-end learning of driving models from large-scale video datasets, in: IEEE Conference on Computer Vision and Pattern Recognition, 2017.
[4] W. Gao, D. Hsu, W.S. Lee, S. Shen, K. Subramanian, Intention-net: Integrating planning and deep learning for goal-directed autonomous navigation, 2017, arXiv preprint arXiv:1710.05627.
[5] S. Hecker, D. Dai, L. Van Gool, End-to-end learning of driving models with surround-view cameras and route planners, in: Proceedings of the European Conference on Computer Vision, ECCV, 2018, pp. 435–453.
[6] A. Sauer, N. Savinov, A. Geiger, Conditional affordance learning for driving in urban environments, 2018, arXiv preprint arXiv:1806.06498.
[7] D. Fox, W. Burgard, S. Thrun, The dynamic window approach to collision avoidance, IEEE Robot. Autom. Mag. 4 (1) (1997) 23–33.
[8] J. Redmon, A. Farhadi, Yolov3: An incremental improvement, 2018, arXiv preprint arXiv:1804.02767.
[9] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, A.C. Berg, Ssd: Single shot multibox detector, in: European Conference on Computer Vision, Springer, 2016, pp. 21–37.
[10] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A.L. Yuille, Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs, IEEE Trans. Pattern Anal. Mach. Intell. 40 (4) (2018) 834–848.
[11] Q. Luo, H. Ma, Y. Wang, L. Tang, R. Xiong, 3D-SSD: Learning hierarchical features from RGB-D images for amodal 3D object detection, 2017, arXiv preprint arXiv:1711.00238.
[12] J. Li, X. Liang, S. Shen, T. Xu, J. Feng, S. Yan, Scale-aware fast R-CNN for pedestrian detection, IEEE Trans. Multimed. 20 (4) (2018) 985–996.
[13] F. Yang, W. Choi, Y. Lin, Exploit all the layers: Fast and accurate cnn object detector with scale dependent pooling and cascaded rejection classifiers, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 2129–2137.

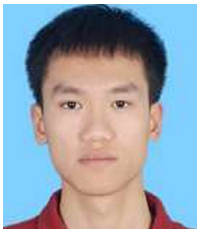
[14] A. Amini, G. Rosman, S. Karaman, D. Rus, Variational end-to-end navigation and localization, 2018, arXiv preprint arXiv:1811.10119.
[15] C. Chen, A. Seff, A. Kornhauser, J. Xiao, Deepdriving: Learning affordance for direct perception in autonomous driving, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 2722–2730.
[16] M. Al-Qizwini, I. Barjasteh, H. Al-Qassab, H. Radha, Deep learning algorithm for autonomous driving using googlenet, 2017 IEEE Intelligent Vehicles Symposium (IV), IEEE, 2017, pp. 89–96.
[17] A. Seff, J. Xiao, Learning from maps: Visual common sense for autonomous driving, 2016, arXiv preprint arXiv:1611.08583.
[18] D. Barnes, W. Maddern, I. Posner, Find your own way: Weakly-supervised segmentation of path proposals for urban autonomy, in: Robotics and Automation (ICRA), 2017 IEEE International Conference on, IEEE, 2017, pp. 203–210.
[19] L. Tang, X. Ding, H. Yin, Y. Wang, R. Xiong, From one to many: Unsupervised traversable area segmentation in off-road environment, in: 2017 IEEE International Conference on Robotics and Biomimetics, ROBIO, IEEE, 2017, pp. 787–792.
[20] Y. Wang, Y. Liu, Y. Liao, R. Xiong, Scalable learning framework for traversable region detection fusing with appearance and geometrical information, IEEE Trans. Intell. Transp. Syst. (2017).
[21] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, in: Advances in Neural Information Processing Systems, 2014, pp. 2672–2680.
[22] P. Isola, J.-Y. Zhu, T. Zhou, A.A. Efros, Image-to-image translation with conditional adversarial networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1125–1134.
[23] M. Mirza, S. Osindero, Conditional generative adversarial nets, 2014, arXiv preprint arXiv:1411.1784.
[24] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2015, pp. 234–241.
[25] D.J. Berndt, J. Clifford, Using dynamic time warping to find patterns in time series, in: KDD Workshop, Vol. 10, No. 16, Seattle, WA, 1994, pp. 359–370.
[26] X. Ding, Y. Wang, D. Li, L. Tang, H. Yin, R. Xiong, Laser map aided visual inertial localization in changing environment, in: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2018, pp. 4794–4801.
[27] L. Tang, Y. Wang, X. Ding, H. Yin, R. Xiong, S. Huang, Topological local-metric framework for mobile robots navigation: a long term perspective, Auton. Robots 43 (1) (2019) 197–211.
[28] H. Yin, Y. Wang, L. Tang, X. Ding, R. Xiong, LocNet: Global localization in 3D point clouds for mobile robots, in: Proceedings of the 2018 IEEE Intelligent Vehicles Symposium, Vol. IV, Changshu, China, 2018, pp. 26–30.
[29] S. Pellegrini, A. Ess, K. Schindler, L.J.V. Gool, You'll never walk alone: Modeling social behavior for multi-target tracking, in: IEEE 12th International Conference on Computer Vision, ICCV 2009, Kyoto, Japan, September 27 - October 4, 2009, 2009.
[30] P. Cai, Y. Sun, Y. Chen, M. Liu, Vision-based trajectory planning via imitation learning for autonomous vehicles, in: 2019 IEEE Intelligent Transportation Systems Conference, ITSC, IEEE, 2019, pp. 2736–2742.
[31] M. Aly, Real time detection of lane markers in urban streets, in: Intelligent Vehicles Symposium, 2008 IEEE, IEEE, 2008, pp. 7–12.

Huifang Ma received the B.S. degree from Department of Control Science and Engineering, Zhejiang University, Hangzhou, P.R. China in 2014. She is currently pursuing the Ph.D. degree from Zhejiang University. Her research interests include semantic segmentation, open-set recognition and visual navigation.

Yue Wang received Ph.D. from Department of Control Science and Engineering, Zhejiang University, Hangzhou, P.R. China in 2016. He is currently a research fellow in Department of Control Science and Engineering, Zhejiang University, Hangzhou, P.R. China and the chief technology officer in iPlusBot, Hangzhou, P.R. China. His latest research interests include mobile robotics and robot perception.

Rong Xiong received the B.Sc. and M.Sc. degrees in computer science and engineering in 1994 and 1997, respectively, and the Ph.D. degree in control science and engineering in 2009, all from Zhejiang University, Hangzhou, China. She has been with the State Key Laboratory of Industrial Control Technology, Zhejiang University, since 1997, where she is currently a Professor and the Director of the Robotics Laboratory. Her research interests include robotics, environment mapping and humanoid robot.

Sarath Kodagoda received B.Sc.Eng.Hons. degree in 1995, specializing in Electrical Engineering, from the University of Moratuwa, Sri Lanka. He received his M.Eng. (2000) and Ph.D. (2004) degrees specializing in robotics from the Nanyang Technological University, Singapore. He is currently an Associate Professor with the ARC Center for Autonomous Systems (CAS) at the University of Technology, Sydney. His research interests include infrastructure robotics, human robot Interaction, machine learning, and mobile robotics.



Li Tang received B.S. from Department of Control Science and Engineering, Zhejiang University, Hangzhou, P.R. China in 2015. He is currently a PhD candidate in Department of Control Science and Engineering, Zhejiang University, Hangzhou, P.R. China. His research interests include vision based localization and autonomous navigation.