

Preface

The 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2019) is the flagship international conference in robotics and intelligent systems. It is co-sponsored by the IEEE, the IEEE Robotics and Automation Society (RAS), the IEEE Industrial Electronics Society (IES), the Robotics Society of Japan (RSJ), the Society of Instruments and Control Engineers (SICE), and the New Technology Foundation (NTF). IEEE is a non-profit, technical professional association of more than 400,000 members in 160 countries. It is a leading authority in technical areas ranging from computer engineering, biomedical technology and telecommunications, to electric power, aerospace and consumer electronics, among others.

This volume contains the papers presented at the workshop TCV2019: Towards Cognitive Vehicles: perception, learning and decision making under real-world constraints. Is bio-inspiration helpful? held on November 8, 2019 at IROS.

Objectives of the workshop:

Autonomous driving is only one out of many aspects of intelligence required for future transportation systems. Human-machine interaction in a cognitive vehicle is an intriguing use case that requires intelligence beyond the state of the art in machine learning, computer vision, and AI. For safe and convenient human-machine interaction, the intelligent system such as a smart vehicle needs to be able to perceive its environment and make decisions based on the received data. Current state-of-the-art approaches to both intelligent perception and decision making typically rely on machine learning with offline training of neural networks using elaborated datasets. To enable truly adaptive intelligence, as we know it from biological systems, learning that supports decision making and perception needs to happen in real time, in an online fashion. But can such adaptive perceiving, deciding, and learning systems be safe enough to actually be deployed in an intelligent vehicle?

While biological inspiration has led to some of the most successful approaches in perception and machine learning – deep neural networks, – its deployment in real-world, safety-critical settings is yet limited. We aim to explore and critically discuss what biological inspiration in perception, learning, and decision making could bring in the future for increasing intelligence of vehicles and other robotic systems.

Thus, the aim of the workshop is to discuss potential benefits and pitfalls in applying bio-inspired approaches when developing intelligent real-world systems that perceive, interact, learn, and make decisions. We will focus on the application area of intelligent, "cognitive" vehicles and will use an unconventional format: for each of three subtopics we invited 2-4 experts from different schools of thought (for example, traditional machine learning and brain-inspired learning, conventional approach to planning and decision making and cognitive architecture-based approach, event-based bio-inspired vision and conventional machine visions, etc.). Each speaker will give a short introductory talk followed by a moderated panel discussion around each topic. Furthermore, we will invite researchers from intelligent robotics and vehicles with a focus on perception, learning and decision making to present their work in posters and short spot-light talks.

The workshop will stimulate discussion of the role of biological inspiration in the development of future AI systems in the context of real-world, safety-critical applications of robotic systems in environments shared with humans.

Topics of interest:

Applications

- Intelligent vehicles (cars, UAVs, ...)

- Human-machine interaction

- Intelligence in the cockpit

Perception

- Robust accountable and scalable perception with neural networks and without

- Multi-modal perception and sensory integration

- Attention and cognitive control in visual and tactile perception

- Gesture recognition

- Perception for action

Learning

- Machine learning for vehicles

- Fast inference and learning

- Online learning and reliability

- Embedded machine learning

- Learning in complex hierarchical control systems

Cognitive Architectures

- Cognitive architectures and machine learning / neuronal networks

- Cognitive architectures for action selection

- Scalable cognitive architectures

- Learning cognitive architectures

We would like to thank the technical committee for cognitive robotics of the IEEE Robotics & Automation Society, NEUROTECH, BMW Group and BOSCH for their support.

November 8, 2019
Macau

Florian Mirus
Mohsen Kaboli
Yulia Sandamirskaya
Nicolai Waniek

Table of Contents

Enhancing Object Detection in Adverse Conditions using Thermal Imaging..... <i>Kshitij Agrawal and Anbumani Subramanian</i>	1
Exploration for Objects Labelling Guided by Environment Semantics using UAVs	4
<i>Reem Ashour, Tarek Taha, Jorge Dias, Lakmal Seneviratne and Nawaf Almousa</i>	
Towards game theoretic AV controllers: measuring pedestrian behaviour in Virtual Reality..... <i>Fanta Camara, Patrick Dickinson, Natasha Merat and Charles W. Fox</i>	7
MSPRT action selection model for bio-inspired autonomous driving and intention prediction..... <i>Riccardo Donà, Gastone Pietro Rosati Papini and Giammarco Valenti</i>	11
A dynamic neural model for endowing intelligent cars with the ability to learn driver routines: where to go, when to arrive and how long to stay there?	15
<i>Flora Ferreira, Weronika Wojtak, Wolfram Erlhagen, Paulo Vicente, Ankit Patel, Sérgio Monteiro and Estela Bicho</i>	
Towards an Evaluation Methodology for the Environment Perception of Automotive Sensor Setups	19
<i>Maike Hartstern, Viktor Rack and Wilhelm Stork</i>	
Risk-Aware Reasoning for Autonomous Vehicles..... <i>Majid Khonji, Jorge Dias and Lakmal Seneviratne</i>	23
Cognitively-inspired episodic imagination for self-driving vehicles	28
<i>Sara Mahmoud, Henrik Svensson and Serge Thill</i>	
A Cognitively Informed Perception Model for Driving	32
<i>Alice Plebe and Mauro Da Lio</i>	
Cognitive Wheelchair: A Personal Mobility Platform	36
<i>Mahendran Subramanian, Suhyung Park, Pavel Orlov and Aldo Faisal</i>	
A Frontal Cortical Loop For Autonomous Vehicles Using Neuralized Perception-Action Hierarchies..... <i>David Windridge and Seyed Ali Ghorashi</i>	40
Following Social Groups: Socially-Compliant Autonomous Navigation in Dense Crowds ... <i>Xinjie Yao, Ji Zhang and Jean Oh</i>	44

Author Index

Agrawal, Kshitij	1
Almousa, Nawaf	4
Ashour, Reem	4
Bicho, Estela	15
Camara, Fanta	7
Da Lio, Mauro	32
Dias, Jorge	4, 23
Dickinson, Patrick	7
Donà, Riccardo	11
Erlhagen, Wolfram	15
Faisal, Aldo	36
Ferreira, Flora	15
Fox, Charles W.	7
Ghorashi, Seyed Ali	40
Hartstern, Maike	19
Khonji, Majid	23
Mahmoud, Sara	28
Merat, Natasha	7
Monteiro, Sérgio	15
Oh, Jean	44
Orlov, Pavel	36
Park, Suhyung	36
Patel, Ankit	15
Plebe, Alice	32
Rack, Viktor	19
Rosati Papini, Gastone Pietro	11
Seneviratne, Lakmal	4, 23
Stork, Wilhelm	19
Subramanian, Anbumani	1
Subramanian, Mahendran	36
Svensson, Henrik	28

Taha, Tarek	4
Thill, Serge	28
Valenti, Giamarco	11
Vicente, Paulo	15
Windridge, David	40
Wojtak, Weronika	15
Yao, Xinjie	44
Zhang, Ji	44

Keyword Index

3D reconstruction	4
action selection problem	11
affordance competition	11
Artificial Cognition	28
autonomous driving	11, 32
Autonomous driving robot	44
Autonomous Exploration	4
Autonomous Vehicles	7, 28
autonomous vehicles	23
Autonomous Wheelchair	36
bio-inspired cognitive models	11
chance constraints	23
Cognitive architectures and neural networks	15
Cognitive architectures for action selection	7
Pedestrian Behaviour	7
Cognitive Vehicle	36
Cognitive Vehicles	40
convergence-divergence zones	32
Cost function	4
deep learning	32
Deep Learning	40
Deep Reinforcement Learning	28
Driver routines	15
Episodic Imagination	28
external vehicle sensors	19
Eye Gaze	36
Fast inference and learning	15
faster-rcnn	1
First Order Logic	40
flir	1
free energy	32
Game Theory	7
Gaze tracking	36
Intelligent vechicles (cars)	15
Intention Decoding	36
MSPRT	11

object detection	1
Online learning	15
perception	19
Perception-Action Learning	40
risk-aware planning	23
Safety in dense crowds	44
Semantic Exploration and Mapping	4
Semi-autonomous Wheelchair	36
sensor performance	19
Severely Disabled	36
Simulation	28
simulation	19
Social navigation	44
thermal imaging	1
Utility function	4
variational autoencoder	32
vehicle sensor setup configuration	19
virtual testing	19
WTA	11

Program Committee

Mohsen Kaboli
Florian Mirus
Yulia Sandamirskaya
Nicolai Waniek

Institute for Cognitive Systems
BMW
University of Zurich
Bosch Center for Artificial Intelligence

Enhancing Object Detection in Adverse Conditions using Thermal Imaging

Anonymous submission

Abstract— Autonomous driving relies on deriving understanding of objects and scenes through images. These images are often captured by sensors in the visible spectrum. For improved detection capabilities we propose the use of thermal sensors to augment the vision capabilities of an autonomous vehicle. In this paper, we present our investigations on the fusion of visible and thermal spectrum images using a publicly available dataset, and use it to analyze the performance of object recognition on other known driving datasets. We present a comparison of object detection in night time imagery and qualitatively demonstrate that thermal images significantly improve detection accuracies.

I. INTRODUCTION

Object detection is one of the primary component for scene understanding in an autonomous vehicle. The detected objects are used to plan the trajectory of a vehicle. Cameras are used to capture images of the environment, which are then input to a myriad of computer vision tasks, including object detection.

While significant progress has been achieved in using visible spectrum for object detection algorithms, it poses inherent limitations due to the response from cameras in visible spectrum. Some of the shortcomings include low dynamic range, slow exposure adjustment, inefficiencies in high contrast scenes etc, while being subject to weather conditions like fog and rain. Bio inspired vision, like infrared based thermal vision, could be an effective tool to augment the shortcomings of imagers that operate in the visible spectrum.

Other sensing modalities like LIDAR based systems are sufficient to detect depth in a scene. However, the data may be too coarse to detect objects at further distances and may lack resolution to classify objects. Thermal imagers on the other hand can easily visualize objects that emit infra-red radiation due to their inherent heat. Due to this property, thermal imagers can visualize important participants on the road like people, cars and animals at any time of the day. Augmenting the detection of objects in the thermal spectrum could be a good way to enable robust object detection for safety critical systems like autonomous vehicles.

Object detection methods have progressed significantly over the years from simple contour based methods using support vector machines (SVM) [1-7] to ones using deep classification models [16]–[20] that utilize hierarchical representation of data. Data driven models are the flavor of the day by dominating the detection benchmarks on large scale datasets like PASCAL VOC [8] and COCO [9].

There is a large body of work done for recognizing and localizing objects in the visible spectrum to recognize objects like people [13, 14], vehicles [10] and traffic lights.

The features extracted from an image can help identify an object in good lighting and normal weather conditions. However, images obtained using camera systems in low light conditions - night, dusk and dawn, and adverse weather conditions - rain and snow, contain partially illuminated objects, low contrast and low information content. These images are often difficult for object detection algorithms.

The primary contribution of our work is to investigate the nature of object detectors in the thermal spectrum in driving scenarios for autonomous navigation. We utilize the FLIR ADAS [11] dataset that consists of annotated thermal images and time synchronized visible images. Datasets like KAIST [12] exist for similar purpose, however they are limited to annotations of only people.

The next sections are organized as follows: in Section 2, we will cover related research, Section 3, we deal with the datasets, generation of a ground truth for the visible and thermal pairs in the FLIR ADAS dataset and the setup of our experiment. In Section 4 we will present our result and subsequent conclusion in Section 5.

II. RELATED WORK

Object detection consists of recognition and localization of object boundaries within an image. Early work in the computer vision field has focused on building task based classifiers using specific image properties. In some of the earlier approaches a sliding window is used to classify parts of an image based on feature pyramids [15], histogram of oriented gradients (HoG) with a combination of SVM has been used to classify pedestrians [13] and features pools of Haar features [14] have been employed for face detection.

A more generalized form of object detection has evolved over the years due to the advancement in deep learning. The exhaustive search for classification has been replaced by convolutional classifiers. Object detection models have been proposed to work with relative good accuracy on the visible spectrum using models that consist of a) a two stage system a classifier connected with a region proposal network, RCNN [16], Fast-RCNN [17] and Faster-RCNN [18] b) a single stage network with the classification and localization layers in a cohesive space, like YOLO [19] and SSD [20].

Models trained on large scale datasets are known to perform to quite a good extent. With driving datasets like KITTI [21], Cityscapes [22] the object detection models have been employed to detect pedestrians, cars and bicycles.

Some work has been done in the detection of objects thermal images [23-26], especially focusing on human detection. Since some of the work has been from static camera,



Fig. 1. Annotated and RGB translated pairs from FLIR ADAS dataset

the proposals can be generated from background subtraction techniques in the thermal domain [26]. However, most of the work does not deal with investigating the effect of multiple day and night conditions across the thermal and visible spectrum in driving scenarios.

III. DATASETS

In this section we will detail the datasets that we utilize in our study and the process we employed to create a baseline for training the Faster-RCNN model.

A. FLIR ADAS Dataset

The FLIR thermal imaging dataset is acquired via a RGB and thermal camera mounted on a vehicle with annotations created for 14,452 thermal images. It primarily is captured in streets and highways in Santa Barbara, California, USA from November to May with clear-sky conditions at both day and night. Annotations exist for the thermal images based on the COCO annotation scheme. However, no annotations exist for the corresponding visible images.

To analyse the night time performance for object detection it was absolutely essential to have corresponding annotated images in the visible spectrum in the day and night scenarios. We build a custom point based correspondence generator and utilized 8 point homography method to generate a correspondence from the thermal to the visible spectrum. Using such

methods we are able to translate the annotations to the visible space as well resulting in about 8000 training and 1247 validation images with 42-58 split in night vs day. In the rest of our work we refer to this translated dataset as the FLIR RGB dataset. Fig 1 shows the translation of bounding boxes from the thermal images to the corresponding registered image in the RGB domain. The input images as part of the FLIR dataset are uncorrected images and slight radial distortions due to the lens can be visualized. The drawback of our technique is that the points closer to the center can be registered, however, the points radially distant from the center do not align well.

TABLE I

SCHEME SHOWING MAPPING OF LABELS

FLIR	IDD	KITTI
Person	Person	Pedestrian
	Rider	Cyclist
Car	Car	Car
	Caravan	-
	Autorickshaw	-
Bicycle	Bicycle	-
	Motorcycle	-
Dog	Animal	-
-	Bus	Truck
	Trailer	
	Truck	
	Vehicle fallback	

B. Indian Driving Dataset

The India Driving Dataset [27] consists of images taken in driving conditions in city and highway situations primarily during the day. It is unique in the 26 classes that it proposes and the high number of objects in each scene. We pick common traffic participants that also exist in the FLIR dataset and translate them to similar labels. Table 1 shows the translation mechanism.

C. KITTI

The KITTI object detection dataset consist of day time images captured in the urban and highway driving conditions in Karlsruhe, Germany. Again classes corresponding to the FLIR dataset are chosen and translated. A detailed translation can be seen from Table 1.

IV. EXPERIMENT & RESULTS

The Faster-RCNN implementation from Ren et al [18] was used to train the model on three datasets: FLIR thermal (FLIR_THM), IDD and KITTI. The Faster-RCNN model used a Resnet-101 for the high level feature extraction and the complete model is initialized from pre-trained COCO weights. The model is trained on each dataset till convergence for about 180,000 iterations. We present the results of each baseline model performance by testing on a validation dataset from the same domain in Table 2.

In the first part of our study the trained model performance is tested on the night time images (653 out of 1247) from the translated FLIR RGB dataset. Table 3, shows that the performance of models trained in the visible spectrum degrades

significantly on the night images from the FLIR RGB. We can also see that training on FLIR thermal does not translate well to the visible domain, with a drop of 40% from the baseline inference on the FLIR thermal dataset. Thus training in the thermal domain does not improve performance in the night time on the same dataset. While training on IDD does retain the highest performance because of better correlation to road scenes from the IDD in day and night conditions.

TABLE II

AVERAGE PRECISION PER CLASS FOR DATASET COMBINATIONS TESTED ON NIGHT TIME IMAGES FROM THE FLIR RGB TRANSLATED DATASET

Train Dataset	Test Dataset	Bicycle	Car	Dog	Person	mAP
FLR.THM	FLIR_RGB	0.1312	0.571	0	0.245	0.237
IDD	FLIR_RGB	0.3314	0.625	0.042	0.365	0.341
IDD+FLIR.THM	FLIR_RGB	0.1319	0.570	0	0.260	0.240
KITTI	FLIR_RGB	-	0	-	0.403	0.201
KITTI	FLR.THM	-	0	-	0.141	0.070
KITTI	KITTI	-	0.970	-	0.899	0.935

We conduct another evaluation, a performance of domain transfer by introducing the large scale driving dataset into the training. The trained models are tested in the thermal and visible domain for performance gains. We observe a significant drop in performance by testing the IDD and KITTI model on FLIR thermal images - 2.6x drop and 13x drop, respectively. This shows that a model trained in visible domain does not infer well in another domain due to the inherent difference of visual representations. In the case of inference on RGB domain itself we can observe a drop of 1.6x and 6.2x respectively from the baseline performance on the same dataset.

V. CONCLUSIONS

TABLE III

BASELINE RESULTS FROM TRAINING OBJECT DETECTION MODEL ON THE THREE DATASETS

Train Dataset	Test Dataset	Bicycle	Car	Dog	Person	mAP
IDD	FLR_RGB	0.192	0.473	0.052	0.339	0.264
IDD	FLR.THM	0.126	0.265	0.099	0.160	0.163
IDD	IDD	0.569	0.617	0.070	0.448	0.426
KITTI	FLR_RGB	-	0	-	0.316	0.158
KITTI	FLR.THM	-	0	-	0.141	0.070
KITTI	KITTI	-	0.970	-	0.899	0.935

From our experiments in Table 2, 3 we can conclude that there is no domain transfer from a model trained in the visible spectrum to inferences in the thermal domain. Thus thermal imagers can prove to be a valuable addition to object detection pipelines, especially for robustness of systems like autonomous vehicles. Results in Table 3 also show that few shot training on the Faster-RCNN model from a previously trained model does not perform well across the domains and on new datasets.

VI. FUTURE WORK

Further investigations to evaluate the effect of fusion strategies in the Faster-RCNN network is ongoing. We would also like to compare the effect of multiple fusion strategies with the baseline performance.

REFERENCES

- [1] S. Agarwal and D. Roth. Learning a sparse representation for object detection. In Proc. ECCV, page IV: 113 ff., 2002.
- [2] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid, "Local features and kernels for classification of texture and object categories: A comprehensive study," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2006.
- [3] A. Opelt, A. Pinz, and A. Zisserman, "Learning an alphabet of shape and appearance for multi-class object detection," International Journal of Computer Vision, vol. 80, pp. 16-44, 2008.
- [4] Z. Si, H. Gong, Y. N. Wu, and S. C. Zhu, "Learning mixed templates for object recognition," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2009.
- [5] J. Shotton, "Contour and texture for visual recognition of object categories," Doctoral of Philosophy, Queen's College, University of Cambridge, Cambridge, 2007.
- [6] J. Shotton, A. Blake, and R. Cipolla, "Multiscale categorical object recognition using contour fragments," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 30, pp. 1270-1281, 2008.
- [7] K. Schindler and D. Suter, "Object detection by global contour shape," Pattern Recognition, vol. 41, 2008.
- [8] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results," <http://www.pascalnetwork.org/challenges/VOC/voc2007/workshop/index.html>
- [9] T.-Y. Lin et al., "Microsoft COCO: Common objects in context," in Proc. Eur. Conf. Computer Vision, Springer, 2014, pp. 740755
- [10] S. Gupte, O. Masoud, R. F. K. Martin, and N. P. Papanikolopoulos, "Detection and classification of vehicles," IEEE Transactions on Intelligent Transportation Systems, 3(1):3747, Mar. 2002.
- [11] <https://www.flir.in/oem/adas/adas-dataset-form/>
- [12] S. Hwang, J. Park, N. Kim, Y. Choi, and I. S. Kweon, "Multispectral pedestrian detection: Benchmark dataset and baseline," in Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), Jun. 2015.
- [13] N. Dalal and B. Triggs. "Histograms of Oriented Gradients for Human Detection," In CVPR, 2005
- [14] P. Viola and M. Jones, "Robust Real-time Face Detection", IJCV, 57(2), 2004
- [15] P. Dollar, R. Appel, S. Belongie, and P. Perona, "Fast Feature Pyramids for Object Detection," TPAMI, 36(8), 2014
- [16] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in Proc. IEEE Conf. Computer Vision and Pattern Recognition, 2014
- [17] R. Girshick, "Fast R-CNN," in Proc. IEEE Conf. Computer Vision, 2015, pp. 14401448
- [18] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards realtime object detection with region proposal networks," in Advances in Neural Information Processing Systems, 2015, pp. 9199.J.
- [19] Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), 2016, pp. 779788.
- [20] W.Liu, D.Angeuelov, D.Erhan, C.Szegedy, S.Reed, C-Y.Fu and A.C.Berg, "SSD: Single shot multibox detector," in Proc. Eur. Conf. Computer Vision. Springer, 2016, pp. 2137
- [21] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," Int. J. Robot. Res., vol. 32, no. 11, pp. 12311237, 2013.
- [22] M. Cordts et al., "The cityscapes dataset for semantic urban scene understanding," in Proc. IEEE Conf. Computer Vision Pattern Recognition (CVPR), Jun. 2015, pp. 32133223.
- [23] J. Ge, Y. Luo, and G. Tei. Real-Time Pedestrian Detection and Tracking at Nighttime for Driver-Assistance Systems. IEEE Transactions on ITS, 10(2), 2009.
- [24] Y. Lee, Y. Chan, L. Fu, and P. Hsiao. Near-Infrared-Based Nighttime Pedestrian Detection Using Grouped Part Models. IEEE Transactions on ITS, 16(4), 2015.
- [25] R. Miezanko and D. Pokrajac. People Detection in Low Resolution Infrared Videos. In CVPR Workshops, 2008
- [26] M. Teutsch, T. Muller, M. Huber, and J. Beyerer. Low resolution person detection with a moving thermal infrared camera by hot spot classification. In CVPR Workshops, 2014.
- [27] G. Varma, A. Subramanian, A. Namboodiri, M. Chandraker and C V Jawahar, "IDD: A Dataset for Exploring Problems of Autonomous Navigation in Unconstrained Environments", in IEEE Winter Conf. on Applications of Computer Vision (WACV 2019)

Exploration for Objects Labelling Guided by Environment Semantics using UAVs

Reem Ashour¹, Tarek Taha², Jorge Dias¹, Lakmal Seneviratne¹, and Nawaf Almoosa¹

Abstract— This paper presents an efficient autonomous exploration strategy of unknown indoor environments. This strategy focuses on 3D mapping the environment and performing a grid level semantic labeling in order to identify all objects. Unlike conventional exploration techniques that utilize geometric heuristics and information gain theory on occupancy grid maps, the work presented in this paper considers semantic information, such as the class of objects, to gear the exploration towards environment segmentation and objects labeling. The proposed approach utilizes deep learning to map 2D semantically segmented images into 3D semantic point clouds that encapsulate both occupancy and semantic annotations. A Rapidly-exploring Random Tree (RRT) algorithm with the proposed semantic cost functions is employed to iteratively evaluate the global map to label all the objects in the environment using a novel utility function that balances exploration and objects labeling. The proposed strategy was evaluated in a realistic simulated indoor environment, and results were benchmarked against other exploration strategies.

I. INTRODUCTION

The growth in aerial robotics has led to their ubiquitous presence in various fields such as urban search and rescue (USAR) [1], infrastructure inspection [2] and surveillance [3], etc. The completeness and efficiency of mapping and exploration processes are crucial to facilitate these applications. Some of the recent research has been focused on rescue and rescue activities performed by Unmanned Aerial Vehicle (UAV). These robots assist rescue team in the form of vital information on time-sensitive situations without endangering human lives. Autonomous capabilities such as exploration and mapping of unknown environments are crucial to provide rescuers with richly reconstructed mapped environments and increase their understanding of the situation.

In this work, an efficient autonomous semantic-aware 3D mapping and exploration method for unknown indoor is proposed by utilizing semantic information encapsulated in

¹Reem Ashour, Jorge Dias and Lakmal Seneviratne are with Khalifa University of Science and Technology, Abu Dhabi, UAE, Emails: reem.ashour@ku.ac.ae, jorge.dias@ku.ac.ae, lakmal.seneviratne@ku.ac.ae and nawaf.almoosa@ku.ac.ae

²Tarek Taha, is with Algorythma's Autonomous Aerial Lab, Abu Dhabi, UAE, email: tarek.taha@algorythma.com

the proposed 3D semantic-aware map for object localization and labeling using a UAV with an onboard RGBD camera. New utility functions are proposed to direct the robot towards the objects in the environment. The results show that the model is capable of exploring unknown environments and label objects effectively.

II. MAPPING AND INFORMATION QUANTIFICATION

The procedure for the mapping and information quantification is provided in Fig. 1. It involves three stages: object detection, annotated point cloud generation, and the 3D semantic-aware mapping. In object detection stage, semantic segments are generated for objects found in a 2D image frame. The deep neural network Pyramid Scene Parsing Network (PSPNet) based on semantic segmentation [4] is employed to provide semantic segments for the different objects in 2D images. After that, the point cloud captured from the depth camera is annotated to the corresponding class from the deep learning model output. This is performed by firstly registering the depth with the same reference frame that the color image is registered, which usually the camera frame. After that, transforming the pixels from the camera frame to the real world frame using the image position, its depth, and the intrinsic camera parameters to form the point cloud. In the last stage, a 3D occupied semantic map structure based on an occupancy grid map the octomap [5] is proposed. The map $M=\{m_1, m_2, \dots, m_i\}$ consists of the cubical elements of the same size where m_i is a voxel for index i . Each voxel m_i encapsulates volumetric information and semantic information, which are the semantic color, confidence value, and the number of visits.

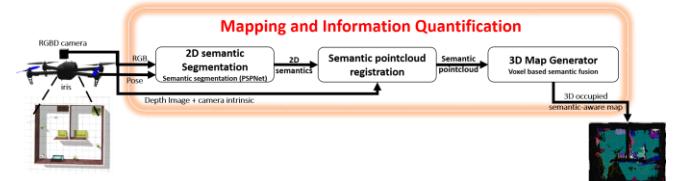


Figure 1. Proposed Semantic-aware Exploration and Mapping System Architecture

III. SEMANTIC-AWARE EXPLORATION

The proposed exploration strategy provides the robot with the ability to explore unknown environments, while simultaneously optimizing information gathering and directing the robot to label objects in the environment. To

enable this functionality, two new multi-objective utility functions are proposed to account for the semantic information (confidence or number of visits) that each voxel encapsulates. The proposed system used quantified information from the semantic-occupied map to evaluate the next best exploration action.

A remarkable technique used to explore an unknown environment is the Next Best View (NBV) [6] approach. The main steps in the exploration task are: A) viewpoint sampling, B) viewpoint evaluation, C) navigating toward the selected viewpoint, and D) termination. The exploration process is summarized in Fig. 2. At the beginning of the exploration process, a robot uses the onboard sensors to observe the scene and produce a set of viewpoints candidates (also known as states or poses). In this work, the Rapidly-Exploring Random Tree (RRT) [7] is used. The RRT selects a series of points randomly in a tree-like manner instead of multiple single points for evaluation. The tree is expanded throughout all the exploration space, and each branch forms a group of random branches. The accepted viewpoints candidates are then evaluated using a utility function (also known as reward, cost, or heuristic function). In this work, each point in the branch is evaluated using a utility function, and the branch which maximizes the utility function is selected as the next goal. Although the evaluation is performed for the whole branch, and the best branch is selected for execution, only the first edge of the selected branch is executed. The exploration process repeats in a receding horizon fashion until a termination condition is met, indicating the end of the exploration. In this work, the termination goal used is a predefined number of iteration.

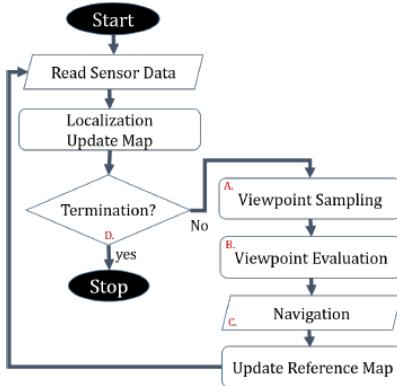


Figure 2. General components of the NBV method

IV. EXPERIMENTAL SETUP

Simulation experiments performed on an ASUS laptop (Intel Core i7 @ 2.8 GHz x 8, 16 GB RAM). The NBV framework was implemented on Ubuntu 16.04 using the Robot Operating System (ROS-kinetic) [8] to handle message sharing and ease the transition to hardware. The gazebo simulator was used to perform the actual simulation, with programming done in both C++ and Python. The simulation were performed using a UAV equipped with one RGB-D camera. The environment, shown in Fig. 3, was designed via gazebo and used as the unknown environment that the robot

should explore. The robot position is assumed to be perfectly known. The simulation environment has multi-connected rooms with a corridor with several objects placed inside the rooms. The environment contains 11 objects which are walls, floors, three people, a sofa, two chairs, a book shelf, a vase, and a table. The constructed maps are based on 3D occupancy grid using OctoMap library with res = 0.15m per pixel. Each utility function is tested separately under controlled environments.

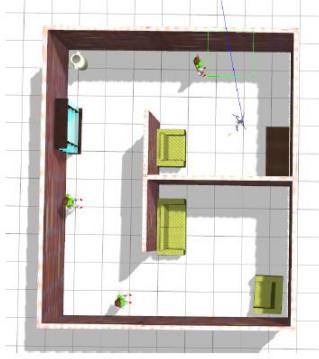


Figure 3: Simulation Environment

V. EVALUATION METRICS

Table I summarizes the evaluation metrics used.

Table I: Evaluation Metrics

Coverage	Percentage of the number of known voxels compared with the total number of voxels the environment can cover. After each iteration, the coverage is calculated as follows $VC = \frac{Free + Occupied}{Free + Occupied + Unknown}$
Detected objects	Counting the number of detected objects in the environment
Efficiently labeled objects	Counting manually the number of objects that are correctly labeled using the semantic color table

VI. EXPERIMENTAL RESULTS

The two proposed utility functions are compared with the state of the art volumetric gain [9]. The reported results are for three different experiments simulation tests. The tests are divided according to the viewpoint evaluation approaches. Table II shows the recorded values for the evaluation metrics. The reconstructed 3D semantic maps are shown in Fig. 4, Fig. 5, and Fig. 6 when using the volumetric gain, semantic visible voxel, and semantic visited voxels for objects of interest respectively. Figure 7 shows the semantic annotations.

Table II : Evaluation Results

Num	VS	Viewpoint Evaluation	VC(%)	NDO	NSDO
1	RRT	Volumetric Gain [9]	91.3 %	11	8
2		Semantic Visible Voxels	88.5 %	11	7
3		Semantic Visited Object	93.1 %	11	8

Viewpoint Sampling (VS), Volumetric Coverage (VC), Number of Detected Object (NDO), Number of Sufficiently Detected Objects (NSDO)



Figure 4: 3D Map Using Volumetric Gain Utility Function



Figure 5: 3D Map Using Semantic Visible Voxels



Figure 6: 3D map using Semantic Visited Voxel for Object of Interest

wall	table	stairs
floor	chair	stairway
building	sofa	coffee table
person	shelf	bench
ceiling	seat	step
grass	box	sculpture
sidewalk	grandstand	vase

Figure 7: Color annotations map

ACKNOWLEDGMENT

This publication is based upon work supported by the Khalifa University of Science and Technology under Award No. RC1-2018-KUCARS.

REFERENCES

- [1] M. Erdelj, E. Natalizio, K. R. Chowdhury, and I. F. Akyildiz, “Help from the sky: Leveraging uavs for disaster management,” IEEE Pervasive Computing, vol. 16, no. 1, pp. 24–32, 2017.
- [2] N. Hallermann and G. Morgenthal, “Visual inspection strategies for large bridges using unmanned aerial vehicles (uav),” in Proc. of 7th IABMAS, International Conference on Bridge Maintenance, Safety and Management, 2014, pp. 661–667.
- [3] A. Wada, T. Yamashita, M. Maruyama, T. Arai, H. Adachi, and H. Tsuji, “A surveillance system using small unmanned aerial vehicle (uav) related technologies,” NEC Technical Journal, vol. 8, no. 1, pp. 68–72, 2015.
- [4] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, “Pyramid scene parsing network,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 2881–2890.
- [5] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, “Octomap: An efficient probabilistic 3d mapping framework based on octrees,” Autonomous Robots, vol. 34, no. 3, pp. 189–206, 2013.
- [6] C. Connolly, “The determination of next best views,” in Robotics and automation. Proceedings. 1985 IEEE international conference on, vol. 2. IEEE, 1985, pp. 432–435.
- [7] S. Karaman and E. Frazzoli, “Sampling-based algorithms for optimal motion planning,” The international journal of robotics research, vol. 30, no. 7, pp. 846–894, 2011.
- [8] Koubâa, A. Robot Operating System (ROS). Springer, 2011
- [9] A. Birch, M. Kamel, K. Alexis, H. Olynikova, and R. Siegwart, “Receding horizon” next-best-view” planner for 3d exploration,” in Robotics and Automation (ICRA), 2016 IEEE International Conference on. IEEE, 2016, pp. 1462–1468

Towards game theoretic AV controllers: measuring pedestrian behaviour in Virtual Reality

Fanta Camara^{1,2}, Patrick Dickinson², Natasha Merat¹ and Charles W. Fox^{1,2,3}

Abstract—Understanding pedestrian interaction is of great importance for autonomous vehicles (AVs). The present study investigates pedestrian behaviour during crossing scenarios with an autonomous vehicle using Virtual Reality. The self-driving car is driven by a game theoretic controller which adapts its driving style to pedestrian crossing behaviour. We found that subjects value collision avoidance about 8 times more than saving 0.02 seconds. A previous lab study found time saving to be more important than collision avoidance in a highly unrealistic board game style version of the game. The present result suggests that the VR simulation reproduces real world road-crossings better than the lab study and provides a reliable test-bed for the development of game theoretic models for AVs.

Keywords: Autonomous Vehicles; Game Theory; Cognitive architectures for action selection; Pedestrian Behaviour;

I. INTRODUCTION

The upcoming arrival of autonomous vehicles on the roads poses several concerns regarding their future interaction with other road users, in particular with pedestrians and cyclists, whose behaviour is more complex and unpredictable. Pedestrian interaction is challenging due to multiple uncertainties in their pose estimation, gestures and intention recognition. We thus recently proposed a game theory model for such interactions [3], where a pedestrian encounters an autonomous vehicle at an unsignalized intersection.

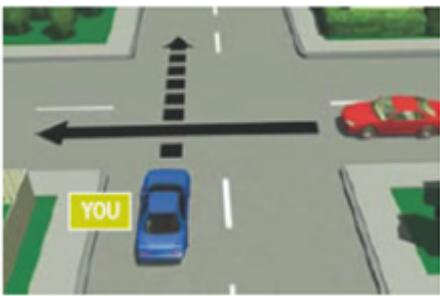


Fig. 1: Two agents negotiating for priority at an intersection

In this model, two agents (e.g. pedestrian and/or human or autonomous driver) called Y and X are driving straight

towards each other at an unmarked intersection as in Fig. 1. In the model, this process occurs over discrete space as in Fig. 2 and discrete times ('turns') during which the agents can adjust their discrete speeds, simultaneously selecting speeds of either 1 square per turn or 2 squares per turn, at each turn. Both agents want to pass the intersection as soon as possible to avoid travel delays, but if they collide, they are both bigger losers as they both receive a negative utility U_{crash} . Otherwise if the players pass the intersection, each receives a time delay penalty $-TU_T$, where T is the time from the start of the game and U_T represents the value of saving one turn of travel time.

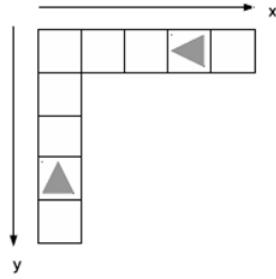


Fig. 2: Sequential Chicken Game

The model assumes that the two players choose their actions (speeds) $a_Y, a_X \in \{1, 2\}$ simultaneously then implement them simultaneously, at each of several discrete-time turns. There is no lateral motion (positioning within the lanes of the roads) or communication between the agents other than via their visible positions. The game is symmetric, as both players are assumed to know that they have the same utility functions (U_{crash}, U_T), hence they both have the same optimal strategies. These optimal strategies are derivable from game theory together with meta-strategy convergence, via recursion. Sequential Chicken can be viewed as a sequence of one-shot sub-games, whose payoffs are the expected values of new games resulting from the actions, and are solvable by standard game theory.

The (discretized) locations of the players can be represented by (y, x, t) at turn t and their actions $a_Y, a_X \in \{1, 2\}$ for speed selection. The new state at turn $t + 1$ is given by $(y + a_Y, x + a_X, t + 1)$. Define $v_{y,x,t} = (v_{y,x,t}^Y, v_{y,x,t}^X)$ as the value (expected utility, assuming all players play optimally) of the game for state (y, x, t) . As in standard game theory the value

¹ Institute for Transport Studies (ITS), University of Leeds, UK

² Lincoln Centre for Autonomous Systems (LCAS), School of Computer Science, University of Lincoln, UK

³ Ibex Automation Ltd, UK

This project has received funding from the European Union's Horizon 2020 Research and Innovation programme under grant agreement No 723395 (InterACT)

of each 2×2 payoff matrix can then be written as,

$$v_{y,x,t} = v\left(\begin{bmatrix} v(y-1, x-1, t+1) & v(y-1, x-2, t+1) \\ v(y-2, x-1, t+1) & v(y-2, x-2, t+1) \end{bmatrix}\right), \quad (1)$$

which can be solved using dynamic programming assuming meta-strategy convergence equilibrium selection. Under some approximations based on the temporal gauge invariance described in [3], we may remove the dependencies on the time t in our implementation so that only the locations (y, x) are required in computation of $v_{y,x}$ and optimal strategy selection.

Virtual Reality (VR) offers the opportunity to experiment on human behaviour in simulated real world environments that can be dangerous or difficult to study, such as pedestrian road crossing. The present study uses VR to run the game theoretic model on a virtual autonomous vehicle and then examines the responses of human participants to that.

Contributions: To our best knowledge, this is the first attempt to evaluate pedestrian behaviour during interaction scenarios with a game theoretic autonomous vehicle in a virtual reality environment. It examines pedestrian road-crossing preferences (U_{crash}, U_T) when interacting with the virtual autonomous vehicle and demonstrates the importance of VR for the development of the model.

II. RELATED WORK

There are few previous studies which investigated on interactions between autonomous vehicles and other road users in VR. Wang et al. [7] developed 5 different behaviours for an autonomous vehicle. The vehicle behaviour was successfully tested in different simulated traffic scenarios such as at intersections and lane changing, in a simulated city and highway road networks. Keferböck et al. [5] studied autonomous vehicles interactions with pedestrians in a virtual environment. In one of their experiments, participants are asked to cross a road in front of them while a car is approaching. This experiment differs from ours in that the AV stops and shows (or not) a stop intent to pedestrians. This study aimed to show the importance of substituting communications between pedestrians and drivers by some explicit communication forms for self-driving cars. Pillai [6] used task analysis to divide pedestrian-vehicle interaction as a sequence of actions giving two outcomes, either the vehicle passes first or the pedestrian crosses and perform some experiments with participants on their crossing behavior using virtual reality. Hartmann et al. [4] proposed a testing procedure of pedestrian collision avoidance for autonomous vehicles using VR techniques. This test bed can take into account different factors that could influence pedestrian behaviour such as their understanding of the environment, their body movement and their personality.

We previously performed laboratory experiments to fit data to the game theory model [3]. We first asked participants to play this game as a board game in [2]. Secondly, participants were asked to play the game in person moving on squares [1]. These previous laboratory experiments have shown unrealistic results, participants preferring time saving rather than

collision avoidance. The present study aims to extend these experiments and put participants in more realistic interaction scenarios with a game theoretic autonomous vehicle in a virtual environment.

III. METHODS

A. VR Setup

The study was conducted using an HTC Vice Pro head mounted display (HMD). Participants did not use the HTC Vice controllers, as no interactions other than walking were required. The HMD was used with the HTC wireless adapter in order to facilitate easier movement during the simulation. We used an area of approximately 6m by 3m to conduct the simulation (as shown in Fig. 3), which was mapped using the usual HTC Vive room mapping system. The size of this area slightly exceeds that recommended by the manufacturer; however, we experienced no technical problem with tracking or system performance. The start position on the floor was marked with an 'X' using floor tape, so that participants knew where to stand at the start of each simulation, prior to placing the HMD on their head. The simulation was created using the Unity 3D engine, and was run under Windows 10 on a PC based on an Intel Core i7-7700K CPU, with 32GB of RAM, and an Nvidia GeForce GTX 1080 GPU.

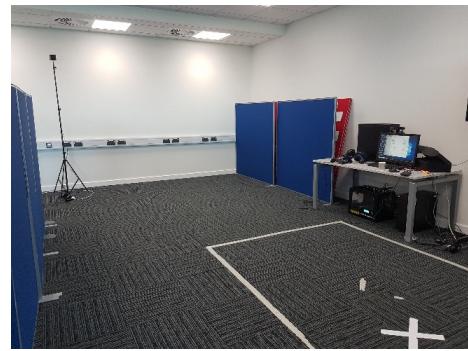


Fig. 3: VR Lab

B. Car behaviour model

The virtual AV was designed to drive using the Sequential Chicken model described above. The car began driving 40 meters away from the intersection, its full speed was $30km/h$ and lowest speed was $15km/h$. The vehicle moved and adapted its behaviour to participants motion. Every 0.02s, the car observed the current position of the pedestrian and made its decision based on the game theory model. The car was designed not to stop for any pedestrian. Indeed, in the sequential chicken model, if the two players play optimally, then there must exist a non-zero probability for a collision to occur. Intuitively, if we consider an AV to be one player that always yields, it will make no progress as the other player will always take advantage over it, hence there must be some threat of collision.



Fig. 4: Virtual Autonomous Vehicle

C. Human experiment

We invited 11 participants, 10 males and 1 female aged between 19 and 37 years old, to take part to the study, under University of Lincoln Research Ethics. 7 participants had previous experience with VR. Participants were asked to cross a road in front of them as they would do in everyday life. They should stop moving on their other side of the road, when they reached a yellow cube, located there for safety reasons. A vehicle approaches from their right hand side. Participants began walking about 4 meters away from the intersection. Prior to the experiment, participants were introduced to the experimental setup and trained on walking within the VR environment with the VR headset. There were 6 trials per participants in the virtual environment with the first trials considered as training data.



Fig. 5: Participant taking part in the study

IV. RESULTS

In total, 55 pedestrian-vehicle interactions were recorded. Among those interactions, pedestrians managed to cross the road before the car reached the intersection only 9 times. These crossings happened after the first trials, by pedestrians who felt confident after evaluating/gauging the car driving style. Most interactions looked similar to Fig. 6, which shows the trajectories of a participant and the autonomous vehicle during one interaction. The trajectory profile shows that pedestrians were slowing down very quickly after seeing the car, they were not playing optimally the game of chicken, so that the AV could cross most of the time.

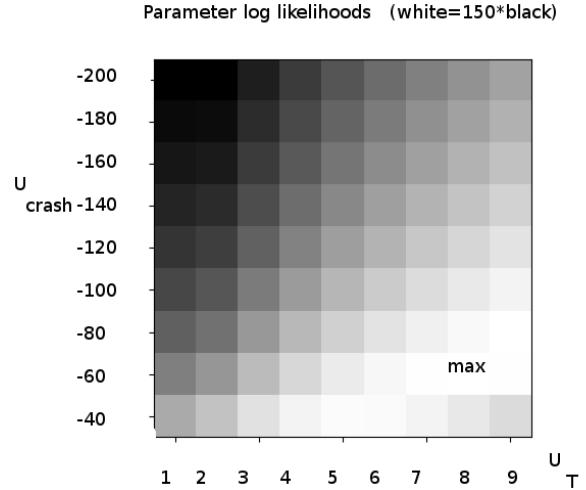


Fig. 7: Pedestrian behaviour preference

Similar to the optimal solution computation method developed in the laboratory experiments [2] [1], we obtain an optimal parameter, $\theta = U_{crash}/U_T = -60/8 = 7.5$, for participants, as shown in Fig. 7. This reveals that pedestrians valued avoidance of a crash 8 times more than a 0.02s time saving per turn, resulting in pedestrians being less assertive in crossing the road. In comparison, previous laboratory experiments found that participants valued time saving more than collision avoidance [2][1].

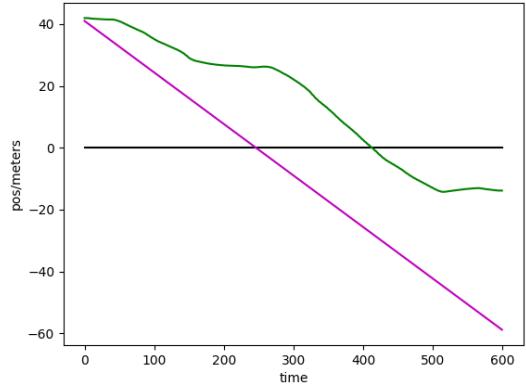


Fig. 6: Example of pedestrian and AV trajectories (magenta: AV; green: pedestrian)

V. CONCLUSION

The present study demonstrated a work-in progress on the use of virtual reality for the development of game theoretic AV controllers. We examined the trajectories of pedestrians interacting with a virtual autonomous vehicle which makes its decisions based on the sequential chicken model. The results reveal that pedestrian behaviour is more natural in VR than in previous laboratory experiments. This is important, as it shows that virtual reality makes pedestrian crossing behaviour more realistic and it can therefore help improve the development of the game theoretic model. Future work would include the evaluation of pedestrian

crossing behaviours with different car models and within different environments. Methods of learning the best behaviour parameters for the autonomous vehicle will be explored in future VR studies.

REFERENCES

- [1] F. Camara, S. Cosar, N. Bellotto, N. Merat, and C. W. Fox. Towards pedestrian-av interaction: method for elucidating pedestrian preferences. In *IEEE/RSJ Intelligent Robots and Systems (IROS) Workshops*, 2018.
- [2] F. Camara, R. Romano, G. Markkula, R. Madigan, N. Merat, and C. W. Fox. Empirical game theory of pedestrian interaction for autonomous vehicles. In *Measuring Behavior 2018: 11th International Conference on Methods and Techniques in Behavioral Research*. Manchester Metropolitan University, March 2018.
- [3] C. W. Fox, F. Camara, G. Markkula, R. Romano, R. Madigan, and N. Merat. When should the chicken cross the road?: Game theory for autonomous vehicle - human interactions. In *Proceedings of VEHITS 2018: 4th International Conference on Vehicle Technology and Intelligent Transport Systems*, January 2018.
- [4] M. Hartmann, M. Viehweger, W. Desmet, M. Stolz, and D. Watzenig. Pedestrian in the loop: An approach using virtual reality. In *International Conference on Information, Communication and Automation Technologies (ICAT)*, 2017.
- [5] F. Keferböck and A. Riener. Strategies for negotiation between autonomous vehicles and pedestrians. In D. G. Oldenbourg, editor, *Mensch und Computer 2015 –Workshopband*, 2015.
- [6] A. Pillai. Virtual reality based study to analyse pedestrian attitude towards autonomous vehicles. Master's thesis, 10 2017.
- [7] H. Wang, J. K. Kearney, J. Cremer, and P. Willemsen. Steering behaviors for autonomous vehicles in virtual environments. In *IEEE Virtual Reality*, 2005.

MSPRT action selection model for bio-inspired autonomous driving and intention prediction

Riccardo Donà¹, Gastone Pietro Rosati Papini¹ and Giammarco Valenti¹

Abstract—This paper proposes the usage of a bio-inspired action selection mechanism, known as multi-hypothesis sequential probability ratio test (MSPRT), as a decision making tool in the field of autonomous driving. The focus is to investigate the capability of the MSPRT algorithm to effectively select the optimal action whenever the autonomous agent is required to drive the vehicle or, to infer the human driver intention when the agent is acting as an intention prediction mechanism. After a brief introduction to the agent, we present numerical simulations to demonstrate how simple action selection mechanisms may fail to deal with noisy measurements while the MSPRT provides the robustness needed for the agent implementation on the real vehicle.

I. INTRODUCTION

Autonomous vehicles (AVs) require effective algorithms to perform robust decision making in the shortest time frame possible. Indeed, in a dynamic environment such as the one faced by the AVs, the capability of reacting promptly is a major factor in potentially avoiding collisions and saving lives. The inherent complexity of the process is worsened by the presence of sensors' noise and uncertainties which affect the way the behavioural level selects the proper action. In the early days of autonomous driving, tactical/behavioral level planning typically relied on manually engineered state machines, this approach has been adopted by many competitors of the 2007 DARPA Grand Challenge (a.k.a. *urban challenge*) [1], [2]. Despite some participants actually managed to succeed, state machines inherently lack the capability of safely generalizing to unmodeled scenarios. More recent autonomous driving softwares are built on top of probabilistic approaches including Markov Decision Process [3] or machine learning-based techniques such as behaviour networks [4] or support vector machines [5]. A promising method is the adoption of reinforcement learning (RL) as a high level biasing mechanism for learning an optimal action selection policy [6] or oppositely, the exploitation of the inverse reinforcement learning (IRL) framework to learn the reward function from human data [7].

Conversely, the problem of action selection is not a peculiar feature of AVs, instead any agent (both artificial and biological) dealing with complex dynamical environments where multiple mutually exclusive behaviours are possible, shares similar dilemmas. Indeed there exists a huge amount of ethology literature investigating “behaviour switching” and “decision making” [8], the common jargon among cognitive scientists to refer to the action selection problem in robotics.

¹ Department of Industrial Engineering, University of Trento, 38123 Trento, Italy riccardo.dona@unitn.it

Several theories have been proposed in literature on how animals perform effective decision making [9]. For instance, in [10] the *affordance competition* concept underlines a parallel processing of multiple actions competing against each other until the selection of the winning behavior. Such a modeling framework is based on the definition of criteria for assessing the *worthiness* of the action and the *selection* process itself.

We exploit this concept of parallel competing actions in the context of the European Projects SafeStrip¹ and Dreams4Cars². In particular, in SafeStrip we take advantage of the mirroring mechanism introduced in [11] to infer the human driver intended action in several dangerous scenarios, like in the proximity of a pedestrian crossing, in a road work zone or in an intersection. In the latter case a more complex mirroring is performed, taking into account the right of way rules and mirroring other vehicles. This is made through vehicle to vehicle and vehicle to infrastructure communication [12].

Such an inference process boils down to the selection among a set of longitudinal maneuvers, called *motor primitives*, of the one matching the driver intended action in terms of instantaneous jerk j_0 . Each motor primitive has an optimality-based formulation characterized by an initial jerk associated with. By defining the jerk space as a one 1-dimensional grid we can explore a set of possible actions taking also into account infrastructure-based information.

In Dreams4Cars we utilize a similar optimality-based motor primitives approach for the synthesis of an autonomous driving agent called Co-driver [13]. In addition to the longitudinal manoeuvres, we also generate set of lateral manoeuvres by defining a 1-dimensional grid on instantaneous lateral jerk r_0 . By combining the two grids we devise a 2-dimensional matrix where each entry is a pair of (j_0, r_0) which encodes a latent action. Each pair is then assigned a merit via the definition of a scenario dependent *salience*.

Common to both the project there is the need to select the best action after the computations of the grids. The rest of this paper is devoted to demonstrate how we can perform such a task taking advantage of a biologically inspired action selection mechanism.

II. THE MOTOR CORTEX CONCEPT

In order to better clarify how the affordances competition process takes place, let us inspect an example simulation

¹<https://www.safestrip.eu>

²<https://www.dreams4cars.eu>

scenario as in Fig. 1. In the proposed situation the ego car, driven by the Co-driver agent, is travelling at high speed on a straight road when a slower vehicle is detected (Fig. 1a). This scenario translates into the control space representation shown in Fig. 1b. Physical space to control space transformation is performed via the analytical solution of a linearized vehicle kinematic plant optimal control similarly to [14], [13]. The green portion of the control space representation expresses the feasible control actions, *i.e.* the set of pairs (j_0, r_0) which allow the ego car to stay within the solid lane markings. On the other side, the orange/yellow portion conveys the control inhibition caused by the presence of a slower leading vehicle. The solid orange region is associated to controls that lead to a collision while the yellow area encodes the potential danger in staying too close to the obstacle. Eventually the white region expresses the speed limit exceedance.

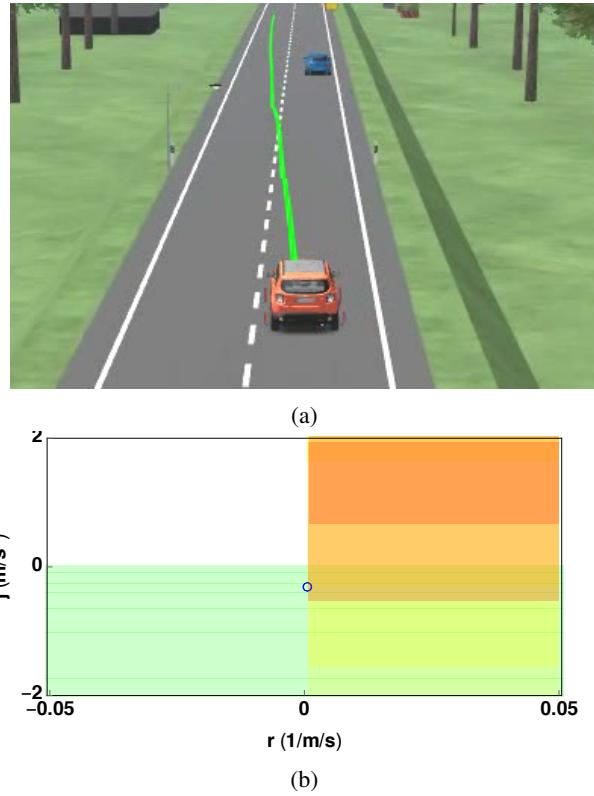


Fig. 1: Example simulation scenario bird-eye view (a) and corresponding control space representation (b).

The motor cortex corresponding to the action space in Fig. 1b can be computed by introducing some merit criterion. For the considered example scenario we model the merit as the maximum time at which, given the pair (j_0, r_0) , the vehicle will leave the road or collide with other road users. In other words we are trying to find which are the controls that allow the vehicle to navigate the longest without any further intervention during the execution. This idea is also known as *minimum intervention principle* [15]. Given the biological inspiration of the procedure, we refer to such a time as the *salience* of the action. By establishing the criterion above, we

can compute an artificial *motor cortex* as in Fig. 2, where the salience is displayed along the z -axis of the 3D plot. It can be noticed how lateral controls close to zero have high merit values as, clearly, steering abruptly will drive the vehicle out of the road sooner than steering mildly while the orange region in Fig. 1b has a close to zero salience due to the inherent risk of collide in a short time frame.

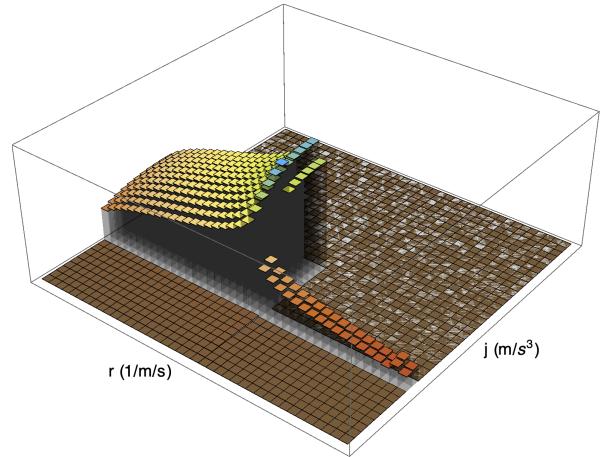


Fig. 2: Minimum intervention principle based motor cortex for the scenario in Fig. 1.

The motor cortex in Fig. 2 encodes the *affordance* concepts previously mentioned. Each of the action is in fact associated to a merit value and compete against the others for winning the selection process. The outcome of the “competition” is the optimal pair (j_0^*, r_0^*) that will eventually guide the car for the next time-step.

In the inference-via-mirroring application, the merit assignment procedure is slightly modified to account for both the potential maneuvers and the one currently performed by the driver. After the computation of the scenario-based merit for each initial control, a bias function measuring the proximity of the driver maneuver to each action is applied to the motor cortex as shown in [16] for the longitudinal control only.

III. ACTION SELECTION

A. WTA algorithm

The most trivial approach to model the affordances competition would be to simply choose the pair having the highest instantaneous salience. This selection mechanism is known as *winner takes all* (WTA) [9] and has proven to be fairly efficient in the simulation environment where there is no signal noise. On the other side, this action selection procedure is likely to choose sub-optimal action in the presence of noise such as when the agent is driving a real car. Furthermore, even in the simulation environment, this mechanism may give rise to hysteresis when two competing actions share similar salience values which can cause loss of vehicle control.

B. MSPRT algorithm

In order to overcome the problematics of the WTA procedure we propose here the introduction of the multi-hypotheses sequential probability ratio test (MSPRT) [17] decision making algorithm.

The key idea of the MSPRT algorithm is to accumulate *evidence* for each channel and then pick an action only when the integral reaches a predefined *threshold* level. This mechanism should guarantee more robustness to noisy decisions by trading off some responsiveness.

The MSPRT has been shown to be asymptotically time-optimal in a multi alternatives process [18]. More recently a link between the action selection process happening in the basal ganglia of the human brain and the MSPRT algorithm has been drawn [19].

The overall procedure for action-selection using the MSPRT algorithm is reported in Algorithm 1. First we append the set of observations at time-step \mathcal{M}_t to the list of observations $\mathcal{M}_{\text{list}}$ which contains observations from $t - ws$ up to t where ws is the dimension of the averaging window. Then we compute the mean of observation along the time dimension in order to get an average of each channel evidence for the considered window. Next we compute the likelihood of each channel according to

$$L(t) = y(t) - \log \sum_{k=1}^N \exp(y_k(t)), \quad (1)$$

where $y(t)$ represents the vector of evidence at time-step t obtained via flattening the motor cortex. Then we compute the $\max(\exp(L))$ to investigate whether some of the channels reached a predefined threshold value. In the positive case we reset the moving average list by taking only a percentage λ of the current average value. Otherwise we continue to follow the previous action until eventually a new action will win the affordance race.

Algorithm 1: MSPRT algorithm

```

Result: Action log-likelihood
 $\mathcal{M}_{\text{list}} \leftarrow \mathcal{M}_t;$ 
 $\bar{\mathcal{M}} \leftarrow \text{mean}\{\mathcal{M}_{\text{list}}\};$ 
compute likelihood  $L$  as in (1);
if  $\max(\exp(L)) > \text{threshold}$  then
    | take action;
    |  $\mathcal{M}_{\text{list}} = \lambda \bar{\mathcal{M}}$ 
else
    | follow previous action;
end
```

Overall the behaviour of the MSPRT algorithm can be shaped by adjusting the hyper-parameters in Table I.

IV. SIMULATION COMPARISON

We compare the performances of the MSPRT against the WTA on simulated logged data. Firstly we let the agent drove on a simulated scenario with no noise affecting the

TABLE I: Parameters of the MSPRT algorithm.

name	symbol	value	effect
threshold	th	0.0005	slows down the switch to a new channel
windows size	ws	8	average out noise, brings in more robustness
forgetting factor	λ	0.9	introduces a memory effect after the switching to a new channel

measurements. According to this set-up we can perform optimal decision making using a simple WTA algorithm. We then select a 9-seconds long critical double lane change maneuver where the responsiveness of the action selection plays a fundamental role. Next, we re-execute the simulation offline, *i.e.* we take the logged motor cortex history, we apply some random noise on the channels and we re-execute the decision making algorithm only on the corrupted motor cortex. We then analyze again the performances of the WTA against MSPRT with respect to the ground-truth case obtained previously. The exact parameters used in the simulation above are reported in Table I.

Fig. 3 reports the results of assessment as a function of the adimensional noise variance σ injected into the motor cortex. In case of limited noise figures, the WTA still outperforms MSPRT due to the worse transient performance of the latter. As soon as we introduce noise in the simulation, however, the advantages of the MSPRT start to be evident. In this case we chose a fairly conservative tuning for the MSPRT that will make it behave correctly even in the presence of high noise while the performance of the WTA drops in a more significant manner. Indeed by shrinking the threshold value and setting the λ to zero MSPRT will perform exactly like WTA.

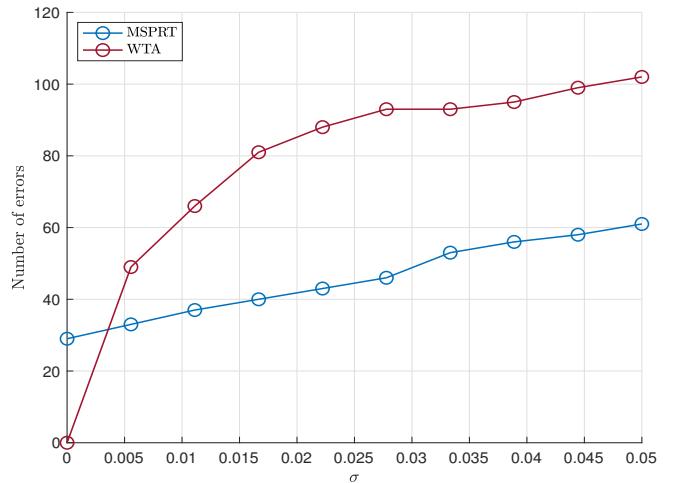


Fig. 3: MSPRT vs. WTA channels selection errors. Parameters of the simulation as in Table I.

Another valuable performance index is the number of switches, the lower the number of switches the more stable the behaviour of the agent. Fig. 4 shows the switching logic for the MSPRT and WTA for a selection of the data-set.

It is evident how the MSPRT not only picks the best action more effectively than WTA but also tends to stick with a sub-optimal action rather than continuously changing the channel which could lead to vehicle instability.

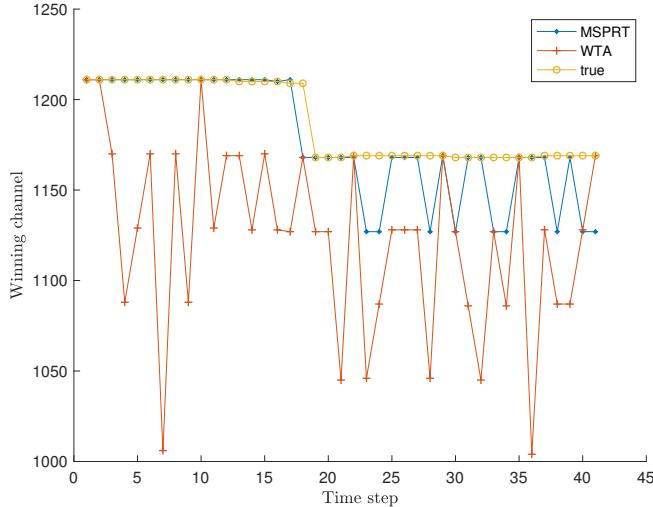


Fig. 4: MSPRT vs. WTA channels switching for $\sigma = 0.5$. Parameters of the simulation as in Table I.

V. CONCLUSIONS

We have shown that bio-inspired cognitive models can play a substantial role in the process of decision making in automated driving. In particular we demonstrated how the biologically inspired MSPRT algorithm can be adapted to both the inference and the action selection process given a suitable lower-level architecture for the agent. The advantages of the proposed formulation lie in an improved robustness to noisy observations (Fig. 3) and a greater stability of the chosen action (Fig. 4) with respect to traditional action selection. Indeed the effectiveness of the MSPRT depends on the tuning of application dependent hyperparameters. The activation *threshold* shapes the sensitivity to the process noise: the lower the threshold the more responsive will be the action picking, the higher the threshold the more robust the selection. Similar considerations apply for the tuning of the forgetting factor λ . However, we proved via simulation that it is possible to find an effective trade off adjustment for the MSPRT such that the algorithm outperforms other techniques. In particular for the considered data-set and $\sigma = 0.5$ the MSPRT guarantees an error rate up to 40% inferior to the WTA algorithm. Further work will be devoted to the set-up of a “layered” action selection process where a lower layer will be in charge of merging the contribution of channels encoding the same action to make sure that the affordance competition takes place among statistically independent channels only in order to run the MSPRT more efficiently.

VI. FUNDING

This work is supported by the European Commission Grant 731593 (Dreams4Cars) and 723211 (SafeStrip).

REFERENCES

- [1] T. Gindele, D. Jagszent, B. Pitzer, and R. Dillmann, “Design of the planner of team anneways autonomous vehicle used in the darpa urban challenge 2007,” in *2008 IEEE Intelligent Vehicles Symposium*. IEEE, 2008, pp. 1131–1136.
- [2] C. Urmson, J. Anhalt, D. Bagnell, C. Baker, R. Bittner, M. Clark, J. Dolan, D. Duggins, T. Galatali, C. Geyer *et al.*, “Autonomous driving in urban environments: Boss and the urban challenge,” *Journal of Field Robotics*, vol. 25, no. 8, pp. 425–466, 2008.
- [3] S. Brechtel, T. Gindele, and R. Dillmann, “Probabilistic MDP-behavior planning for cars,” in *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2011, pp. 1537–1542.
- [4] J. Schroder, M. Hoffmann, M. Zollner, and R. Dillmann, “Behavior decision and path planning for cognitive vehicles using behavior networks,” in *2007 IEEE Intelligent Vehicles Symposium*. IEEE, 2007, pp. 710–715.
- [5] C. Vallon, Z. Ercan, A. Carvalho, and F. Borrelli, “A machine learning approach for personalized autonomous lane change initiation and control,” in *2017 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2017, pp. 1590–1595.
- [6] M. Mukadam, A. Cosgun, A. Nakhaei, and K. Fujimura, “Tactical decision making for lane changing with deep reinforcement learning,” 2017.
- [7] D. Sadigh, S. Sastry, S. A. Seshia, and A. D. Dragan, “Planning for autonomous cars that leverage effects on human actions.” in *Robotics: Science and Systems*, vol. 2. Ann Arbor, MI, USA, 2016.
- [8] D. McFarland, *Problems of animal behaviour*. Longman Sc & Tech, 1989.
- [9] P. Redgrave, T. J. Prescott, and K. Gurney, “The basal ganglia: a vertebrate solution to the selection problem?” *Neuroscience*, vol. 89, no. 4, pp. 1009–1023, 1999.
- [10] P. Cisek, “Cortical mechanisms of action selection: the affordance competition hypothesis,” *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 362, no. 1485, pp. 1585–1599, 2007.
- [11] M. Da Lio, A. Mazzalai, and M. Darin, “Cooperative Intersection Support System Based on Mirroring Mechanisms Enacted by Bio-Inspired Layered Control Architecture,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 5, pp. 1415–1429, May 2018. [Online]. Available: <https://ieeexplore.ieee.org/document/8011474/>
- [12] G. Valenti, D. Piscini, and F. Biral, “A cooperative intersection support application enabled by safe strip technology both for c-its equipped, non-equipped and autonomous vehicles,” in *European conference on Intelligent Transportation Systems*, 2019.
- [13] M. Da Lio, F. Biral, E. Bertolazzi, M. Galvani, P. Bosetti, D. Windridge, A. Saroldi, and F. Tango, “Artificial Co-Drivers as a Universal Enabling Technology for Future Intelligent Vehicles and Transportation Systems,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 1, pp. 244–263, 2015.
- [14] M. Werling, S. Kammel, J. Ziegler, and L. Gröll, “Optimal trajectories for time-critical street scenarios using discretized terminal manifolds,” *The International Journal of Robotics Research*, vol. 31, no. 3, pp. 346–359, 2012.
- [15] E. Todorov and M. I. Jordan, “A minimal intervention principle for coordinated movement,” in *Advances in neural information processing systems*, 2003, pp. 27–34.
- [16] G. Valenti, L. De Pascali, and F. Biral, “Estimation of longitudinal speed profile of car drivers via bio-inspired mirroring mechanism,” in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, pp. 2140–2147.
- [17] C. W. Baum and V. V. Veeravalli, “A sequential procedure for multihypothesis testing,” *IEEE Transactions on Information Theory*, vol. 40, no. 6, 1994.
- [18] V. Draglia, A. G. Tartakovsky, and V. V. Veeravalli, “Multihypothesis sequential probability ratio tests. i. asymptotic optimality,” *IEEE Transactions on Information Theory*, vol. 45, no. 7, pp. 2448–2461, 1999.
- [19] R. Bogacz and K. Gurney, “The basal ganglia and cortex implement optimal decision making between alternative actions,” *Neural computation*, vol. 19, no. 2, pp. 442–477, 2007.

A dynamic neural model for endowing intelligent cars with the ability to learn driver routines: where to go, when to arrive and how long to stay there?

Flora Ferreira¹, Weronika Wojtak^{1,2}, Wolfram Erlhagen¹, Paulo Vicente^{1,2}, Ankit R. Patel², Sérgio Monteiro², and Estela Bicho²

Abstract—For many people, driving a car is a routine activity where they tend to go to the same places at about the same time of day or day of week. We propose a learning system – based on dynamic neural fields– that allows cognitive vehicles/cars to acquire sequential information about driver destinations and corresponding time properties. Importantly, the learning system allows to memorize long sequences, to deal with different temporal scales, and the destinations do not need to be fixed in advance. Learning occurs implicitly and it is a continuous process. Memory recall allows the car to predict driver’s destination intention, when she/he intends to arrive/leave, and for how long she/he intends to stay at a destination. Such personalized information can be used to plan the next trip.

I. INTRODUCTION

Many studies report that human mobility is characterized by a high degree of regularity [1], [2], a significant tendency to spend most of the time in a few locations [3] and a tendency to visit specific destinations at specific times [4], [5]. For example, for many drivers, weekdays consist of leaving home in the morning, driving to children’s school, work, again to children’s school and returning home in the evening. A person’s daily routines are typically coupled with routines across other temporal scales, such as going to the gym or the church, at specific days of the week.

Several different approaches, most of them statistical models [6], [7], have been proposed for predicting the next location in human mobility, in which a big data is necessary. Traditional Markov models work well for specific set of behaviors but have difficulty incorporating temporal patterns across different timescales [8] and destinations need to be fixed in advance. Here, we propose a dynamic neural model for learning information about the sequence of places and timing on the habits of individual drivers. The fundamental assumption is that driving is mostly a routine, and memory

*The work received financial support from European Structural and Investment Funds in the FEDER component, through the Operational Competitiveness and Internationalization Programme (COMPETE 2020) and national funds, through the FCT (Project “Neurofield”, ref POCI-01-0145FEDER-031393) and ADI (Project “Easy Ride:Experience is everything”, ref POCI-01-0247-FEDER-039334), FCT PhD fellowship PD/BD/128183/2016 and the Pluriannual Funding Programs of the research centres CMAT and Algoritmi.

¹Flora Ferreira, Weronika Wojtak, Wolfram Erlhagen and Paulo Vicente are with Center of Mathematics, University of Minho, Portugal wolfram.erlhagen@math.uminho.pt

²Weronika Wojtak, Paulo Vicente, Ankit R. Patel, Sérgio Monteiro and Estela Bicho are with Center Algoritmi, University of Minho, Portugal estela.bicho@dei.uminho.pt

recall of the past sequences (of destinations) with time information can be used to predict what is the driver’s intent. Learning occurs implicitly – driver does not need to be asked for his/her destinations – and is a continuous process modeled in the form of coupled dynamic neural fields (DNFs). The theoretical framework of DNFs has been proven to provide key processing mechanisms to implement working memory, prediction, and decision making in cognitive systems (e.g. [9], [10]), including the learning of sequential tasks ([12], [13], [14]).

In this study, the central idea is to explore learning mechanisms able to learn not only the sequence of driver destinations but also time properties, e.g. (i) when to be at a destination and (ii) when to leave. Memory recall allows the car to predict driver’s destination intention, when she/he intends to arrive, and for how long she/he intends to stay there.

II. THE APPROACH

The approach presented in this paper is based on previous work on memory mechanisms for order and timing of sequential processes [11], [13], [15] based on Dynamic Neural Fields (DNFs) [16], [17]. The central idea of dynamic field models is that relevant information is expressed by supra-threshold bumps of neural activity where each bump represents a specific parameter value. Input from external sources, such as information from a sensor, causes activation in the corresponding population that remains active with no further external input due to recurrent excitatory and inhibitory interactions within the populations. Those interactions are able to hold auto-sustained multi-bump patterns. We assume that the vehicle GPS coordinates and the information whether the car is turning on or off are available. We consider as input the GPS coordinates (latitude and longitude) when the vehicle is turning off or on, which represent the GPS coordinates of a destination at the time the driver arrives or departs, respectively. Figure 1 illustrates an overview of the model architecture consisting of several interconnected neural fields assuming as input the GPS coordinates when the car departs or arrives. For concreteness, we assume the case of “arrives signals” to describe the model.

The 2D field and the four 1D fields on top of the figure implement the encoding and memorizing of the GPS coordinates (latitude and longitude) of the places where the driver was and the relative timing between the moments in

which the driver arrived at those places. At the moment the driver turns off the car at specific GPS coordinates trigger the evolution of a bump in the input encode ON/OFF field $u_{ION/OFF}$. This activity is projected to corresponding neurons in two 1D fields: latitude perception field $u_{P_{lat}}$ and longitude perception field $u_{P_{long}}$ resulting in a localized bump in each of these fields. Each of these bumps triggers through excitatory connections the evolution of a self-sustained activity pattern at the corresponding site in latitude sequence memory field $u_{SM_{lat}}$ and longitude sequence memory field $u_{SM_{long}}$, respectively. Inhibitory feedback from $u_{SM_{lat}}$ to $u_{P_{lat}}$ (from $u_{SM_{long}}$ to $u_{P_{long}}$) destabilizes the existing bump in the perception field. This ensures that newly arrived localized input to $u_{P_{lat}}$ ($u_{P_{long}}$) will automatically drive the evolution of a bump at a different field location even if the specific cue value is repeated during the course of the sequence. The series of GPS coordinates at the moments when the car was turned off creates a multi-bump pattern in $u_{SM_{lat}}$ and in $u_{SM_{long}}$ that stores the last sequence of visited places with a strength of activation decreasing from bump to bump as a function of elapsed time since sequence onset. To guarantee the memory of successive routines for the same period of time, a dynamic building of a long term memory in $u_{LM_{lat}}$ ($u_{LM_{long}}$) is generated through excitatory connection from in $u_{SM_{lat}}$ ($u_{SM_{long}}$). Multi-bump pattern of $u_{SM_{lat}}$ ($u_{SM_{long}}$) is projected via excitatory connections in $u_{P_{lat}}$ ($u_{P_{long}}$) ensuring the robustness of the encoding process in the face of noisy and potentially incomplete sensory information. The resulting preshaping in $u_{P_{lat}}$ ($u_{P_{long}}$) based on prior experience modulates perception thresholds and speeds up the processing of inputs. When prediction are needed, the sequence recall mechanism is activated. During the recall the four 1D fields and the 2D field on bottom of Figure 1 become active. The latitude decision field $u_{D_{lat}}$ ($u_{D_{long}}$) receives the multi-bump pattern of $u_{LM_{lat}}$ ($u_{LM_{long}}$) as subthreshold input. By a continuous increase of the baseline activity in $u_{D_{lat}}$ ($u_{D_{long}}$), all subpopulations are brought closer to the threshold for the evolution of self-stabilized bumps. When the currently most active population reaches this threshold, the corresponding output in u_R is triggered. At the same time, the excitatory-inhibitory connections between associated populations in $u_{D_{lat}}$ ($u_{D_{long}}$) and latitude working memory field $u_{WM_{lat}}$ ($u_{WM_{long}}$) guarantee that the suprathreshold activity representing the latest sequence event becomes first stored in working memory field and subsequently suppressed. The global initial value of h in $u_{D_{lat}}$ ($u_{D_{long}}$) is proportional to the sequence duration (e.g. 24 hours) minus the time early (e.g. 10 minutes) in which arrive time to or depart time from a specific place should be predicted.

The population dynamics in each memory field is governed by the an integro-differential equation, which describes the activation of interconnected neurons along a one or two dimensional domain [18]:

$$\tau \frac{\partial u(\mathbf{r}, t)}{\partial t} = -u(\mathbf{r}, t) + S(\mathbf{r}, t) + h + \int_{\Omega} w(\mathbf{r}, \mathbf{r}') f(u(\mathbf{r}', t)) d\mathbf{r} \quad (1)$$

where $u(\mathbf{r}, t)$ represents the activity at time t at position \mathbf{r} on the domain Ω as a subset of \mathbf{R}^d with $d = 1$ or $d = 2$. The constant $\tau > 0$ defines the time scale of the field dynamics. The function $S(\mathbf{r}, t)$ represents the time-dependent, localized input to the field. The global inhibition, $h < 0$ defines the baseline level of activation to which field excitation decays without external stimulation. The connectivity function $w(\mathbf{r}, \mathbf{r}')$ models how a population of neurons at position \mathbf{r} in the field interacts with a population at position \mathbf{r}' . For the fields on which only one bump at a time should evolve (e.g. $u_{P_{lat}}$, u_D), we use a standard kernel of lateral inhibition type [18]. To enable multi-bump solutions in the fields (e.g. $u_{P_{lat}}$, u_D) we assume a kernel with oscillatory rather than monotonic decay [16], [17]:

$$w(\mathbf{r}) = A_w e^{-br} (b \sin(\alpha r) + \cos(\alpha r)) \quad (2)$$

where $r = |x|$ for 1D and $r = \sqrt{x^2 + y^2}$ for 2D, the parameters $A_w > 0$, $b > 0$ and $\alpha > 0$ control the amplitude, the rate at which the oscillations in w decay with distance, and the zero crossings of w , respectively. The firing function f is taken as the Heaviside step function with threshold 0.

A. Memory of interval timing between the destinations

To establish a stable activation gradient in the sequence memory fields we consider the following state-dependent dynamics [11], [13], [14]:

$$\tau_h \frac{\partial h(x, t)}{\partial t} = (1 - f(u(x, t))) (-h(x, t) + h_0) + k f(u(x, t)) \quad (3)$$

where h_0 defines the level to which h relaxes without suprathreshold activity at position x and $k > 0$ measures the growth rate when it is present. The adaptation of the resting level h is performed locally at field sites with suprathreshold activity. The memory of interval timing between the places in which the car arrives or departs is memorized in the peak amplitudes. Considering as input the GPS coordinates of a place at the moment in which the car is turned off, the difference between two successive peak amplitudes represents the interval timing between two successive places that the car was parked.

B. Memory of time spent in each destination

To create a memory of time duration in each place a 2D dynamic neural field is used. During recall this field receives as input the corresponding localized activation from output recall OFF and a representation of the time duration spent in each place is obtained by applying the following dynamics for h :

$$\tau_h \frac{\partial h(x, y, t)}{\partial t} = k f(u_{R_{OFF}}(x, y, t)) (1 - f(u_{R_{ON}}(x, y, t))). \quad (4)$$

The h value increases locally as a function of elapsed time only in the presence of activation in $u_{R_{OFF}}$ and when

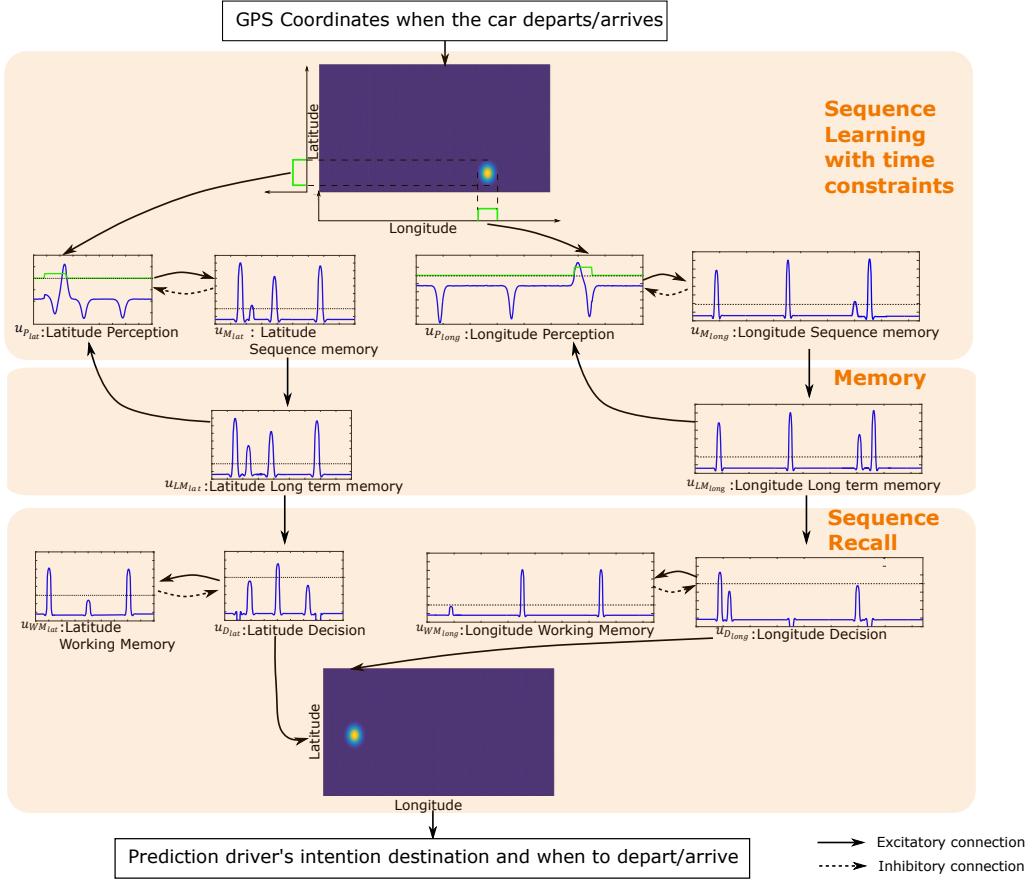


Fig. 1. Schematic view of the model architecture with several interconnected neural fields implementing sequence learning, memory and sequence recall for the depart/arrive signals (GPS coordinates when the car arrives or departs at destination as input). For details see the text.

simultaneously corresponding subpopulation in $u_{R_{ON}}$ is not activated (see Figure 2).

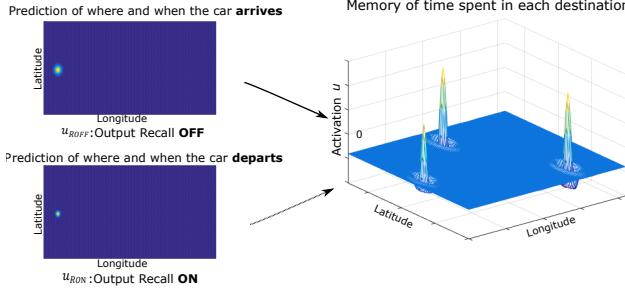


Fig. 2. Left: snapshot of activation of a bump in output recall fields $u_{R_{OFF}}$ and $u_{R_{ON}}$. Right: snapshot of a stable activation pattern corresponding to the duration time in each place already recalled.

Having a field with information about the time spent in each place will be useful, for example, to predict the time the car will stay in a specific destination.

III. RESULTS

As an example, we consider a day driver routine: depart from home to take the kids to school, next go to work, go to the restaurant to have lunch, come back to work and in the evening go to pick up the children from school,

take them to the gym and finally come back to home. To simulate this example we assume as input a real GPS coordinates of a Portuguese city (Guimarães) at a realistic time. Figure 3 (A) illustrate the sequence memory of where and when the car departs. The GPS coordinates memorized are represented with letter **P** symbol. The closer points (less than 400 meters in this case) represent the same place. The map illustrates the memory of five different places (Home, School, Work, Restaurant and Gym) in which two places (School and Work) were visited by the driver twice. Stable activation patterns corresponding to memory of the GPS coordinates (latitude and longitude) sequence are represented in two 1D fields where the bump amplitudes reflect the order of places where the car departed from and relative timing between them. Figure 3 (B) shows the representation in a 2D field of the time duration that the car was parked in each place during a day (from 0:00 to 24:00). Each **P** symbol on the map corresponds to a bump in the 2D field, and the amplitudes represent the duration in each location. The higher amplitudes represent the places in which the car was parked for a longer time (i.e. work and home).

IV. CONCLUSION

We have presented an approach to learn ordinal and temporal aspects of driver routines using the theoretical framework

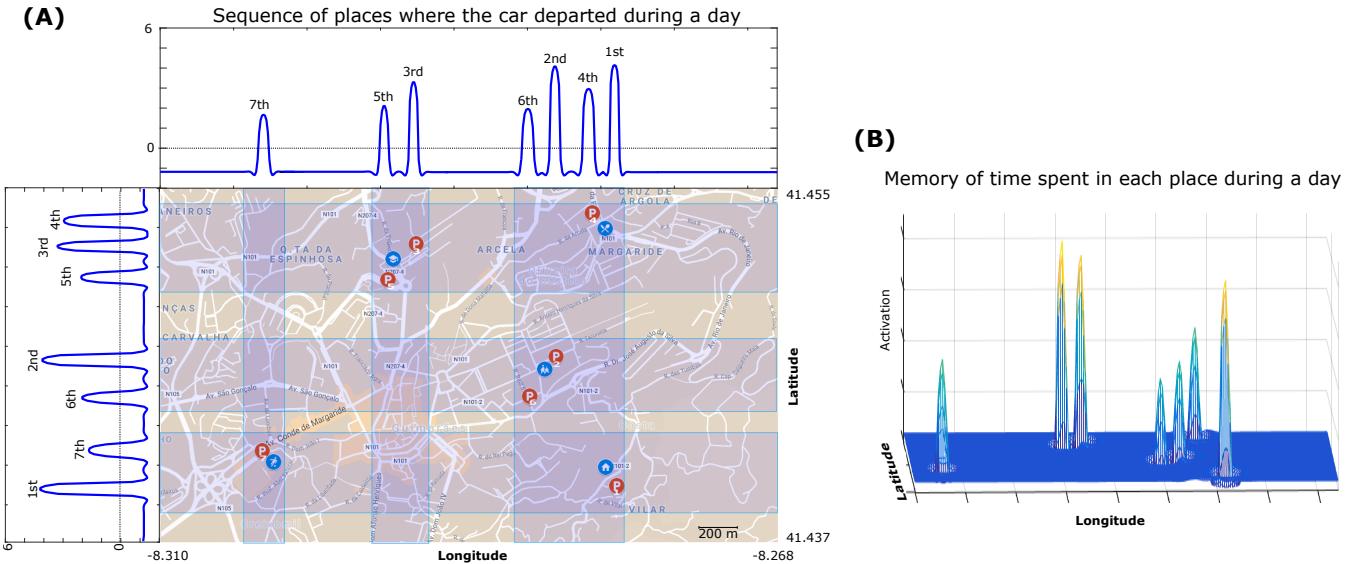


Fig. 3. (A) Part of Guimarães map in Portugal generated from Google Maps showing a sequence memory of the GPS coordinates of where the car departed during a day. The GPS coordinates represented in the bump centers are marked with letter P symbol. The closer points represent the same place. The seven marks are in five different places (Home, School, Work, Restaurant and Gym) in which two places have two marks. (B) Representation of time duration in each visited place.

of dynamic neural fields. The learning is implicit, continuous and can be scaled to different temporal scales. The model can be instantiated for each day of the week, and hence different routines can be learned. There are several possible uses for such learning memories. In terms of navigation systems, smarter route selection/recommendation could be provided through the integration of these memories with other factors such as traffic conditions without requiring input from the driver. The car could predict the next destination and the desired time of arrival and alert the driver if she is getting late to come to the car. Predicting the next departure time could be used for preparing in advance the cockpit's comfort – e.g. demist/defrost the windows and pleasant temperature – sometime before the driver (and occupants) enter the car. Future work concerns implementing and testing this learning system in real driving scenarios, in the scope of the joint project UMinho and Bosch – “Easy Ride: Experience is everything” (ref POCI-01-0247-FEDER-039334).

REFERENCES

- [1] N. Eagle and A. S. Pentland, Eigenbehaviors: Identifying structure in routine, *Behavioral Ecology and Sociobiology*, 63(7), 2009, pp.1057-1066.
- [2] C. Song, Z. Qu, N. Blumm, and A.L. Barabási, Limits of predictability in human mobility, *Science*, 327(5968), 2010, pp.1018-1021.
- [3] C. Song, T. Koren, P. Wang, and A.L. Barabási, Modelling the scaling properties of human mobility, *Nature Physics*, 6(10), 2010, p.818.
- [4] S. Jiang, J. Ferreira, M.C. and González, Clustering daily patterns of human activities in the city, *Data Mining and Knowledge Discovery*, 25(3), 2012, pp.478-510.
- [5] S. Rinzivillo, L. Gabrielli, M. Nanni, L. Pappalardo, D. Pedreschi, and F. Giannotti, The purpose of motion: Learning activities from individual mobility networks. In 2014 International Conference on Data Science and Advanced Analytics (DSAA), 2014, pp. 312-318.
- [6] R. Simmons, B. Browning, Y. Zhang, Y. and V.Sadekar, Learning to predict driver route and destination intent, In 2006 IEEE Intelligent Transportation Systems Conference, 2006, pp. 127-132.
- [7] M. Boukhechba, A. Bouzouane, S. Gaboury, C. Gouin-Vallerand, S. Giroux, and B. Bouchard, Prediction of next destinations from irregular patterns, *Journal of Ambient Intelligence and Humanized Computing*, 9(5), 2018, pp.1345-1357.
- [8] B.P. Clarkson, Life patterns: structure from wearable sensors, PhD diss, Massachusetts Institute of Technology, 2002.
- [9] G. Schöner, Dynamical systems approaches to cognition. Cambridge handbook of computational cognitive modeling, 2008, pp.101-126.
- [10] Y. Sandamirskaya, S. K. Zibner, S. Schneegans, and G. Schöner, Using dynamic field theory to extend the embodiment stance toward higher cognition. *New Ideas in Psychology*, 31(3), 2013, pp.322-339.
- [11] F. Ferreira, W. Erlhagen, and E. Bicho, A dynamic field model of ordinal and timing properties of sequential events, In International Conference on Artificial Neural Networks, Springer, Berlin, Heidelberg, 2011, pp. 325-332.
- [12] Y. Sandamirskaya, and G. Schoner, Dynamic field theory of sequential action: A model and its implementation on an embodied agent. In 2008 7th IEEE international conference on development and learning, 2008, pp. 133-138.
- [13] F. Ferreira, W. Erlhagen, E. Sousa, L. Louro, and E. Bicho, Learning a musical sequence by observation: A robotics implementation of a dynamic neural field model, In 4th International Conference on Development and Learning and on Epigenetic Robotics, 2014, pp. 157-162.
- [14] F. Ferreira, W. Wojtak, E. Sousa, L. Louro, E. Bicho, and W. Erlhagen, Rapid learning of complex sequences with time constraints: A dynamic neural field model, (Under review)
- [15] W. Wojtak, F. Ferreira, L. Louro, E. Bicho, and W. Erlhagen, Towards temporal cognition for robots: A neurodynamics approach, In 2017 Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob), 2017, pp. 407-412.
- [16] C.R. Laing, W.C. Troy, B. Gutkin, and G.B. Ermentrout, Multiple bumps in a neuronal model of working memory. *SIAM Journal on Applied Mathematics*, 63(1), 2002, pp.62-97.
- [17] F. Ferreira, W. Erlhagen, and E. Bicho, Multi-bump solutions in a neural field model with external inputs. *Physica D: Nonlinear Phenomena*, 326, 2016, pp.32-51.
- [18] S. Amari, Dynamics of pattern formation in lateral-inhibition type neural fields. *Biol. Cybern.* 27, 1977, pp. 77-87

Towards an Evaluation Methodology for the Environment Perception of Automotive Sensor Setups

How to define optimal sensor setups for future vehicle concepts?

Maike Hartstern^{1,2}, Viktor Rack¹ and Wilhelm Stork²

¹ BMW Group Research, New Technology and Innovation, Munich, Germany

² Institute for Information Processing Technologies, KIT Karlsruhe Institute of Technology, Karlsruhe, Germany
maiike.hartstern@bmw.de

Abstract— With increasing degree of automation, vehicles require more and more perception sensors to observe their surrounding environment. Car manufacturers are facing the challenge of defining a suitable sensor setup that covers all requirements. Besides the sensors' performance and field of view coverage, other factors like setup costs, vehicle integration and design aspects need to be taken into account. Additionally, a redundant sensor arrangement and the sensors' sensitivity to environmental influences are of crucial importance for safety. It is not feasible to explore every possible sensor combination in test drives. This paper presents a new simulation-based evaluation methodology, which allows the configuration of arbitrary sensor setups and enables virtual test drives within specific scenarios to evaluate the environmental perception in early development phases with metrics and key performance indicators. This evaluation suite is an important tool for researchers and developers to analyze setup correlations and to define optimal setup solutions.

Keywords— external vehicle sensors, perception, vehicle sensor setup configuration, sensor performance, virtual testing, simulation

I. INTRODUCTION

Advanced Driver Assistance Systems (ADAS) support the driver with functions like Adaptive Cruise Control (ACC), Emergency Braking and Parking Assistant [1, 2]. Vehicles are equipped with several external perception sensors to provide these functions with information about the car's environment. The automotive development is now heading towards Highly Automated Driving (HAD), Fully Automated Driving (FAD) and finally towards driverless Autonomous Driving (AD) to enhance the driving comfort and road safety. This rising degree of automation comes along with an increasing number of required perception sensors like cameras, radars, lidars and ultrasonic sensors to ensure sufficient coverage of the vehicle's surroundings. While the sensor setup for a typical ADAS was still manageable with one radar, a few cameras and four ultrasonic sensors at the front and back respectively, future setups will be more complex. To satisfy all requirements resulting from the higher automation degree like 360° surround view, far and near field coverage and redundancy, they will need to be built up with a much higher quantity of diverse sensors.

This leads to a variety of sensor configurations, which have to be analyzed to find the best solution [3]. Concerning the setup, three aspects have to be considered:

Sensor Configuration: The setup can be established with diverse sensors that are working according to different measurement principles. A redundant sensor arrangement shall ensure that important functions are still executable if one sensor drops out or a particular sensor technology is weak in a specific situation. The setup has to cover the vehicle's surrounding without dangerous blind spots. Aside from that, three different areas of interest have to be covered: the far field (highway driving), near field (urban driving) and ultra-near field (parking/start driving).

Sensor Integration: Besides optimal mounting positions and sensor alignment concerning Field Of View (FOV) coverage and sensor functionality, the feasibility of the geometrical integration and design aspects have to be considered. In addition, environmental influences like sensor occlusion due to dirt, weather and lighting conditions are crucial for sensor performance.

Sensor Benchmarking: Sensor specifications like FOV, range, detection probability and accuracy are crucial. However, another important factor is the costs of the overall setup. Cost-benefit analyses can reveal e.g. whether two adjoining sensors with small FOV can replace an expensive sensor with high FOV.

Defining an adequate sensor setup is a complex task. So far, there is a lack of a consistent evaluation procedure and suitable tool to support developers to solve this task in a time- and cost-efficient way. Thus, we are addressing the research question:

“Which evaluation methodology can be applied to determine the performance of automotive sensor setups regarding their environmental perception in an early development phase?”

In this paper, we introduce a simulation-based evaluation concept that assists the procedure of evolving an optimal sensor setup in the context of automated driving based on a reliable evaluation methodology. This also helps researchers to analyze setup correlations and influences of sensor parameters.

II. RELATED WORK AND BASICS

Perception sensors are the first part of a complex data processing chain, which is visualized schematically in Fig. 1.

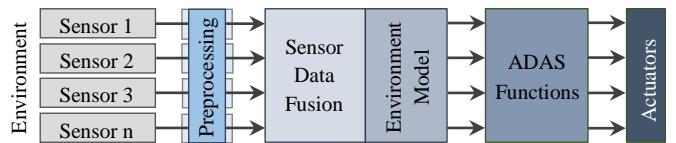


Figure 1: Schema of the data processing chain in automated vehicles.

The measurement data are preprocessed and are then entering the data fusion part. Afterwards, an environment model provides all the gathered information about the vehicle's surroundings, which is then forwarded to the ADAS functions that can trigger actuators, e.g. braking and steering. The determination of the final sensor setup plays a central role in the development of the whole data processing system since all these parts base on it. Consequently, the setup has to be defined at an early phase of the development process.

To date, sensor evaluation consists of various measurement procedures to analyze specific use cases and to validate the stated sensor specifications and measurement protocols under specified conditions. This allows comparing different sensors of the same technology. However, a practicable method is missing to constitute the performance of the complete setup. The execution of test drives, as well as the subsequent data analysis, are time-consuming and cost-intensive. Hence, this method is not feasible to compare many setup concepts in an early development phase. Thus, we propose the use of a simulation tool that assists developers in sensor configuration and setup evaluation concerning the environmental perception. Using our approach, the process of defining the particular sensor setup could follow a three-step-procedure:

1 Identification of sensor specifications

Determining the performance of particular sensors based on datasheets, measurement protocols, corner-case tests [4].

2 Simulation of the sensor setup

Using the information of step (1) to feed a simulation tool and obtain results on the overall setup performance.

3 Test drives

Optimizing the setup through performance tests under real environmental conditions for the chosen setup in step (2).

Step (1) and (3) describe the state-of-the-art evaluation method. The intermediate step (2) provides the potential to clarify correlations within this complex interrelationship of sensors. An appropriate simulation tool can thus be an important instrument to assist the process of defining the optimal setup in a time- and cost-efficient way at this early development stage [5].

III. APPROACH TOWARDS A SIMULATION FRAMEWORK

As Table I illustrates, many simulation tools exist for the purpose of automotive systems engineering [6]. Most of them address the technical consolidation and test bench applications in Hardware-in-the-Loop (HiL) and Software-in-the-Loop (SiL) systems to validate ADAS functions or to develop data fusion through virtual testing [7, 8]. For our use case, the focus has to be shifted towards the sensor configuration part.

TABLE I. SIMULATION TOOLS FOR AUTOMOTIVE SYSTEMS ENGINEERING

Simulation tool	Company
CarMaker	IPG Automotive
PreScan + DRS360	Siemens
DRIVE	NVIDIA
Virtual Test Drive VTD	Vires
CANape / vADASdeveloper	Vector
DYNA4 Driver Assistance	TESIS
ASM Traffic	dSPACE
Pro-SiVIC	CIVITEC
Automated Driving System Toolbox	MathWorks
Other simulation tools	rFpro, ANSYS, Addfor, ...

We are looking for a compact toolchain, which enables a configuration of sensor setups and a report on the overall setup performance. Thus, the main tool aspects are investigated:

Virtual Environment: Environment simulations create object lists as ground truth output, i.e. abstract information as classified objects with indications like object type, position, heading, velocity, acceleration and bounding box dimensions for each time-stamp (high-level data). The trend is towards physics-based simulation, which generates low-level sensor data (radar signals, pixel pictures, lidar point clouds). This allows to incorporate environmental influences on the signal propagation (weather/lighting). Those tools are still in development stage particularly with regard to the physical parameters like object shape and material properties for various object surfaces.

Sensor Models: Virtual sensors can be divided into four groups regarding their abstraction level (see Fig. 2). As first approach, we decided for probabilistic models that consider parameters like FOV, range and the statistical error behavior (detection probability, accuracy, false positives/negatives) [9]. With those modification options, these generic models can be adjusted to the properties of a specific sensor with good fidelity [10]. Phenomenological models are extensions of the probabilistic ones, which incorporate "situational effects" e.g. more measurement errors while driving into a tunnel. Usually, this information is only available after test drives. Thus, it cannot be implemented for research purposes. Ideal models are purely geometry-based (FOV, range) and highly simplified: they are not covering measurement errors. Physics-based models provide the highest fidelity [11]. Based on raytracing, they simulate signal propagation considering influences of weather and lighting effects as well as signal reflections on objects [12]. They cannot be used for our use case so far, since they are not available at the early time of setup configuration. Besides, their low-level data output requires a processing module to extract object detections out of the simulated raw data, which is not provided by any tools.

Data Fusion: There are several approaches regarding data fusion solutions for multi-sensor systems [13]. Usually, extended Kalman filters are applied in this context [14, 15]. Simulations are used to adapt the fusion algorithms to the specific sensor. In contrast, this part should remain fixed for the purpose of a sensor evaluation to ensure a consistent base.

Unfortunately, available tools do not include a ready to use data fusion module and an evaluation suite to assess the environmental perception of sensor setups. However, the sensor setup evaluation requires an intermediate stage of framework which focus on the sensor part with many modification options and an adequate fidelity. Thus, we established a new simulation workflow that allows the configuration of arbitrary setups and enables virtual test drives within specific scenarios to evaluate the environmental perception in early development phases quantitatively with key performance indicators (KPIs).

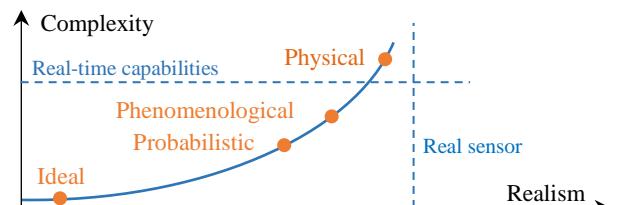


Figure 2: Abstraction levels of sensor models. [cf. Baselabs GmbH [17]]

IV. WORKFLOW CONCEPT AND METHODOLOGY

Our framework covers the entire process chain from data acquisition (ground truth, sensor data), data processing (data fusion), data evaluation (metrics) to data analysis (KPIs) to derive data insights and knowledge (setup performance). The workflow is visualized in Fig. 3 and is divided into two parts.

In the first part, ground truth data are generated with an environment simulation based on object lists. In the presented context, IPG CarMaker [16] is used. In this simulation part, the virtual environment, traffic and the driving maneuver of the ego vehicle are designed to create a scenario. The simulated ground truth data of the scenario are transferred to the second workflow part, which consists of the sensor configuration and evaluation suite. We used the software Baselabs Create [17], that is a development tool for environmental perception applications for automated vehicles. The ground truth data can either be processed directly or it can be recorded in a scenario collection for the use in replay mode. After the scenarios have been recorded, the first workflow part is not required anymore for working with the toolchain. Thus, the second workflow part is an independent framework, which leads to a flexible working tool.

The sensor configuration and evaluation suite contains three modules: First, the ground truth data enter the data fusion designer with sensor models and the setup configuration. In this part, the ground truth data are modified according to the settings of probabilistic sensor models. For each virtual sensor of the setup, a measurement model, a detection model as well as a track management strategy is selected via a graphical user interface (GUI). For this, the individual sensor parameters are set according to the sensor properties maintained from the three-step-procedure (1) *Identification of sensor specifications*. In addition, it is possible to modify the available sensor models or to add new models by the programmatic usage of a Software Development Kit (SDK). The simulated sensor data are fused in the data fusion module, which relies on extended Kalman filters and feeds the subsequent environment model. Afterwards, the simulated and fused sensor data are compared with the initial ground truth data in the evaluation metrics module that was built especially for this workflow. In this part, custom evaluation metrics can be added. Based on those metrics, a KPI report containing the results for all scenarios is created to assess the perception performance quantitatively in a compact overview. After that, KPI reports of different setups are compared and further aspects like cost-benefit considerations are analyzed to decide whether a particular sensor setup is worth being tested in the next step of the three-step-procedure: (3) *Test drives*.

V. EVALUATION AND METRICS

The evaluation suite calculates the KPIs “detection time”, “detection rate” and “false alarm rate”. In addition, three metrics are implemented and presented below. A set $G\{g_i\}$ of ground truth objects g_i for $i = 1, \dots, m$ as well as a set $E\{e_j\}$ of estimated objects e_j for $j = 1, \dots, n$ are assumed. The elements of the sets are multidimensional including object position, heading, velocity and acceleration. The Euclidean distance $d(g_i, e_j)$ calculates the distance between actual and estimated objects.

The *Hausdorff metric* $d_H(G, E)$ in (1) is insensitive to different cardinalities and weights outliers heavily [18].

$$d_H(G, E) = \max \left\{ \max_{g_i \in G} \min_{e_j \in E} d(g_i, e_j), \max_{e_j \in E} \min_{g_i \in G} d(g_i, e_j) \right\} \quad (1)$$

The *Optimal SubPattern Assignment (OSPA)* $\bar{d}_{OSPA}^{(c)}(G, E)$ in (2) punishes outliers less than the Hausdorff metric, depending on the cutoff parameter c . Instead, it penalizes the scenario when a ground truth object has several estimated objects. [18]

$$\bar{d}_{OSPA}^{(c)}(G, E) = \sqrt{\left(\frac{1}{n} \left(\min_{\pi \in \Pi_n} \sum_{i=1}^m d^{(c)}(g_i, e_{\pi(i)})^2 + c^2(n-m) \right) \right)} \quad (2)$$

To consider the position confidence of the estimated objects, the *Normalized Estimation Error Squared (NEES)* in (4), also known as Mahalanobis distance [15], includes the position error ($g - e$) and the covariance matrix P_{ge} .

$$NEES(g_i, e_j) = \sqrt{(g_i - e_j)^T P_{ge}^{-1} (g_i - e_j)} \quad (3)$$

VI. CONCLUSION

We presented a new simulation-based concept that is suitable for the evaluation of perception sensor setups for automated vehicles regarding particular sensor mounting positions, diverse setup configurations and different sensor technologies. Our established framework is an important instrument that assists automotive system engineers during the early stages of development and supports them with a performance overview for different setups in relevant scenarios. The evaluation suite allows configuring the sensor setup via a simple GUI while it is still possible to access the software code to modify the probabilistic sensor models and to implement evaluation metrics. By studying the correlations within the sensor data processing chain, requirement profiles in terms of a roadmap for future sensor technologies can be derived. In future work, the method could be adapted to a physics-based simulation like PreScan [19] to extend the sensor evaluation by considering physical effects.

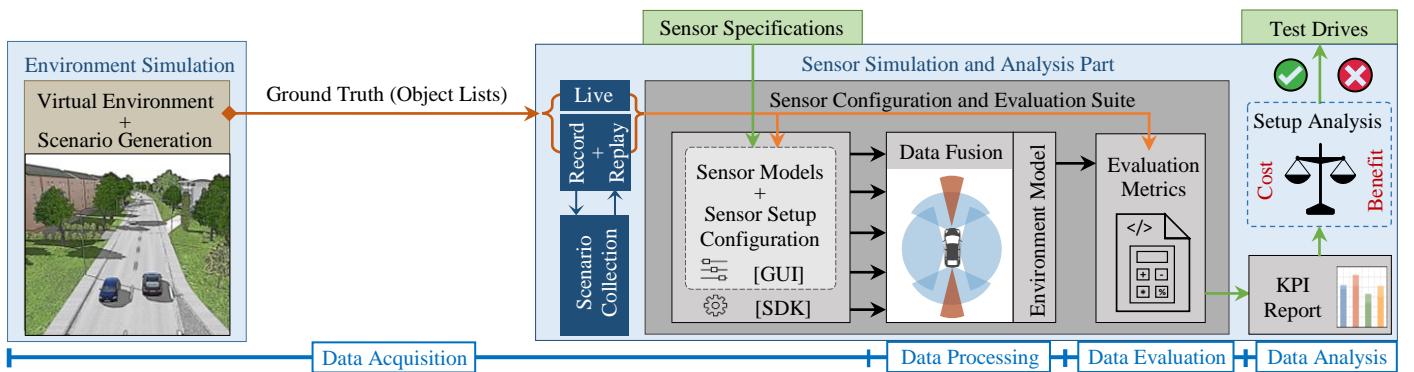


Figure 3. Schematic visualization of our simulation framework and our proposed workflow for the configuration and evaluation of perception sensor setups.

REFERENCES

- [1] A. Ziebinski, R. Cupek, D. Grzechca, and L. Chruszczyk, "Review of advanced driver assistance systems (ADAS)," in *AIP Conference Proceedings 1906*, p. 120002.
- [2] K. Bengler, K. Dietmayer, B. Farber, M. Maurer, C. Stiller, and H. Winner, "Three decades of driver assistance systems: Review and future perspectives," *IEEE Intelligent Transportation Systems Magazine*, vol. 6, no. 4, pp. 6–22, Winter 2014.
- [3] J. van Brummelen, M. O'Brien, D. Gruyer, and H. Najjaran, "Autonomous vehicle perception: The technology of today and tomorrow," *Transportation Research Part C: Emerging Technologies*, vol. 89, pp. 384–406, 2018.
- [4] S. Hasirlioglu, A. Kamann, I. Doric, and T. Brandmeier, "Test methodology for rain influence on automotive surround sensors," *IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, pp. 2242–2247, 2016.
- [5] T. Dörfler, "Testing ADAS sensors in early development phases," *ATZelektronik worldwide*, vol. 2018, no. 13, p. 54.
- [6] P. Viswanath, M. Mody, S. Nagori, J. Jones, and H. Garud, "Virtual simulation platforms for automated driving: Key care-about and usage model," *Electronic Imaging*, vol. 2018, no. 17, 164-1-164-6, 2018.
- [7] D. Gruyer, S. Choi, C. Boussard, and B. d'Andrea-Novel, "From virtual to reality, how to prototype, test and evaluate new ADAS: Application to automatic car parking," in *2014 IEEE Intelligent Vehicles Symposium Proceedings*, MI, USA, 2014, pp. 261–267.
- [8] W. Huang, K. Wang, Y. Lv, and F. Zhu, "Autonomous vehicles testing methods review," in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC): Windsor Oceanico Hotel, Rio de Janeiro, Brazil, November 1-4, 2016*, Rio de Janeiro, Brazil, 2016, pp. 163–168.
- [9] T. Hanke, N. Hirsenkorn, B. Dehlink, A. Rauch, R. Rasshofer, and E. Biebl, "Classification of sensor errors for the statistical simulation of environmental perception in automated driving systems," in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC): Windsor Oceanico Hotel, Rio de Janeiro, Brazil, November 1-4, 2016*, Rio de Janeiro, Brazil, 2016, pp. 643–648.
- [10] R. Schubert, N. Mattern, and R. Bours, "Simulation of sensor models for the evaluation of advanced driver assistance systems," *ATZelektronik worldwide*, vol. 9, no. 3, pp. 26–29, 2014.
- [11] C. van Driesten and T. Schaller, "Overall approach to standardize AD sensor interfaces: Simulation and real vehicle," in *Proceedings, Fahrerassistenzsysteme 2018*, T. Bertram, Ed., Wiesbaden: Springer Fachmedien Wiesbaden, 2019, pp. 47–55.
- [12] C. Goodin, R. Kala, A. Carrillo, and L. Y. Liu, "Sensor modeling for the virtual autonomous navigation environment," in *2009 IEEE Sensors*, Christchurch, New Zealand, Oct. 2009 - Oct. 2009, pp. 1588–1592.
- [13] S. Matzka and R. Altendorfer, "A comparison of track-to-track fusion algorithms for automotive sensor fusion," in *Lecture Notes in Electrical Engineering, Multisensor Fusion and Integration for Intelligent Systems*, H. Hahn, H. Ko, and S. Lee, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 69–81.
- [14] S. Thrun, W. Burgard, and D. Fox, *Probabilistic robotics*: The MIT Press, 2005.
- [15] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with applications to tracking and navigation: Theory algorithms and software*, 1st ed.: Wiley-Interscience, 2001.
- [16] IPG Automotive GmbH, *CarMaker*. [Online] Available: <https://ipg-automotive.com/de/produkte-services/simulation-software/carmaker/>. Accessed on: Jun. 24 2019.
- [17] Baselabs GmbH, *Data fusion for automated driving: Data fusion results*. [Online] Available: <https://www.baselabs.de/>. Accessed on: Mar. 21 2019.
- [18] D. Schuhmacher, B.-T. Vo, and B.-N. Vo, "A consistent metric for performance evaluation of multi-object filters," *IEEE Trans. Signal Process.*, vol. 56, no. 8, pp. 3447–3457, 2008.
- [19] Leneman, F., D. Verburg, and S. Buijsse, Ed., *PreScan, testing and developing active safety applications through simulation*, 2008.

Risk-Aware Reasoning for Autonomous Vehicles

Majid Khonji, Jorge Dias, and Lakmal Seneviratne

Abstract—A significant barrier to deploying autonomous vehicles (AVs) on a massive scale is safety assurance. Several technical challenges arise due to the uncertain environment in which AVs operate such as road and weather conditions, errors in perception and sensory data, and also model inaccuracy. In this paper, we propose a system architecture for risk-aware AVs capable of reasoning about uncertainty and deliberately bounding the risk of collision below a given threshold. We discuss key challenges in the area, highlight recent research developments, and propose future research directions in three subsystems. First, a perception subsystem that detects objects within a scene while quantifying the uncertainty that arises from different sensing and communication modalities. Second, an intention recognition subsystem that predicts the driving-style and the intention of agent vehicles (and pedestrians). Third, a planning subsystem that takes into account the uncertainty, from perception and intention recognition subsystems, and propagates all the way to control policies that explicitly bound the risk of collision. We believe that such a white-box approach is crucial for future adoption of AVs on a large scale.

I. INTRODUCTION

Over the past hundred years, innovation within the automotive industry has created more efficient, affordable, and safer vehicles, but progress has been incremental so far. The industry now is on the verge of a substantial change due to the advancements in Artificial Intelligence (AI) and Autonomous Vehicle (AV) sensing technologies. These advancements offer the possibility of significant benefits to society, saving lives, and reducing congestion and pollution. Despite the progress, a significant barrier to large scale deployment is safety assurance. Most technical challenges are due to the uncertain environment in which AVs operate such as road and weather conditions, errors in perception and sensory input data, and uncertainty in the behavior of the pedestrians and agent vehicles. A robust AV control algorithm should account for different sources of uncertainty and generate control policies that are quantifiably safe. In addition, algorithms that respect precise safety measures can assist policymakers addressing legislative issues related to AVs, such as insurance policies and ultimately convince the public for a wide deployment of AVs.

One of the most prevalent measures for AV safety is the number of crashes per million miles [1]. Although such a measure provides some estimate on overall safety performance in a particular environment, it fails to capture unique differences and the richness of individual scenarios. As AVs become more prevalent, the reasoning behind in-

Majid Khonji, Jorge Dias, and Lakmal Seneviratne are with KU Center for Autonomous Robotic Systems, Khalifa University, Abu Dhabi, UAE (email {majid.khonji, jorge.dias, lakmal.seneviratne}@ku.ac.ae).

dividual events becomes of critical importance as the public would require transparency and explainable AI. Recent AV fatal crashes raise further debates among scholars and pioneers in the industry concerning how an autonomous vehicle should act when human safety is at risk. On a more philosophical level, a study [2] sheds light on the major challenges of understanding societal expectations about the principles that should guide the decision making in life-critical situations. As an illustrative example, suppose a self-driving vehicle, experiencing a partial system failure, forced into an ultimatum choice between running over pedestrians or sacrificing itself and its passenger to save them. What should be the reasoning behind such a situation, and more fundamentally, what should be the moral choice? Despite the profound philosophical dilemma and the impact on the public perception of AI as a whole and the regulatory aspects for AVs in particular, the current state-of-the-art of the technological stack of AVs does not explicitly capture and propagate uncertainty sufficiently well throughout decision processes in order to accurately assess these edge scenarios.

In this work, we discuss algorithmic pipeline and a technical stack for AVs to capture and propagate uncertainty from the environment throughout perception, prediction, planning, and control. An AV has to be able to plan and optimize trajectories from its current location to a goal while avoiding static and dynamic (moving) obstacles, while meeting deadlines and efficiency constraints. The risk of collision should be bounded by a given safety threshold that meets governmental regulations, while meeting deadlines should meet a quality of service threshold.

To expand AV perception range, we consider the Vehicular Ad-Hoc Network (VANET) communication model. Vehicle-to-Vehicle (V2V), Vehicle-to-Infrastructure (V2I), and more recently Vehicle-to-Everything (V2X), are technologies that enable vehicles to exchange safety and mobility information between each other and with the surrounding agents, including pedestrians with smart phones and smart wearables. Vehicles can collect information en route, such as road conditions and position estimates of static and dynamic objects, and can use this information to continuously predict actions performed by other vehicles and infrastructure. V2V messages would have a range of approximately 300 meters, which exceeds the capabilities of systems with cameras, ultrasonic sensors, and LIDAR, allowing greater capability and time to warn vehicles.

In this work, we propose a system architecture (Sec. II) and discuss key challenges in quantifying uncertainty at different levels of abstractions: scene representation (Sec. III), intention recognition (Sec. IV), risk-bounded planning

(Sec. V), and control (Sec. VI). We highlight current state-of-the-art, and propose research directions at each level.

II. SYSTEM ARCHITECTURE

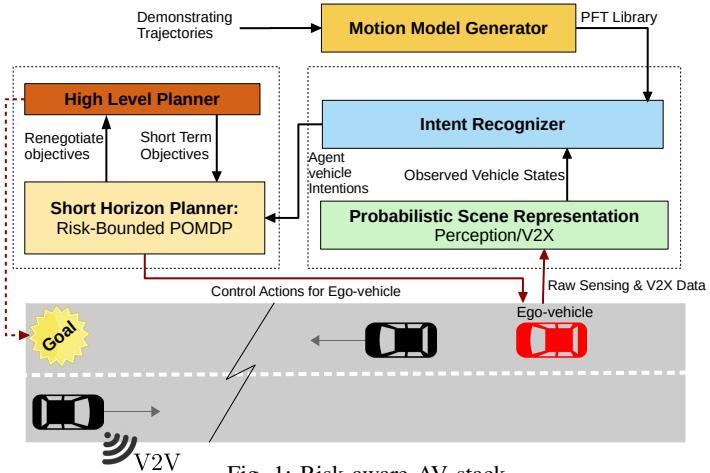


Fig. 1: Risk-aware AV stack.

In the following, we present the architecture of a risk-aware AV stack with six technical objectives in mind:

- A probabilistic perception and object representation system that takes into consideration uncertainty that arises from hardware modalities and sensor fusion. The system will capture uncertainty in object classification, bounding geometries, and temporal inconsistencies under diverse conditions.
- Leverage the communication network to gain knowledge of the surrounding agents (vehicles and pedestrians) that are beyond line-of-sight, and then improve upon scene representation.
- An intention recognition system that takes into account all dynamic objects (vehicles and pedestrians), from perception and V2X communication, and estimates a distribution over potential future trajectories.
- Generalize upon recently developed risk-aware optimization algorithms [3], [4], in order to ensure that movements are safe.
- On a higher level, propose goal-directed autonomous planners that strive to meet the passenger goals and preferences, and help the passengers to think through adjustments to their goals, when they can't be safely met.
- To ensure that decisions are made in a timely manner, design polynomial-time approximation algorithms that offer formal bounds on sub-optimality, and which produce near-optimal results.

In addition, by specifying the probability that a plan is executed successfully, the system operator or policymaker can set the desired level of conservatism in the plan in a meaningful manner and can trade conservatism against performance. Fig. 1 shows the interaction between key components of the system as we illustrate throughout the paper.

III. PROBABILISTIC SCENE REPRESENTATION

Scene understanding is research topic with strong impact on technologies for autonomous vehicles. Most of the efforts have been concentrated on understanding the scenes surrounding the ego-vehicle (autonomous vehicle itself). This is composed by sensor data processing pipeline that includes different stages such as low-level vision tasks, detection, tracking and segmentation of the surrounding traffic environment –e.g., pedestrian, cyclists and vehicles. However, for an autonomous vehicle, these low-level vision tasks are insufficient to comprehensive scene understanding. It is necessary to include reasoning about the past and the present of the scene participants. This paper intends to guide future research on interpretation of traffic scene in autonomous driving from a probabilistic event reasoning perspective.

A. Probabilistic Context Layout for Driving

Scene representation includes context representations that include spatially geometrical relationships [5] among different traffic elements with certain semantic labels. It is different from the semantic segmentation frameworks [6], [7], because the context representation does not only contain the static components of traffic scene (typical technique for this aspect is simultaneous localization and mapping (SLAM)), such as road, the type of traffic lanes, traffic direction, and participant orientation, but also consists of several kinds of dynamic elements, e.g., motion correlation of participants. The study [8],[9] has given a detailed review on semantic segmentation, taking the traffic geometry inferring into consideration.

A key aspect of context representation is to extract salient features from a large set of sensor data. For that purpose, it is necessary to establish a saliency mechanism, that is a critical region extraction and information simplification technique that is widely used for attractive region selection in images. Over the past few decades, saliency has been generally formulated as bottom-up and top-down modes. Bottom-up modes [10], [11] are fast, data-driven, pre-attentive and task-independent. Top-down approaches [12], [13], [14], [15] often entail supervised learning with pre-collected task labels by a large set of training examples and are task-oriented and vary in different environments.

A recent work [16] presents a fast algorithm that obtains a probabilistic occupancy model for dynamic obstacles in the scene with few sparse LIDAR measurements. Typically the occupancy states exhibit highly nonlinear patterns that cannot be captured with a simple linear classification model. Therefore, deep learning models and kernel-based models can be considered as potential candidates. However, these approaches require either a massive amount of data or a high number of hyper-parameters to tune. A promising future direction is to extend this approach to account for different object classes (rather than occupancy map) and other sensors as well such as cameras.

B. Beyond Line-of-sight

Any sensing modality has blind spots. For objects that lie beyond-line-of-sight, one can consider a communication network to improve upon the scene representation. This can

be critical in certain edge scenarios. For example, in Fig. 2, the ego-vehicle (red) has two options: either maintain speed or overtake the vehicle ahead. Suppose that another agent vehicle is approaching from a distance that is not detected by onboard sensors of the ego-vehicle. In this scenario, both the speed and location of the distant vehicle might not be accurately estimated, therefore maneuver A_2 leading to a collision.

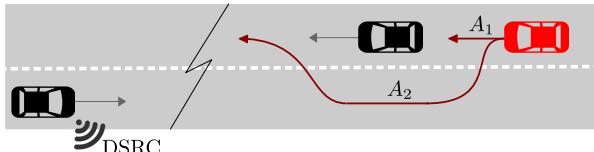


Fig. 2: V2V communication.

There has been substantial progress for the standardization of vehicle-to-everything/V2X (V2V/V2I/V2P) communication protocols. The major V2X standards are known as DSRC (Dedicated Short-Range Communications) [17] as well as 5G [18]. The introduction of 5G's millimeter-wave transmissions brings a new paradigm to wireless communications. Depending on the application, 5G positioning can also enhance tracking techniques, which leverage short-term historical data (local signatures and key features). Uncertainty can be captured by probabilistic models (e.g., Gaussian) through sampling temporal inconsistencies in historical data streams such as localization data, and parameter tuning.

IV. INTENTION RECOGNITION

This subsystem involves prediction and machine learning tasks to reliably estimate the future trajectories of uncontrollable agents in the scene, including pedestrians and other agent vehicles. Many existing trajectory prediction algorithms [19], [20] obtain deterministic results quite efficiently. However, these approaches fail to capture the uncertain nature of human actions. Probabilistic predictions are beneficial in many safety-critical tasks such as collision checking and risk-aware motion planning. They can express both the intrinsically uncertain prediction task at hand (human nature) and reasoning about the limitations of the prediction method (knowing when an estimate could be wrong [21]). To incorporate uncertainties into prediction results, data-driven approaches can learn common characteristics from datasets of demonstrated trajectories [22], [23]. These methods often express uni-modal predictions, which may not perform well in sophisticated urban scenarios where the driver can choose among multiple actions. A recent work [24] presents a hybrid approach using a variational neural network that predicts future driver trajectory distributions for the ego-vehicle based on multiple sensors in urban scenarios. The work can be extended in future to predict trajectories for agent-vehicles using V2V data streams, if available.

We propose a simple intent recognition that is divided into two steps. First we continuously record high-level maneuvers of surrounding vehicles (both off-line and online). Examples of such maneuvers are merge left, merge right, accelerate all at different velocities and variations and so on. Each of these

maneuvers comprises of a set of collected trajectories. Due to the uncertainties in the motions of human-driven vehicles, we learn a compact motion representation called Probabilistic Flow Tube (PFT) [25] from demonstrating trajectories to capture human-like driver styles and uncertainties for each maneuver. A library of pre-learned PFTs can be used to estimate the current maneuver as well as predict the probabilistic motion of each agent vehicle using a Bayesian approach.

V. RISK-BOUNDED PLANNING

Deterministic optimization approaches have been well developed and widely used in several disciplines and industries, in order to optimize processes both off-line and on-line. In this work, we characterize uncertainty in a probabilistic manner and find the optimal sequence of ego-vehicle trajectory control, subject to the constraint that the probability of failure must be below a certain threshold. Such constraint is known as a chance constraint. In many applications, the probabilistic approach to uncertainty modeling has a number of advantages over a deterministic approach. For instance, disturbances such as vehicle wheel slip can be represented using a stochastic model. When using a Kalman Filter for enhancing localization, the location estimate is provided as a probabilistic distribution. In addition, by specifying the probability that a plan is executed successfully, the system operator or policymaker can set the desired level of conservatism in the plan in a meaningful manner and can trade conservatism against performance. Therefore, robustness is achieved by designing solutions that guarantee feasibility as long as disturbances do not exceed these bounds. Furthermore, if the passenger goals cannot be safely achieved, then the chance constraints can be analyzed to pinpoint the sources of risk, and the user goals can be adjusted, based on their preferences, in order to restore safety.

Reasoning under uncertainty has several challenges. The optimization problem of trajectory optimization is non-convex, due to discrete choices and the presence of obstacles in the feasible space. One approach to tackle the challenges is by introducing multiple layers of abstractions. Instead of solving high-level problems (e.g., route planning) and low-level problems (e.g., steering wheel angle, acceleration, and brake commands) in a single shot, one can decouple them into sub-problems. We achieve such hierarchy through a high-level planner, short-horizon planner, and precomputed and learned maneuver trajectories as we illustrate below.

A. High Level Planner

High-level planning involves route planning, applying traffic rules, and consequently setting short-term objectives (aka set points), which will be fed into Short Horizon Planner (as shown in Fig. 1). The planner adjusts those short-term objectives when no safe solution exists. To be able to model the feasibility of an obtained plan, we leverage Temporal Plan Networks (TPN) [26]. A TPN is a graph where the nodes represent events, and the edges represent activities. In temporal planning, the ego-vehicle is presented with a series of events and must decide precisely when to schedule them. STNs with Uncertainty (STNUs) is an extension allowing

to reason over stochastic, or uncontrollable, actions and their corresponding durations [27]. Such formalism allows to check the feasibility of a high-level plan and prompt the user to adjust his or her intermediate goals and time constraints to output smooth intermediate plans, fed into the short horizon planner.

B. Short Horizon Planner

Planning under uncertainty is a fundamental area in artificial intelligence. For the application of AV, it is crucial to plan for potential contingencies instead of planning a single trajectory into the future. This often occurs in dynamic environments where the vehicle has to react quickly (in milliseconds) to any potential event. Partially observable Markov decision processes (POMDP)[28], [29] provide a model for optimal planning under actuator and sensor uncertainty, where the goal is to find policies (contingency plans) that maximize (or minimize) some measure of expected utility (or cost).

In many real-world applications, a single measure of performance is not sufficient to capture all requirements (e.g., an AV tasked to minimize commute time while keeping the distance from obstacle below a given threshold). This extension is often called constrained POMDP (C-POMDP) [30]. When constraints involve stochasticity (e.g., distance following a probabilistic model), the problem is modeled as chance-constrained POMDP (CC-POMDP) [4], where we have a bound on the probability of violating constraints. To calculate the risk of each decision, one can leverage the probabilistic flow-tube (PFTs) concept to model a set of possible trajectories [25]. The current state-of-the-art solver of CC-POMDP is called RAO* [4]. RAO* generates a conditional plan based on action and risk models and likely possible scenarios for agent vehicles.

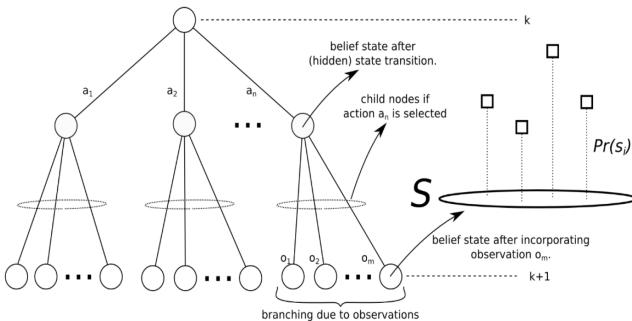


Fig. 3: CC-POMDP Hypergraph: Nodes are the probability distributions of states (belief states) of ego vehicle. At each node, there are n possible actions that can be taken by the ego vehicle. At each level, belief state is updated with respect to chosen action and observations of the environment.

RAO* explores from a probability distribution of vehicle states (belief state), by incrementally constructing a hypergraph, called the explicit hyper-graph shown in Fig. 3. At each node of the hyper-graph, the planner considers possible actions provided by Motion Model Generator (see Fig. 1) and receives several possible observations. At each level, it utilizes a value heuristic to guide the search towards optimal

policies. It also uses a risk heuristic to prune the search space, removing high-risk branches that violate the chance constraints. Hence, at each level, the action that maximizes expected reward and meets chance constrained is selected for the vehicle. However, one of the drawbacks of RAO* is that it does not always return optimal solutions and also does not provide any bound on the sub-optimality gap. In a recent work [3], we provide an algorithm that provides guarantee on optimality (namely, a fully polynomial time approximation scheme (FPTAS)) while preserving safety constraints, all within polynomial running time.

Recently [31] applied RAO* for the application of self-driving vehicles under restricted settings (e.g., known distribution of actions taken by agent-vehicles). CC-POMDP, while otherwise expressive, allow only for sequential, non-durative actions. This poses restrictions in modeling real-world planning problems. In our recent ongoing work, we extend the framework of CC-POMDP to account for durative actions, and leverage heuristic forward search to prune the search space to improve upon the running time.

VI. MOTION MODEL GENERATOR

Based on each driving scenario, we compute a library of maneuvers. Each maneuver is associated with nominal control signals by solving a model predictive control (MPC) optimization problem [31]. The set of possible maneuver actions are constrained by traffic rules and vehicle dynamics and are informed by the expected evolution of the situation. Computing the actions can be accomplished through offline and online computation, and also through publicly available datasets (e.g., Berkeley DeepDrive BDD100k).

The size of the search space of CC-POMDP, described above, is sensitive to the number of maneuver actions. To tackle this issue, we consider three different levels for abstractions. i) Micro Actions are primitive actions like Accelerate, Decelerate, Maintain. ii) Maneuver Actions are sequences of micro actions like Merge left, Merge right, iii) Macro Actions are sequences of maneuver actions such as pass the front vehicle, go straight until next intersection [32].

To calculate the risk of collision, we leverage PFT, which represents a sequence of probabilistic reachable sets. PFTs show probabilistic future predictions for states of the vehicles under a selected action. In this context, the intersection between two, temporally aligned, PFT trajectories represents the risk of collision. To construct PFTs, we use vehicle dynamics and also probabilistic information about uncertainties, as well as through learning from datasets. By propagating the probability distributions of uncertainties through the continuous dynamics of the vehicle, we construct probability distributions for the locations of the vehicle over a finite planning horizon.

VII. CONCLUSION

In this work, we proposed a system architecture for risk-aware AVs that can deliberately bound the risk of collision below a given threshold, defined by the policymaker. We presented the related work, discussed key challenges, and proposed research directions in three key subsystems: perception, intention recognition, and risk-aware planning. We

believe that our white-box approach is crucial for a better understanding of AV decision making and ultimately for future adoption of AVs on a large scale.

REFERENCES

- [1] N. Kalra and S. M. Paddock, "Driving to safety: How many miles of driving would it take to demonstrate autonomous vehicle reliability?" *Transportation Research Part A: Policy and Practice*, vol. 94, pp. 182–193, 2016.
- [2] J.-F. Bonnefon, A. Shariff, and I. Rahwan, "The social dilemma of autonomous vehicles," *Science*, vol. 352, no. 6293, pp. 1573–1576, 2016.
- [3] M. Khonji, A. Jasour, and B. Williams, "Approximability of constant-horizon constrained pomdp," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*. International Joint Conferences on Artificial Intelligence Organization, 7 2019, pp. 5583–5590. [Online]. Available: <https://doi.org/10.24963/ijcai.2019/775>
- [4] P. Santana, S. Thiébaux, and B. Williams, "Rao*: an algorithm for chance constrained pomdps," in *Proc. AAAI Conference on Artificial Intelligence*, 2016.
- [5] C. Landsiedel and D. Wollherr, "Road geometry estimation for urban semantic maps using open data," *Advanced Robotics*, vol. 31, no. 5, pp. 282–290, 2017.
- [6] E. Levinkov and M. Fritz, "Sequential bayesian model update under structured scene prior for semantic road scenes labeling," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1321–1328.
- [7] Z. Zhang, S. Fidler, and R. Urtasun, "Instance-level segmentation for autonomous driving with deep densely connected mrfs," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 669–677.
- [8] J. Janai, F. Güney, A. Behl, and A. Geiger, "Computer vision for autonomous vehicles: Problems, datasets and state-of-the-art," *arXiv preprint arXiv:1704.05519*, 2017.
- [9] B. Zhao, J. Feng, X. Wu, and S. Yan, "A survey on deep learning-based fine-grained object classification and semantic segmentation," *International Journal of Automation and Computing*, vol. 14, no. 2, pp. 119–135, 2017.
- [10] J. Zhang and S. Sclaroff, "Exploiting surroundedness for saliency detection: a boolean map approach," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 5, pp. 889–902, 2015.
- [11] Q. Wang, Y. Yuan, P. Yan, and X. Li, "Saliency detection by multiple-instance learning," *IEEE transactions on cybernetics*, vol. 43, no. 2, pp. 660–672, 2013.
- [12] S. He, R. W. Lau, and Q. Yang, "Exemplar-driven top-down saliency detection via deep association," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 5723–5732.
- [13] J. Yang and M.-H. Yang, "Top-down visual saliency via joint crf and dictionary learning," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 3, pp. 576–588, 2016.
- [14] J. Pan, E. Sayrol, X. Giro-i Nieto, K. McGuinness, and N. E. O'Connor, "Shallow and deep convolutional networks for saliency prediction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 598–606.
- [15] Y. Xia, D. Zhang, A. Pozdnoukhov, K. Nakayama, K. Zipser, and D. Whitney, "Training a network to attend like human drivers saves it from common but misleading loss functions," *arXiv preprint arXiv:1711.06406*, 2017.
- [16] R. Senanayake, A. Tompkins, and F. Ramos, "Automorphing kernels for nonstationarity in mapping unstructured environments," in *CoRL*, 2018, pp. 443–455.
- [17] H. Hartenstein and K. Laberteaux, *VANET: vehicular applications and inter-networking technologies*. Wiley Online Library, 2010, vol. 1.
- [18] F. G. . S. L. Gante, J., "Deep learning architectures for accurate millimeter wave positioning in 5g," *Neural Process Letters* <https://doi.org/10.1007/s11063-019-10073>, pp. 1–28, 2019.
- [19] A. Houenou, P. Bonnifait, V. Cherfaoui, and W. Yao, "Vehicle trajectory prediction based on motion model and maneuver recognition," in *2013 IEEE/RSJ international conference on intelligent robots and systems*. IEEE, 2013, pp. 4363–4369.
- [20] H. Woo, Y. Ji, H. Kono, Y. Tamura, Y. Kuroda, T. Sugano, Y. Yamamoto, A. Yamashita, and H. Asama, "Lane-change detection based on vehicle-trajectory prediction," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 1109–1116, 2017.
- [21] A. Kendall and Y. Gal, "What uncertainties do we need in bayesian deep learning for computer vision?" in *Advances in neural information processing systems*, 2017, pp. 5574–5584.
- [22] D. Vasquez, T. Fraichard, and C. Laugier, "Growing hidden markov models: An incremental tool for learning and predicting human and vehicle motion," *The International Journal of Robotics Research*, vol. 28, no. 11–12, pp. 1486–1506, 2009.
- [23] J. Wiest, M. Höffken, U. Kreßel, and K. Dietmayer, "Probabilistic trajectory prediction with gaussian mixture models," in *2012 IEEE Intelligent Vehicles Symposium*. IEEE, 2012, pp. 141–146.
- [24] X. Huang, S. McGill, B. C. Williams, L. Fletcher, and G. Rosman, "Uncertainty-aware driver trajectory prediction at urban intersections," *arXiv preprint arXiv:1901.05105*, 2019.
- [25] S. Dong and B. Williams, "Learning and recognition of hybrid manipulation motions in variable environments using probabilistic flow tubes," *International Journal of Social Robotics*, vol. 4, no. 4, pp. 357–368, 2012.
- [26] A. G. Hofmann and B. C. Williams, "Temporally and spatially flexible plan execution for dynamic hybrid systems," *Artificial Intelligence*, vol. 247, pp. 266–294, 2017.
- [27] N. Bhargava and B. C. Williams, "Faster dynamic controllability checking in temporal networks with integer bounds," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*. International Joint Conferences on Artificial Intelligence Organization, 7 2019, pp. 5509–5515. [Online]. Available: <https://doi.org/10.24963/ijcai.2019/765>
- [28] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, "Planning and acting in partially observable stochastic domains," *Artificial intelligence*, vol. 101, no. 1–2, pp. 99–134, 1998.
- [29] E. J. Sondik, "The optimal control of partially observable markov decision processes." *PhD thesis, Stanford University*, 1971.
- [30] P. Poupart, A. Malhotra, P. Pei, K.-E. Kim, B. Goh, and M. Bowling, "Approximate linear programming for constrained partially observable markov decision processes," in *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.
- [31] X. Huang, A. Jasour, M. Deyo, A. Hofmann, and B. C. Williams, "Hybrid risk-aware conditional planning with applications in autonomous vehicles," in *2018 IEEE Conference on Decision and Control (CDC)*. IEEE, 2018, pp. 3608–3614.
- [32] S. Omidshafiei, A.-A. Agha-Mohammadi, C. Amato, and J. P. How, "Decentralized control of partially observable markov decision processes using belief space macro-actions," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 5962–5969.

Cognitively-inspired episodic imagination for self-driving vehicles

Sara Mahmoud¹, Henrik Svensson¹ and Serge Thill^{2,1}

Abstract—The controller of an autonomous vehicle needs the ability to learn how to act in different driving scenarios that it may face. A significant challenge is that it is difficult, dangerous, or even impossible to experience and explore various actions in situations that might be encountered in the real world. Autonomous vehicle control would therefore benefit from a mechanism that allows the safe exploration of action possibilities and their consequences, as well as the ability to learn from experience thus gained to improve driving skills.

In this paper we demonstrate a methodology that allows a learning agent to create simulations of possible situations. These simulations can be chained together in a sequence that allows the progressive improvement of the agent’s performance such that the agent is able to appropriately deal with novel situations at the end of training. This methodology takes inspiration from the human ability to imagine hypothetical situations using episodic simulation; we therefore refer to this methodology as episodic imagination.

An interesting question in this respect is what effect the structuring of such a sequence of episodic imaginations has on performance. Here, we compare a random process to a structured one and initial results indicate that a structured sequence outperforms a random one.

I. INTRODUCTION

A. Simulation abilities in humans

The ability to internally simulate what has or will happen in past and future situations provides agents with increased flexibility when interacting with the world. In humans, these mental simulations occur in many forms, ranging from low-level embodied simulations to higher level episodic simulation [1]. These can briefly be described as follows:

Embodied simulations, in which the sensorimotor systems of the brain are extensively reactivated in similar ways as during overt interaction with the world have been shown to improve subsequent motor performance in, for example, path navigation [2], sports activities [3], and rehabilitation [4]. Thus, embodied simulations seem to facilitate learning despite absence of direct feedback from the environment. *Episodic simulations*, on the other hand, refer to simulations concerning more abstract aspects of interactions not directly affecting motor performance, but rather being more flexible and diverse in terms of the content of the simulations and influencing action selection on a higher level, such as contemplating different places for the next vacation or preparing your arguments in the next salary negotiation, or imagining where you’ll be in 10 years [5], [6].

Funding: the authors acknowledge financial support from the EC H2020 research project Dreams4Cars (no. 731593)

¹Interaction Lab, University of Skövde, 54128 Skövde, Sweden
sara.mahmoud@his.se, henrik.svensson@his.se

²Donders Institute for Brain, Cognition, and Behaviour, Radboud University, 6525 HR Nijmegen, Netherlands s.thill@donders.ru.nl

Another instance of episodic simulations is found in dreams, during which the brain is more or less cut off from sensory input and motor output. Although the function of dreams remains heavily debated, some theories suggest that they might help to prepare agents for action and can improve performance in the wake state. For example, Revonsuo [7] suggests the “Threat Simulation Theory”, according to which a major function of dreams is to rehearse possibly threatening situations. Others have hypothesized that rudimentary mental simulations during early childhood interact with wake behavior to facilitate the formation of more mature mental simulations during development [8], [9]. It follows from this that the importance lies not only in some general reactivation of previous sensorimotor activity, but also in the content since this might influence the usefulness of the simulation for future behaviors.

B. Implementations of simulations in artificial agents

Simulation theories of various kinds have previously also been implemented in various artificial agents to investigate how such an ability affects behavior and can improve performance [10], [11], [12]. For example, an early approach was adopted by Mel [13] who created a robot arm that by means of forward models could plan its movements by “imagining” its future movements. Many other approaches have since then utilized the ability for internal re-creation of sensory and motor states to assist in various tasks (see e.g. [12], [14], [8]). Many of the previous attempts of implementing mental simulations in robots have been rather simplistic and have been more related to embodied simulations rather than episodic simulation due to the nature of the mechanisms used.

The hallmark of episodic simulations is increased flexibility and diversity with regards to the content of the simulations not being dictated to the same degree by the physical constraints of body and environment as would be the case in embodied simulations [1]. Implementing a more strongly biologically inspired mechanism [8], [7] would require such simulations to be more flexible with respect to their content. Deep neural networks [15] and Generative Adversarial Networks (GAN) [16], for example, may provide viable approaches for implementing episodic like simulations in artificial agents. GANs, for example, provide the required flexibility because they are able to create previously unseen data in a useful way. As such, GANs have been used for imagination to generate video scenes similar to collected real world video data, which subsequently was used to run a reinforcement-learning based driving agent in the generated scenes [17]. Initial results showed that a trained GAN

generated simulated images very close to the real data. Thus, rather than recreating very simple sensor data these networks are able to create more episodic like images [18]. In other work, using a setup with a variational auto-encoder combined with a recurrent neural network, Ha and Schmidhuber [19] also showed promising results of using episodic like simulations (or dreams/hallucinations in their terms) in the OpenAI gym [20] and VizDoom [21] environments.

While previous work has put much effort into image generation mechanisms, it is not clear what variables affect the learning process when learning and generating behaviors are instead based on episodic simulations. In particular, it is an open question whether the *structure of the content* of episodic simulations affects the learning performance. This may be critical for autonomous driving [22]. For example, one could vary the number of vehicles encountered when learning to overtake in such simulations, but is it enough, as has been done in previous work, to merely randomly hallucinate different overtaking scenarios [19], [8] until performance converges to a satisfactory level, or should there be some guiding structure to the process?

In the remainder of this paper, we investigate this using a lane-keeping task for an autonomous vehicle. Since the focus is not on the image generation process per se, but on how the content of episodic simulations interacts with the learning process, we here use the rendered simulation in a driving simulator directly as a model for embodied and episodic simulation. The simulation consists of both embodied aspects, such as the physics model of the vehicle, and episodic aspects, such as the type of road environment. However, since the study only manipulates the road environment, we use the term episodic imagination for the test conditions in the study. This allows us to create a tool that is able to flexibly create new episodic-like simulations and focus on the question of how their content may affect the learning and subsequent performance. It should also be noted that the imagination mechanism proposed here differs from the common approach of manually designing the simulations – here, these are automatically generated by the proposed system architecture. The work here thus also contributes to the development of more effective means of learning from imagination by developing an automatic imagination mechanism.

The remainder of the paper is structured as follows: Section II describes the research method. Section III presents the results and Section IV concludes the paper.

II. METHODS

The aim is to evaluate how the structure of the episodic imagination affects the learning performance. We achieve this by training the same Deep Q-network on a lane-keeping task in three different imagination conditions; no imagination, stochastic imagination and systematic imagination, as shown in Figure 1. In the following subsections, the task and conditions are described in more detail.

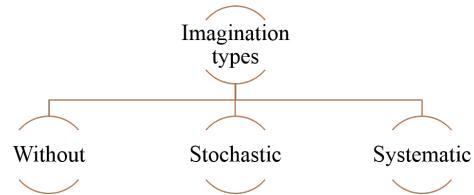


Fig. 1: Imagination types used for training the driving agent: no imagination, stochastic structure and systematic structure

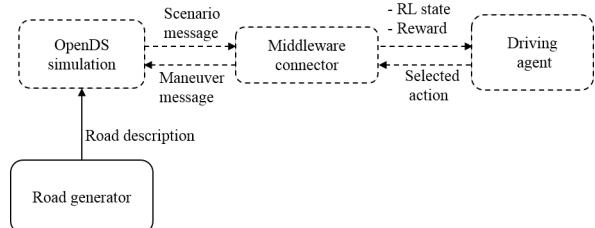


Fig. 2: Episodic generator system architecture for self-driving car on OpenDS simulation

A. System Architecture

In a nutshell, the system architecture consists of four main components (see Figure 2): (1) OpenDS, the physical simulation, in which the training and testing driving is executed (Figure 3), (2) the learning agent that is trained in different imagination conditions, (3) a middleware connector that converts the simulation into a RL environment and (4) the road generator that describes the road specifications used in the simulations.

We use the middleware connector to calculate the reward function (optimized for a lane keeping task) at each step (see Eqns. 1–3). The function depends on the lateral distance from the left lane margin of the road (Eqn. 1), and the car heading angle between the lane and the car (Eqn. 2), as shown in Figure 3b.

$$r_e = \min(d_l, w - d_l) \quad (1)$$

$$r_h = 2 * e^{(-15 * |l_h|)} \quad (2)$$

$$r_t = r_e + r_h \quad (3)$$

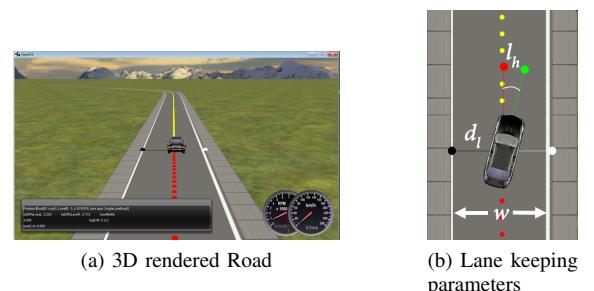


Fig. 3: OpenDS road environment

Where r_e is the reward for the distance from the side of the road, d_l is the distance of the car from the left edge of the road in meters, w is the width of the lane, r_h is the reward for the car heading, l_h is the angle between the car heading and the road heading in radius and r_t is the total reward. In plain terms, the function returns the highest reward when the car is at the middle of the lane and aligned with the road direction.

The Road Generator is a python script that automatically creates the road scenarios based on the defined features. The generator is primarily used for generating episodic imagination. The generator describes the road features and then send the description to OpenDS which construct the described road (Figure 3a). Across all imagination conditions, roads are single lane with a width of six meters. Since the task is lane keeping, the main factors for a road generation are the ratio of straight to curved segments and the geometries of the curves. All roads used in this paper have an approximate length of 500m.

The driving agent in this study is in the form of deep neural networks using Deep-Q-Learning. During training, the driving agent receives a representation of the road through the Middleware Connector which is the RL state of the driving agent in the simulation environment. The driving agent then selects the action from set of available actions (turning the steering wheel 0.05 rad/s to the left, right, or maintaining its current position). The agent receives a reward that represents how good the chosen action is in the given state. The agent updates the network weights based on the obtained experience and continues with this process until it reaches a terminal state, which is arrived at either when the agent successfully reaches the end of the road, or when it leaves the road prematurely.

B. Road Generation Setup

Different types of imagination are used by the agents in the various experimental conditions, as shown in Figure 1. In detail, they are implemented as follows.

1) *No imagination*: A single road is generated, randomly selecting for the number of curves, curvature values and length of components such that the road provides a reasonable variation of the features (so as to not make the training fundamentally impossible), as shown in Figure 4a. Training completes when the agent successfully completes this road 100 times.

2) *Stochastic Imagination*: In this experiment setup the road generator creates 100 different roads before starting the training phase, some examples are shown in Figure 4b in respect to the training order. The parameters determining the complexity of the road are stochastically assigned based on a ratio and within ranges.

During training, the agent iterates to the next road upon successful completion of the current one, and finishes when the agent has successfully trained on all 100 roads. As shown in Figure 4b, the learning agent may start with learning a very curvy road and then after successfully learning this road, the agent moves to an easier road with slight curves.

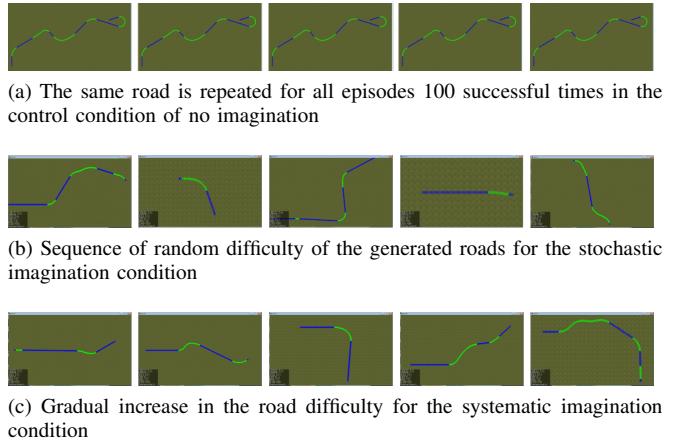


Fig. 4: Road samples of the three imagination conditions

3) *Systematic Imagination*: Systematic Episodic Imagination differs from the previous one in that the difficulty of the roads, quantified based on the number of roads and their curvature, increases during training, some examples are shown in Figure 4c in respect to the training order: the first road consists of 99% straight segments, which then gradually drops until the ratio reaches the previously used [40 curves : 60 straight] after 40 roads. Curvature limits were set to (-0.007 to 0.007 m^{-1}) for the first 40 roads, increasing to (-0.01 to 0.01 m^{-1}) until road 80, and settling at (-0.015 to 0.015 m^{-1}) for the last 20 roads.

III. RESULTS

To measure the effectiveness of the imagination approaches, the three trained driving agents were testing on 100 new roads that they haven't been trained on. The measurement is the mean total rewards that the agent obtains from the testing roads. The theoretical maximum mean total reward is 8500 (if the agent scores reward of five at each of the 1700 steps per episode for the 100 episodes).

In the *No Imagination* condition, which functions as a control in our setup, the agent completed all the 100 testing roads in openDS successfully. The condition resulted in a mean total reward of 6750 (see Fig. 5) which is 79% of the theoretical maximum.

The *Stochastic Imagination* condition performed the worse compared to the two other conditions and failed to finish some testing roads. The mean total rewards for the 100 roads were 5800 (68%).

The agent in this setup with *Systematic Imagination* performed the highest among the other experiments with an mean total rewards of over 7000 (82%). This shows a significant improvement (t-test $p < 0.01$) from the controlled condition.

IV. CONCLUSION

In this paper, we demonstrated how to generate imagination road scenarios for training a self-driving vehicle using physics simulation. The imagination generator generates sequence of episodes with different road features that the

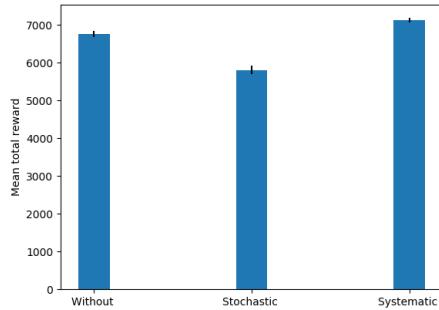


Fig. 5: Mean total rewards at testing phase for the three experimental conditions. Error bar indicate 95% confidence intervals.

driving agent needs to learn. This work doesn't focus on the agent's optimization but on the scenario generation as a learning environment. The paper presents two ways of generating the sequence of episodes as either stochastic or systematic and compares the learning performance for each. A controlled condition is no imagination which means various of features are collected in a single road and the training is conducted on this road. The results showed that even for a relatively simple task, the structure of the imagination has an impact on learning performance. These results are also in line with theories of human episodic simulation, in particular the observation that human dreams increase in complexity during development [9], suggesting that there is a benefit to bio (and cognitively) inspired approaches in this domain.

A lot of research has been put into investigations of how to design and optimize various learning agents, much less efforts have focused on the environment. This work shows that the structure of the environment plays a considerable role in learning. For future work, additional investigation can be done to mathematically analyze how the episodic structure contributes to the learning performance. Besides, further mechanisms can be proposed to improving the criteria of the episodic generation. For example, continuously assess the agent's learning performance and accordingly generate the suitable episodic imagination that the agent actually needs.

REFERENCES

- [1] H. Svensson and S. Thill, "Beyond bodily anticipation: Internal simulations in social interaction," *Cognitive Systems Research*, vol. 40, pp. 161–171, 2016. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1389041716300894>
- [2] S. Vieilledent, S. M. Kosslyn, A. Berthoz, and M. D. Giraldo, "Does mental simulation of following a path improve navigation performance without vision?" *Cognitive Brain Research*, vol. 16, no. 2, pp. 238 – 249, 2003. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0926641002002793>
- [3] A. Guillot and C. Collet, "Duration of mentally simulated movement: a review," *Journal of motor behavior*, vol. 37, no. 1, pp. 10–20, 2005. [Online]. Available: <http://www.tandfonline.com/doi/abs/10.3200/JMBR.37.1.10-20>
- [4] J. Munzert, B. Lorey, and K. Zentgraf, "Cognitive motor processes: the role of motor imagery in the study of motor representations," *Brain research reviews*, vol. 60, no. 2, pp. 306–326, 2009. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0165017309000022>
- [5] R. L. Buckner and D. C. Carroll, "Self-projection and the brain," *Trends in cognitive sciences*, vol. 11, no. 2, pp. 49–57, 2007. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1364661306003275>
- [6] K. K. Szpunar, R. N. Spreng, and D. L. Schacter, "A taxonomy of prospection: Introducing an organizational framework for future-oriented cognition," *Proceedings of the National Academy of Sciences*, vol. 111, no. 52, pp. 18 414–18 421, 2014. [Online]. Available: <http://www.pnas.org/content/111/52/18414.short>
- [7] A. Revonsuo, "The reinterpretation of dreams: An evolutionary hypothesis of the function of dreaming," *Behavioral and Brain Sciences*, vol. 23, no. 6, pp. 877–901, 2000.
- [8] H. Svensson, S. Thill, and T. Ziemke, "Dreaming of electric sheep? exploring the functions of dream-like mechanisms in the development of mental imagery simulations," *Adaptive Behavior*, vol. 21, no. 4, pp. 222–238, 2013.
- [9] S. Thill and H. Svensson, "The inception of simulation : a hypothesis for the role of dreams in young children," in *Proceedings of the 33rd Annual Conference of the Cognitive Science Society* .: Cognitive Science Society, Inc., 2011, pp. 231–236.
- [10] T. Ziemke, D.-A. Jirehned, and G. Hesslow, "Internal simulation of perception: a minimal neuro-robotic model," *Neurocomputing*, vol. 68, pp. 85–104, Oct. 2005.
- [11] H. Hoffmann and R. Müller, "Action selection and mental transformation based on a chain of forward models," *From Animals to Animats*, vol. 8, pp. 213–222, 2004.
- [12] E. Billing, H. Svensson, R. Lowe, and Z. Tom, "Finding Your Way from the Bed to the Kitchen: Re-enacting and Re-combining Sensorimotor Episodes Learned from Human Demonstration," *Frontiers in Robotics and AI*, vol. 3, no. 9, 2016. [Online]. Available: <http://www.diva-portal.org/smash/record.jsf?pid=diva2:915452>
- [13] B. W. Mel, "A connectionist model may shed light on neural mechanisms for visually guided reaching," *Journal of cognitive neuroscience*, vol. 3, no. 3, pp. 273–292, 1991.
- [14] Y. Demiris and B. Khadhouri, "Hierarchical attentive multiple models for execution and recognition of actions," *Robotics and autonomous systems*, vol. 54, no. 5, pp. 361–369, 2006.
- [15] J. Schmidhuber, "Deep learning in neural networks: An overview," vol. 61, pp. 85–117, 2015.
- [16] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.
- [17] E. Santana and G. Hotz, "Learning a driving simulator," *arXiv preprint arXiv:1608.01230*, 2016.
- [18] A. Plebe, R. Don, G. P. Rosati, and M. D. Lio, "Mental imagery for intelligent vehicles," in *Proceedings of the 5th International Conference on Vehicle Technology and Intelligent Transport Systems - Volume 1: VEHTIS*, INSTICC. SciTePress, 2019, pp. 43–51.
- [19] D. Ha and J. Schmidhuber, "World models," *arXiv preprint arXiv:1803.10122*, 2018.
- [20] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, "Openai gym," 2016.
- [21] M. Kempka, M. Wydmuch, G. Runc, J. Toczek, and W. Jaśkowski, "Vizdoom: A doom-based ai research platform for visual reinforcement learning," in *2016 IEEE Conference on Computational Intelligence and Games (CIG)*. IEEE, 2016, pp. 1–8.
- [22] M. D. Lio, A. Mazzalai, D. Windridge, S. Thill, H. Svensson, M. Yksel, K. Gurney, A. Saroldi, L. Andreone, S. R. Anderson, and H. J. Heich, "Exploiting dream-like simulation mechanisms to develop safer agents for automated driving," in *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, Oct. 2017, pp. 1–6.

A Cognitively Informed Perception Model for Driving

Alice Plebe¹ and Mauro Da Lio²

Abstract—Deep learning is responsible for the current renewed success of artificial intelligence. Applications that in the recent past were considered beyond imagination, now appear to be feasible. The best example is autonomous driving. However, despite the growing research aimed at implementing autonomous driving, no artificial intelligence can claim to have reached or closely approached the driving performance of humans, yet. Deep learning is an evolution of artificial neural networks introduced in the '80s with the *Parallel Distributed Processing* (PDP) project. There is a fundamental difference in aims between the first generation of artificial neural networks and deep neural models. The former was motivated primarily by the exploration of cognition. Current deep neural models are instead developed with engineering goals in mind, without any ambition or interest in exploring cognition. Some important components of deep learning – for example reinforcement learning or recurrent networks – owe indeed an inspiration to neuroscience and cognitive science, as PDP far legacy. But this connection is now neglected, what matters is only the pragmatic success in applications. We argue that it urges to reconnect artificial modeling with an updated knowledge of how complex tasks are realized by the human mind and brain. In this paper, we will first try to distill concepts within neuroscience and cognitive science relevant for the driving behavior. Then, we will identify possible algorithmic counterparts of such concepts, and finally build an artificial neural model exploiting these components for the visual perception task of an autonomous vehicle.

I. FROM THE COGNITIVE SIDE

A. The Simulation Theory

A well-established theory in cognitive science is the one proposed by Jeannerod and Hesslow, the so-called *simulation theory of cognition*, which proposes that thinking is essentially a simulated interaction with the environment [1], [2]. In their view, simulation is a general principle of cognition, which can be expressed in at least three different components: perception, actions and anticipation.

The most simple case of simulation is mental imagery, especially in visual modality. This is the case, for example, when a person tries to picture an object or a situation. During this phenomenon, the primary visual cortex (V1) is activated with a simplified representation of the object of interest, but the visual stimulus is not actually perceived.

This work was developed inside the EU Horizon 2020 Dreams4Cars Research and Innovation Action project, supported by the European Commission under Grant 731593. The Authors want also to thank the Deep Learning Lab at the ProM Facility in Rovereto (TN) for supporting this research with computational resources funded by Fondazione CARITRO.

¹ Alice Plebe is with the Dept. of Information Engineering and Computer Science, University of Trento, Italy alice.plebe@unitn.it

² Mauro Da Lio is with the Dept. of Industrial Engineering, University of Trento, Italy mauro.dalio@unitn.it

B. Convergence–Divergence Zones

Although the simulation theory is one of the most established, it does not identify how simulation takes place at neural level. A prominent proposal in this direction is the formulation of the convergence-divergence zones (CDZs) [3]. They highlight the “convergent” aspect of certain neuron ensembles, located downstream from primary sensory and motor cortices. Such convergent structure consists in the projection of neural signals on multiple cortical regions in a many-to-one fashion. On the other hand, the neuron ensembles have the ability to reciprocate feedforward projections with feedback projections in a one-to-many fashion, realizing the divergent flow.

The primary purpose of convergence is to exploit synaptic plasticity in order to record which patterns of features – coded as knowledge fragments in the early cortices – occur in relation with a specific higher-level concept. Such records are built through experience, by interacting with objects. The convergent flow is dominant during perceptual recognition, while the divergent flow dominates imagery.

Convergent-divergent connectivity patterns can be identified for specific sensory modalities, but also in higher order association cortices. It should be stressed that CDZs are rather different from a conventional processing hierarchy, where processed patterns are transferred from earlier to higher cortical areas. In CDZs, part of the knowledge about perceptual objects is retained in the synaptic connections of the convergent-divergent ensemble. This allows to reinstate an approximation of the original multi-site pattern of a recalled object or scene.

C. Transformational Abstraction

One major challenge in cognitive science is explaining the mental mechanisms by which we build conceptual abstractions. The conceptual space is the mental scaffolding the brain gradually learns through experience, as internal representation of the world. In particular, conceptual abstraction is derived mostly from perceptual experience, which fits perfectly with the approach implemented by artificial neural networks.

As highlighted by [4] CDZs are a valid systemic candidate for how the formation of high-level concepts takes place at brain level. However, the idea of CDZs is just sketched and cannot provide a detailed mechanism for conceptual abstractions. A difficulty with acquiring abstract categories lies in the inconsistent manifestations of the characteristic features across real exemplars.

A suggested solution to this difficult issue is the *transformational abstraction* [5], [6] performed by a hierarchy

of cortical operations, as in the ventral visual cortex. The essence of transformational abstraction, from a mathematical point of view, lies in the combination of two operations: linear convolutional filtering and nonlinear downsampling. Operations of this sort have been identified in the V1 [7], [8], and are well recognized in the primate ventral visual path as well [9], [10].

D. The Predictive Theory

The reason why cognition is mainly explicated as simulation, according to Hesselow or Jeannerod, is because the brain can achieve through simulation the most precious information of an organism: a prediction of the state of affairs in the future environment. The need of prediction, and how it molds the entire cognition, has become the core of another popular theory popular known as “Bayesian brain”, “predictive brain”, or “free-energy principle for the brain” introduced by Friston [11]. According to him the behavior of the brain – and of an organism as a whole – can be conceived as minimization of free-energy, a quantity that can be expressed in several ways depending on the kind of behavior and the brain systems involved.

Free-energy is a concept originated in thermodynamics, as a measure of the amount of work that can be extracted from a system. What is borrowed by Friston is not the thermodynamic meaning of the free-energy, but its mathematical form only, which is derived from the framework of variational Bayesian methods in statistical physics. We will see in §II-B how the same probabilistic framework will be used in the derivation of a deep neural model. For example, this is his free-energy formulation in the case of perception [12, p.427]:

$$F_P = \Delta_{\text{KL}}(\check{p}(\mathbf{c}|\mathbf{z}) \| p(\mathbf{c}|\mathbf{x}, \mathbf{a})) - \log p(\mathbf{x}|\mathbf{a}) \quad (1)$$

where \mathbf{x} is the sensorial input of the organism, \mathbf{c} is the collection of the environmental causes producing \mathbf{x} , \mathbf{a} are actions that act on the environment to change sensory samples, and \mathbf{z} are inner representations of the brain. The quantity $\check{p}(\mathbf{c}|\mathbf{z})$ is the encoding in the brain of the estimate of causes of sensorial stimuli. The quantity $p(\mathbf{c}|\mathbf{x}, \mathbf{a})$ is the conditional probability of sensorial input conditioned by the actual environmental causes \mathbf{c} . The discrepancy between the estimated probability and the actual probability is given by the Kullback-Leibler divergence Δ_{KL} . The minimization of F_P in equation (1) optimizes \mathbf{z} .

II. TO THE ARTIFICIAL SIDE

A. Convergence–divergence as Autoencoder

In the realm of artificial neural networks, the computational idea that most closely resonate with CDZ is the *autoencoder*. It is an idea that has been around for a long time, it was the cornerstone of the evolution from shallow to deep neural architectures [13], [14]. More recently, autoencoders have been widely adopted for their ability to capture compact information from high dimensional data. The basic structure of an autoencoder is composed of a feature-extracting part called *encoder* and a *decoder* part mapping from feature

space back into input space. There is a clear correspondence between the encoder and the convergence zone in the CDZ neurocognitive concept, and similarity between the decoder and the divergence zone.

Then how exactly convergence–divergence can be achieved inside autoencoders? An interesting approach is the one closely related to the transformational abstraction hypothesis described in §I-C: the *deep convolutional neural networks* (DCNNs). They implement the hierarchy of convolutional filtering alternated with nonlinear downsampling, and are considered the essence of transformational abstraction. In addition, there is growing evidence of striking analogies between patterns in DCNN models and patterns of voxels in the brain visual system. Several studies have successfully related results of deep learning models with the visual system [15], [16], finding reasonable agreement between features computed by DCNN models and fMRI data. Convolutional–deconvolutional autoencoders are therefore a highly biologically plausible implementation for the CDZ theory, at least in the case of visual information.

B. Predictive Brain as Variational Autoencoder

In the last few years there has been renewed interest in the area of Bayesian probabilistic inference in learning models of high dimensional data. The Bayesian framework, variational inference in particular, has found a fertile ground in combination with neural models. Two concurrent and unrelated developments [17], [18] have made this theoretical advance possible, connecting autoencoders and variational inference. This new approach became quickly popular under the term *variational autoencoder*, and a variety of neural models have been proposed over the years.

The loss function for a variational autoencoder is defined as follows:

$$\begin{aligned} \mathcal{L}(\Theta, \Phi | \mathbf{x}) = & \Delta_{\text{KL}}(q_\Phi(\mathbf{z}|\mathbf{x}) \| p_\Theta(\mathbf{z})) + \\ & - \mathbb{E}_{\mathbf{z} \sim q_\Phi(\mathbf{z}|\mathbf{x})} [\log p_\Theta(\mathbf{x}|\mathbf{z})] \end{aligned} \quad (2)$$

where \mathbf{x} is a high dimensional random variable, \mathbf{z} the representation of the variable in the low-dimensional latent space. Θ and Φ are parameters describing, respectively, the decoder and encoder of the network. p_Θ is computed by the decoder and represents the desired approximation of the unknown input distribution p , and q_Φ is the auxiliary distribution computed by the encoder from which to sample \mathbf{z} . $\mathbb{E}[\cdot]$ is the expectation operator, and Δ_{KL} is the Kullback-Leibler divergence.

It is evident how this mathematical formulation is impressively similar to the concept of free energy in Friston. Despite this close analogy, all the proposers of variational autoencoder are either unaware or fully disinterested of this coincidence. It is not so surprising because mainstream deep learning is driven by engineering goals without any interest in connections with cognition. We believe instead that a strong connection between a well established cognitive theory and a computational solution greatly argues in favor of adopting such a solution.

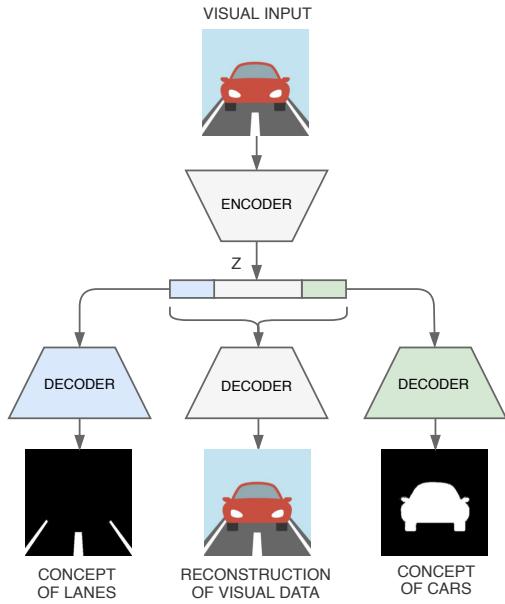


Fig. 1. The architecture of our model.

III. IMPLEMENTATION

In the previous section we have reviewed several components that match quite closely the relevant neurocognitive theories identified in §I. Our proposed model attempts to weave together these components, finalized at visual perception in autonomous driving agents.

Similarly to the hierarchical arrangement of CDZs in the brain, our model is provided with different levels of processing paths. A first processing path starts from the raw image data and converges up to a low-dimension representation of visual features. Consequently, the divergent path outputs in the same format as the input image. The other processing path leads to representations that are no more in terms of visual features, rather in terms of concepts. As discussed in §I-C, our brain naturally projects sensorial information – especially visual – into conceptual space, where the local perceptual features are pruned and neural activation code the nature of entities present in the environment that produced the stimuli. In the driving context it is not necessary to infer categories for every entity present in the scene, it is useful to project in conceptual space only the objects relevant to the driving task. In the model presented here we choose to consider the two main concepts of cars and lane markings.

As depicted in Fig. 1, the presented variational autoencoder is composed by one shared encoder and three independent decoders. All the components of the architecture are trained jointly. The encoder compresses an RGB image to a compact high-feature representation. Then the decoders map different part of the latent space back to separated output spaces: one into the same visual space of the input; the other two into conceptual space, producing binary images containing, respectively, car entities and lane marking entities.

So, in our implementation the entire latent vector \mathbf{z} represents inside the visual space, and at the same time two inner segments represent specifically the car and lane concepts. The rationale for this choice is that in mental imagery there is no clear cut distinction between low-level features and semantic features, the entire scene is mentally reproduced, but including the awareness of the salient concepts present in the scene.

Note that the idea of partitioning the entire latent vector into meaningful components is not new. In the context of processing human heads the vector has been forced to encode separate representations for viewpoints, lighting conditions, shape variations [19]. In [20] the latent vector is partitioned in one segment for the semantic content and a second segment for the position of the object. Our approach is different. While we keep disjointed the two segments for the car and lane concepts, we fully overlap these two representations within the entire visual space. This way, we adhere entirely to the CDZ principle, and try to achieve the full scene by divergence, but at the same time including awareness for the car and lane concepts.

IV. RESULTS

We present here a selection of results achieved with an instance of the model described in the previous section. The final architecture is trained for 200 epochs, and used 4 convolutional layers in the encoder, 4 deconvolutional layers for each decoder, and a latent space representation of 128 neurons, of which 16 encoding the car concept and another 16 for the lane marking concept. We would like to highlight that, since the images fed to the network have dimension of $256 \times 128 \times 3$ and the latent space dimension is 128, the compression performed by the network is almost of 4 orders of magnitude. This is a considerable achievement compared to other relevant works adopting variational autoencoder [21], [22] which limit the compression of the encoder to only 1 order of magnitude.

We trained and tested the presented model on the SYNTHIA dataset [23], a large collection of synthetic images representing various urban scenarios. The dataset contains about 100,000 color images (and as many corresponding segmented images, used for ground truth of the conceptual branches of the network). We used 70% of the data for training, 25% for validation and 5% for testing.

Fig. 2 shows the image results produced by our model for a selection of driving scenarios. The images are processed to better show at the same time the results on conceptual space and visual space. The colored overlays highlight the concepts computed by the network: the cyan regions are the output of the car divergent path, and the pink overlays are the output of the lane markings divergent path. Fig. 2 includes a variety of driving situations, going from sunny environments (top rows) to very adverse driving conditions (bottom rows) in which the detection of other vehicles can be challenging even for a human. These results nicely show how the projection of the sensorial input (original frames) into conceptual representation is very effective in identifying

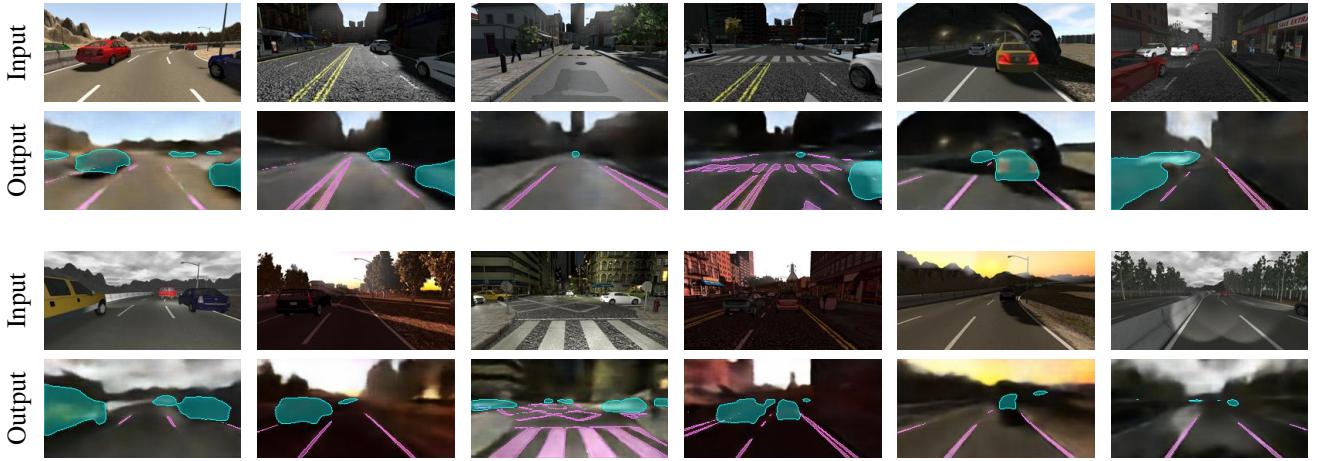


Fig. 2. Results of our model for a selection of frames from the SYNTHIA dataset, with different environmental and lighting conditions.

and preserving the sensible features of cars and lane markings, despite the large variations in lighting and environmental conditions.

Lastly, we would like to stress that the purpose of our network is not mere segmentation of visual input. The segmentation task is to be considered as a support task, used to enforce the network to learn a more robust latent space representation, which now is explicitly taking into consideration two of the concepts that are fundamental to the driving tasks.

V. CONCLUSIONS

The model here presented is an attempt to convert into an artificial neural network model the fundamental theories about how the brain processes its sensory inputs to produce purposeful representations. We especially identified the consolidated variational autoencoder architecture as the best candidate for implementing convergence-divergence zone schemes. The reason for constraining a deep learning model on cognitive theoretical grounds, instead of starting from scratch as often done, derives from the observation of how humans excel in sophisticated sensorimotor control tasks such as driving.

REFERENCES

- [1] M. Jeannerod, “Neural simulation of action: A unifying mechanism for motor cognition,” *NeuroImage*, vol. 14, pp. S103–S109, 2001.
- [2] G. Hesslow, “The current status of the simulation theory of cognition,” *Brain*, vol. 1428, pp. 71–79, 2012.
- [3] K. Meyer and A. Damasio, “Convergence and divergence in a neural architecture for recognition and memory,” *Trends in Neuroscience*, vol. 32, pp. 376–382, 2009.
- [4] J. S. Olier, E. Barakova, C. Regazzoni, and M. Rautenberg, “Reframing the characteristics of concepts and their relation to learning and cognition in artificial agents,” *Cognitive Systems Research*, vol. 44, pp. 50–68, 2017.
- [5] D. Hassabis, D. Kumaran, C. Summerfield, and M. Botvinick, “Neuroscience-inspired artificial intelligence,” *Neuron*, vol. 95, pp. 245–258, 2017.
- [6] C. Buckner, “Empiricism without magic: transformational abstraction in deep convolutional neural networks,” *Synthese*, vol. 195, pp. 5339–5372, 2018.
- [7] D. Hubel and T. Wiesel, “Receptive fields, binocular interaction, and functional architecture in the cat’s visual cortex,” *Journal of Physiology*, vol. 160, pp. 106–154, 1962.
- [8] C. D. Gilbert and T. N. Wiesel, “Morphology and intracortical projections of functionally characterised neurones in the cat visual cortex,” *Nature*, vol. 280, pp. 120–125, 1979.
- [9] D. J. Felleman and D. C. Van Essen, “Distributed hierarchical processing in the primate cerebral cortex,” *Cerebral Cortex*, vol. 1, pp. 1–47, 1991.
- [10] D. C. Van Essen, “Organization of visual areas in macaque and human cerebral cortex,” in *The Visual Neurosciences*, L. Chalupa and J. Werner, Eds. Cambridge (MA): MIT Press, 2003.
- [11] K. Friston, “The free-energy principle: a unified brain theory?” *Nature Reviews Neuroscience*, vol. 11, pp. 127–138, 2010.
- [12] K. Friston and K. E. Stephan, “Free-energy and the brain,” *Synthese*, vol. 159, pp. 417–458, 2007.
- [13] G. E. Hinton and R. R. Salakhutdinov, “Reducing the dimensionality of data with neural networks,” *Science*, vol. 28, pp. 504–507, 2006.
- [14] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, “Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion,” *Journal of Machine Learning Research*, vol. 11, pp. 3371–3408, 2010.
- [15] U. Güçlü and M. A. J. van Gerven, “Unsupervised feature learning improves prediction of human brain activity in response to natural images,” *PLoS Computational Biology*, vol. 10, pp. 1–16, 2014.
- [16] B. P. Tripp, “Similarities and differences between stimulus tuning in the inferotemporal visual cortex and convolutional networks,” in *International Joint Conference on Neural Networks*, 2017, pp. 3551–3560.
- [17] D. P. Kingma and M. Welling, “Auto-encoding variational bayes,” in *Proceedings of International Conference on Learning Representations*, 2014.
- [18] D. J. Rezende, S. Mohamed, and D. Wierstra, “Stochastic backpropagation and approximate inference in deep generative models,” in *Proceedings of Machine Learning Research*, E. P. Xing and T. Jebara, Eds., 2014, pp. 1278–1286.
- [19] T. D. Kulkarni, W. F. Whitney, P. Kohli, and J. B. Tenenbaum, “Deep convolutional inverse graphics network,” in *Advances in Neural Information Processing Systems*, 2015, pp. 2539–2547.
- [20] J. Zhao, M. Mathieu, R. Goroshin, and Y. LeCun, “Stacked what-where auto-encoders,” in *International Conference on Learning Representations*, 2016, pp. 1–12.
- [21] E. Santana and G. Hotz, “Learning a driving simulator,” *CoRR*, vol. abs/1608.01230, 2016.
- [22] D. Ha and J. Schmidhuber, “World models,” *CoRR*, vol. abs/1803.10122, 2018.
- [23] G. Ros, L. S. J. M. D. Vazquez, and A. M. Lopez, “The SYNTHIA dataset: A large collection of synthetic images for semantic segmentation of urban scenes,” in *Proc. of IEEE International Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3234–3243.

Cognitive Wheelchair: A Personal Mobility Platform

Mahendran Subramanian, Suhyung Park, Pavel Orlov, and A. Aldo Faisal

Abstract— Cognitive technologies towards smart vehicles are evolving, but specific to wheelchairs are often overlooked. High-level gaze informatics can be complemented with context-aware algorithms for natural interaction with the environment through robotic devices or autonomous vehicles. Herein, we harvest the eye movements to enhance the cognitive abilities of an autonomous wheelchair platform as eye movements are correlated with motor intention and act as a precursor to movement. First, we developed tools to estimate 3D gaze point and to recognise the object at the wheelchair user's 'Area Of Interest' for high-level intention decoding. This 3D eye-tracking tool with practical accuracy levels to analyse natural human behaviour during physical interaction with the environment was used to obtain human intention in autonomous systems for natural, user-centric interaction with the environment. With high-level intention decoding capability, such cognitive wheelchair systems can not only perform autonomous mobility tasks but also function in a contextual, semantic and continuous spectrum of tasks.

I. INTRODUCTION

Autonomous navigation and context-aware algorithms have been studied within the framework of autonomous wheelchairs for urban mobility [1-3]. These algorithms most often make use of the radar, lidar, RGB-Depth and ultrasonic proximity sensors to reconstruct the map of the environment to compute the path to reach a predefined destination. However, the Artificial Intelligence (AI) algorithms employed in these applications work in a closed-loop manner, where the algorithms adapt to the environment directly based on the sensor data. Such an approach is expected to work well in known environments, which have well-defined conditions like traffic systems and landmarks for GPS based navigation. However, wheelchairs' operation does not have predefined conditions, as the users have a more diverse range of needs and environments that they need to move in. For this, the human intention needs to be included in the AI-environment loop, so that the wheelchair can understand users' intentions while monitoring for environment conditions through the sensor data. Through such semi-autonomy, users can provide one-time goal-based commands rather than continuous directional commands, freeing them from the constant interaction with the interface, while AI algorithms can take care of the navigation. Natural gaze-based intention decoding is worthy of consideration [4,5]. Herein, we incorporate autonomous driving technology with gaze-based intention decoding. Specifically, we demonstrate how gaze information can be

* This work was supported by EPSRC [EP/N509486/1: 1979819] and Toyota Mobility Discovery Grant. MS, SP, PO, and AAF are with the Brain & Behaviour Lab, Dept. of Bioengineering & Dept. of Computing, Imperial College London, South Kensington, SW7 2AZ, UK (Corresponding author email: a.faisal@imperial.ac.uk). We sincerely thank Paul Moore and Tom Nebarro – Our wheelchair pilots.

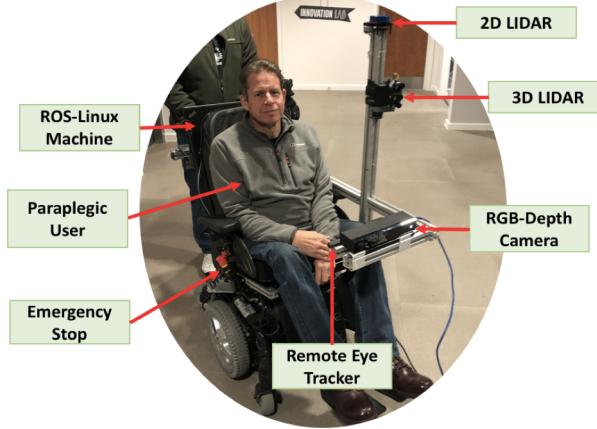


Figure 1. Wheelchair setup with a paraplegic pilot.

translated into user intention for an autonomously driving wheelchair, which essentially leads to a human-in-the-loop cognitive wheelchair. Our patient-centred approach considerably reduces the need for constant interaction with an interface and lowers their cognitive load. With improved context-aware algorithms, this cognitive wheelchair can recognise the objects and surface conditions in the proximity of the wheelchair and can work with high-level information like semantically annotated locations and more natural eye-gaze interaction. This empowers users to communicate or interact with the environment while navigating the wheelchair. Our cognitive, personal mobility technology makes the wheelchair control contextually responsive to the dynamic environment of the user and potentially enables us to navigate an urban continuum from room to city scale.

II. MATERIAL AND METHODS

A. System Hardware Architecture

A 2D lidar (YDLIDAR X4) and an own custom developed lidar, i.e. 3D lidar was fitted onto a powered wheelchair (Invacare) to fabricate an autonomous wheelchair platform, as shown in Fig. 1. The control system on the powered wheelchair was replaced with a regenerative dual channel motor driver (H-bridge from Dimension Engineering) to control the wheelchair's motors. The prototype is fitted with a RGB-D camera (RealSense D435 – 1280 x 720 @ 90fps; 10 m range; $87^\circ \pm 3^\circ$ H x $58^\circ \pm 1^\circ$ V x $95^\circ \pm 3^\circ$ D-angle of view or Kinect v2 – 512 x424 @ 30fps; 0.5-4.5 m range; 70/60 angle of view). RGB-D camera in combination with Tobii Eye 4c remote eye tracker was used for 3D gaze estimation. The RGB feed in the SMI wearable eye-tracking spectacles for the ego-centric view was used to perform 2D gaze point objection identification and simultaneously record eye movements.

B. Gazeinformatics-based intention decoding

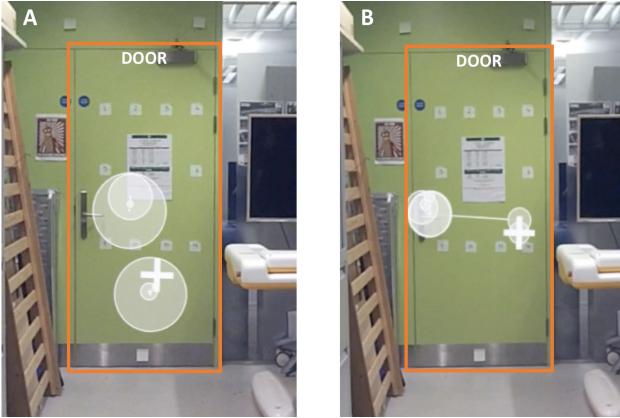


Figure 2. Labelled ego-centric image with gaze fixation and fixation duration – White cross and white circle. (A) ‘Observing a door’ (Non-interactive task). (B) ‘Open the door’ (Interactive task). The bounding boxes of the same objects have different sizes and proportions from a different point of view.

This AI module decodes users' high-level intention such as "Take me to the Door" by investigating their eye movements. To collect data for model training and evaluation, we performed a study with five healthy subjects. During the experimental session, each subject performs 10 trials. We propose meta-tasks, in which participants are shown a door <targetObject> and are asked to perform different actions with them (see Fig. 2.). A computer provides voice commands based on these tasks for each subject, and the gaze tracking software tool (BeGaze, SensoMotoric Instruments) records the corresponding eye movements. 3-point calibration was performed prior to recording. The meta-tasks include tasks such as look at the open door; look at the door and imagine opening the door, viz.

Macro eye-movement events, like gaze fixations, fixation duration, saccades, and long drifts have been used previously to model subjects' actions [6,7] and for the development of gaze controlled systems [8-10]. However, during their everyday activity, humans use all available information for motor planning, for example, information from the memory, context, and visual information from the foveal and extrafoveal area. In our system, we used raw eye- tracker data with a sampling frequency of 120Hz. Fig. 2. illustrates the bounding boxes that we used for object labelling using AOI tool within BeGaze. The size of the bounding box corresponds with the object's pixels on the current ego-centric image. From one image to another, the pixel area that is drawn for a single object might change because of head movement, different object locations, and different viewing points (see Fig. 2., e.g. the door). That is why we average bounding boxes per object and normalise gaze point position with respect to this. We have collected 3718 gaze points (from 5 subjects). Out of which ~50 % gaze fixations were related to non-interactive meta tasks and were labelled as Class 0. Gaze fixations related to interactive meta tasks were labelled as Class 1. In order to predict a high-level intention, we used visual attention density to determine intention. To this end, we used a simple but robust approach: gaze locations within the object bounding box were normalised. This 2D location was fed to an object-specific SVM classifier that was trained on the two classes (10-fold cross-validated). Each frame classification output was fed into

a ring-buffer of 40 frames, and we performed a winner-take-all vote to determine the so temporally averaged intention. The above process was done offline, but, in order to achieve real-time intention decoding, Semantic Fovea was used to compute object label and bounding box [11,12]. An evaluation of our real-time gazeinformatics based intention decoding module is under review [13].

C. 3D Gaze-based destination definition

3D gaze estimation is a suitable technique for end-point control [14]. Herein, in order to determine the 3D coordinates of the user's intended destination, we combine remote eye-trackers with an RGB-D camera. The remote eye-trackers, placed at a distance of 60 cm from the user, can track users gaze and provide 2D screen coordinates of the gaze on a 60 cm x 34 cm display at a rate of 60Hz. To convert this 2D information into 3D coordinates, we overlay 2D gaze-point of the user on the 3D point cloud map of the environment reconstructed by the RGB-D camera. Calibration process and the accuracy of the 3D gaze-point estimation, along with absolute errors in three dimensions as well as the Euclidean distance error are reported in our earlier study [15].

D. System architecture

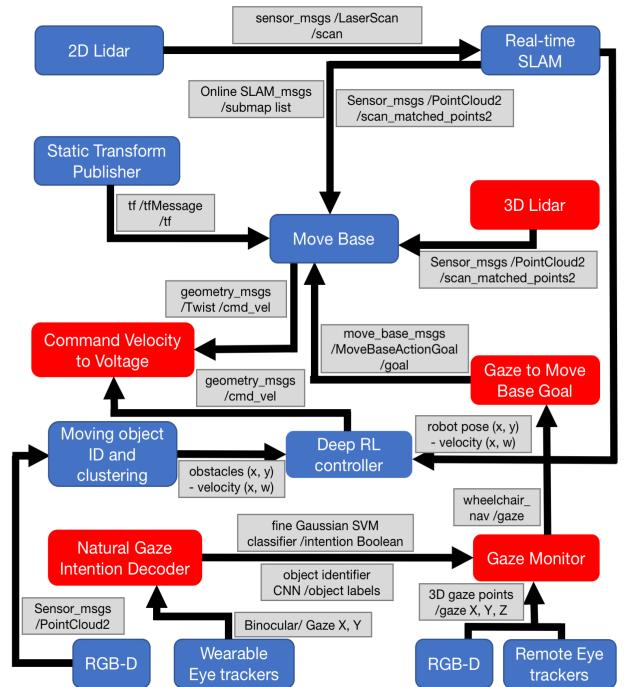


Figure 3. Diagram showing the system architecture of the semi-autonomous wheelchair prototype. The nodes in Red are native to this architecture. Grey boxes show the ROS message type.

The ROS based autonomous wheelchair platform (Fig. 3.) was built on a 2D real-time SLAM algorithm [16]. Home built 3D lidar was used for obstacle detection. Navigation_stack in combination with a GPU – Asynchronous Advantage Actor-Critic (G-A3C) based Reinforcement Learning (RL) controller trained for collision avoidance in a small unmanned ground vehicle [17] was utilised for path planning, obstacle detection and collision avoidance. The autonomous navigation architecture was then incorporated with the gaze-based destination commands. The natural_gaze_intention_decoder node publishes predictor_msgs and

`object_identifier_msgs`, i.e. whether the user intends to interact with the object of interest within the field of view. The `gaze_monitor` node subscribes to these messages, and the gaze-based commands are published in ROS message (`wheelchair_nav/gaze`) received from Windows client. 3D gaze-based end-point control ('Wink' detection) was used for low-level intention (free navigation) input as well as a double confirmation of the decoded high-level intention. The `gaze_to_move_base_goal` node subscribes to this message and publishes the goal pose as a `move_base_msg` to the `move_base` node. Once a path to the goal has been computed, the required velocity commands are sent to the `cmd_vel_to_wheelchair_drive` node, which is the driver for the wheel motors, and all of these processes were achieved in real-time.

III. RESULTS AND DISCUSSION

A. SLAM generated map evaluation and pose estimation

SLAM generated map was evaluated by superimposing the resulting map on top of the blueprint of the scanned floor. It did form fit. To evaluate the fit, area (pixel) comparison was performed (Image J, NIH, USA) between the SLAM generated-map and the floor plan after adjusting their scale to be uniform. Floor plan section area was 37202 px; SLAM derived Map area for the same section was 38383 px. The additional area 1181 px detected was due to the laser scan penetrating through windows and fitted glasses. Which infer that the SLAM derived map accuracy is 96.8 %, i.e. the discrepancy is 3.1%. However, comparison of surface area between a rectangle sized room (2 x 4 m) and a SLAM derived-map of the same room resulted in 95% accuracy.

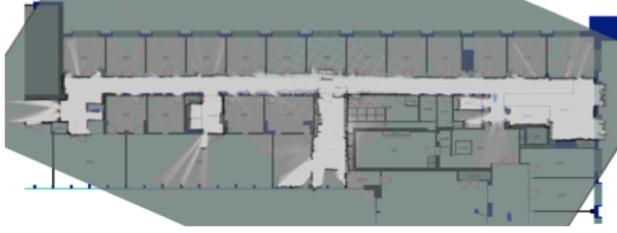


Figure 4. SLAM derived map form fit with the floor plan (Floor-4, Royal School of Mines, Imperial College London)

Similarly, calculation of SLAM derived pose accuracy was evaluated by measuring the room coordinates with respect to the wheelchair and comparing it with the SLAM derived map and localisation coordinates. Localisation pose accuracy measured had a tolerance of +/- 10 cm. However, the pose update frequency parameter had to be set to low, as higher values resulted in a constant oscillation of the pose within the tolerance.

B. Autonomous Wheelchair Performance Evaluation

In order to find the optimal specifications for safe use of the wheelchair, we investigated three parameters: planner frequency, position tolerance and orientation tolerance. Different values for the planner frequency were investigated to optimise the pose based path update during navigation and understand their effect on wheelchair transmission. These parameters were first evaluated in three different tasks by measuring the time it takes the wheelchair to reach the destination. In all three tasks, the wheelchair was positioned at

one end of the room, facing directly towards the centre of the other end of the room. For the first task, the wheelchair was instructed to travel 4m ahead in the x-direction. In task 1 (n=5), the wheelchair was unobstructed by any obstacles. For task 2 and 3 (n=5), the wheelchair was made to move 5m ahead in the x-direction. In task 2, a static obstacle of height 2.5m and width 0.5m was placed 2.5m ahead of the wheelchair; and in task 3, the static obstacle remained in the same position and another person, acting as a dynamic obstacle, of height 2m and width 0.5m was instructed to walk by the wheelchair and stand near the static obstacle. It was found that planner frequency of 5Hz [out of 20, 10, 5 Hz], position tolerance of 0.13m (13 cm) [out of 100, 50, 25, 13, 6 cm] and orientation tolerance of 0.06 radians (3.4 degrees) [out of 1, 0.5, 0.25, 0.125, 0.0612, 0.0306 radians] gave the best performance as seen from Fig. 5-6. As these parameter values are corresponding to the lowest travel times. The optimisation experiments also demonstrate that achieving wheelchair orientation with a tolerance of 3.4 degrees at low travel time is possible.

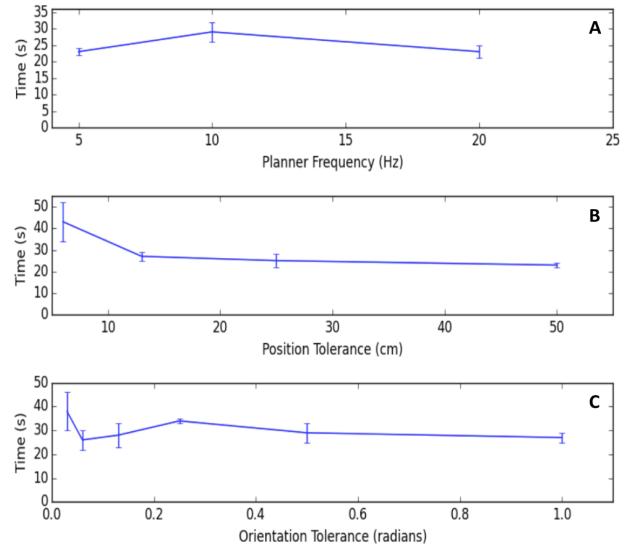


Figure 5. Task: 1. (A) shows the time taken to travel 4m varied as a function of the planner frequency. (B) shows the time taken to travel 4m varied as a function of the goal (x, y) position tolerance. (C) shows the time taken to travel 4m varied as a function of orientation tolerance.

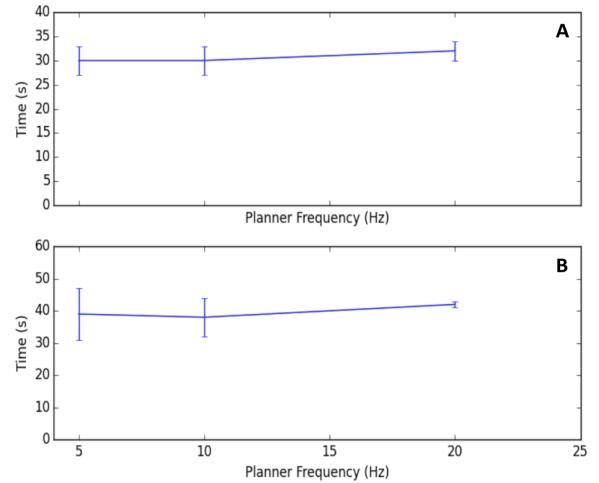


Figure 6. Performance results for Task 2 and Task 3. (A) shows the time taken to travel 5m around a static obstacle varied as a function of the planner frequency. (B) depicts the time taken to travel 5m around both a

static obstacle and a dynamic obstacle varied as a function of the planner frequency.

Based on these results, the parameters for the system architecture were optimised, and task 4 (n=6) was performed. In task 4, the goal for the autonomous wheelchair was defined as to move from A to B (5m), within different dynamic environment scenarios, i.e. the number of static and dynamic obstacles and their position were changed for each run to simulate different routes but within the same room for the same start and end goal coordinates. Time taken to move from A to B varied for different routes. However, the autonomous wheelchair was able to detect both static and dynamic obstacles with perfect accuracy using optimised parameters.

Next, a questionnaire about the comfort level and required additional features was provided to 3 volunteers (2 Quadriplegic and 1 paraplegic) who showed interest in evaluating our system further to a demonstration. State of the passenger during autonomous navigation was recorded. Based on the observations, the system architecture was improved [13].

D. 3D Gaze Based Semi-Autonomous Wheelchair Evaluation

The wheelchair participant's (subject) intention to get to the Object of Interest was decoded successfully. From the accuracy level comparisons, we see that Naive gaze-pointer approach with fine Gaussian SVM results in 78.9% accuracy, and that result was just for, i.e. "Take me to the Door". The SVM was built from the 2D gaze point positions, meaning that we can explain how the system works to our users. Such an explanation of machine learning results for end-users is highly essential from the perspective of usability, accessibility and trust with the system. From Fig. 2., we can observe that there is a clear separation between the positions of gaze point for different tasks. The obstacle detection and collision avoidance were improved in the autonomous wheelchair platform by deploying a deep RL controller. The 3D gaze-based modules were then integrated with the autonomous wheelchair platform to test the semi-autonomous functionality and cognitive ability. In addition to the intention decoding module, 3D gaze-based end-point control used to decode user's navigational intentions was investigated. A 'Wink' lasting for 3 seconds was used to define a destination within depth camera FOV. 3 'Winks' with the right eye was used for turning right, and 3 'Winks' with the left eye was used for turning left. Thus, continuous gaze-based semi-autonomy was achieved successfully. For evaluation, the wheelchair was positioned between static and dynamic obstacles and the task was carried out six times (n=6). When a low-level navigation intention was successfully detected, the wheelchair was able to navigate to the destination accounting for the obstacles along its path.

IV. CONCLUSION

Our fully functioning cognitive wheelchair can be built using frugal sensors and state-of-the-art modules based on Machine Learning algorithms. Natural gaze-based intention decoder and 3D gaze-based destination estimation method was developed and showed practical accuracy levels. We place the user in the centre of the AI loop, allowing the user to provide minimum input for successful navigation. Such cognitive, personal mobility technology can benefit severely

disabled to gain more independence and mobility in daily life activities. Our natural user interface may lead to better adoption by patients over time as we have approached its development from an embodied perspective [18] by relying on natural interactions to drive human-robot interaction.

REFERENCES

- [1] H. Grewal, A. Matthews, R. Tea, and K. George, "Lidar-based autonomous wheelchair," in IEEE Sensors Applications Symposium (SAS), pp. 1–6, 2017.
- [2] P. Marin-Plaza, A. Hussein, D. Martin, and A. de la Escalera, "Global and local path planning study in a ros-based research platform for autonomous vehicles," Journal of Advanced Transportation, 2018.
- [3] D. Schwesinger, A. Shariati, C. Montella, and J. Spletzer, "A smart wheelchair ecosystem for autonomous navigation in urban environments," Autonomous Robots, vol. 41, no. 3, pp. 519–538, 2017.
- [4] S. I. Ktena, W. Abbott and A. A. Faisal, "A virtual reality platform for safe evaluation and training of natural gaze-based wheelchair driving," in Proc. 7th IEEE/EMBS Conf. Neural Engineering (NER), pp. 236–239, 2015.
- [5] L.A. Raymond, M. Piccini, M. Subramanian, P. Orlov, G. Zito, and A. A. Faisal, "Natural Gaze Data Driven Wheelchair," bioRxiv, p.252684, 2018.
- [6] A. M. Brouwer, V.H. Franz, and K.R. Gegenfurtner, "Differences in fixations between grasping and viewing objects," Journal of Vision, vol. 9, no. 1, pp.18, 2009.
- [7] A. H. Fathaliyan, X. Wang, and V.J. Santos, "Exploiting Three-Dimensional Gaze Tracking for Action Recognition During Bimanual Manipulation to Enhance Human–Robot Collaboration," Frontiers in Robotics and AI, vol. 5, pp. 25, 2018.
- [8] S. Dziemian, W. W. Abbott, and A. A. Faisal, "Gaze- based teleprosthetic enables intuitive continuous control of complex robot arm use: writing & drawing," In Proceedings of the 6th IEEE RAS/EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob). IEEE, pp. 5, 2016
- [9] P. M. Tostado, W. W. Abbott, A. A. Faisal, "3D gaze cursor: Continuous calibration and end-point grasp control of robotic actuators," In Robotics and Automation (ICRA), IEEE International Conference, pp. 3295-3300, 2016.
- [10] P. Orlov, and Gorshkova K, "Gaze-Based Interactive Comics," In Proceedings of the 9th Nordic Conference on Human-Computer Interaction, ACM, pp. 116, 2016.
- [11] C. Auepanwiriyakul, A. Harston, P. Orlov, A. Shafti, and A.A. Faisal, "Semantic fovea: real-time annotation of ego-centric videos with gaze context," In Proceedings of the ACM Symposium on Eye Tracking Research & Applications, p. 87, 2018.
- [12] A. Shafti, P. Orlov, and A. A., Faisal, "Gaze-based, context-aware robotic system for assisted reaching and grasping," In Robotics and Automation (ICRA), IEEE International Conference, pp. 863-869, 2019.
- [13] M. Subramanian, S. Park, P. Orlov, A. Shafti, A. A. Faisal, "Gaze-contingent decoding of human navigation intention on an autonomous wheelchair platform," [Under review].
- [14] W. W. Abbott, A. A. Faisal, "Ultra-low-cost 3D gaze estimation: an intuitive high information throughput compliment to direct brain-machine interfaces," Journal of neural engineering, vol. 9, pp. 046016, 2012.
- [15] M. Subramanian, N. Songur, D. Adjei, P. Orlov, and A.A. Faisal, "A.Eye Drive: Gaze based semi-autonomous wheelchair interface," in Engineering in Medicine and Biology (EMBC), IEEE International Conference, 2019. In Press.
- [16] W. Hess, D. Kohler, H. Rapp, and D. Andor, "Real-time loop closure in 2D LIDAR SLAM," in Robotics and Automation (ICRA), IEEE International Conference. pp. 1271-1278, 2016.
- [17] M. Everett, F. C. Yu, and P. H. Jonathan, "Motion planning among dynamic, decision-making agents with deep reinforcement learning." In 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 3052-3059. IEEE, 2018.
- [18] T. R. Makin, F. de Vignemont, A. A. Faisal, "Neurocognitive barriers to the embodiment of technology," Nature Biomedical Engineering vol. 1, pp. 0014, 2017.

A Frontal Cortical Loop For Autonomous Vehicles Using Neuralized Perception-Action Hierarchies

David Windridge and Seyed Ali Ghorashi, Middlesex University, London, UK *Member, IEEE*

Abstract— By modelling driving as a Perception-Action (PA) hierarchy it is possible to combine high-level symbolic logical reasoning (in particular, the Highway Code applied to hypothetical road configurations) with low-level sub-symbolic processes (specifically, Optimal Control and stochastic machine learning). In this context, we propose a cortical frontal loop analogue for autonomous vehicles in which progressively abstracted bottom-up scene understanding is followed by top-down legal action specification (with progressive contextual grounding), such that final action selection is carried out via simulated basal ganglia model. Although the top level of the PA-hierarchy employs explicit first-order logical reasoning we can exploit the duality principle of Hölldobler to generate a functionally equivalent deep neural network such that the PA hierarchy can learn adaptively at all levels.

I. INTRODUCTION

Perception-Action (PA) learning proposes an intrinsic link between the perceptive and active capabilities of an agent (motto: *action precedes perception*). This may be modelled as an explicit bijection constraint between percept transitions and actions: $P \times P \rightarrow A$ such that any perceptual redundancy is eliminated in relation to the agent's affordances with respect to the environment.

The notion of a *Perception-Action Hierarchy* further relates the Brooksian notion of action subsumption to this progressive perceptual abstraction via layer-wise application of the PA bijection principle. We can thus model human car-driving as a PA hierarchy by enabling the combination of high-level symbolic logical reasoning (i.e. in relation to the Highway code) with low-level sub-symbolic processes.

To this end, we here outline a run-time Cortical Frontal Loop analogue in which (progressively abstracted) bottom-up scene representation is followed by top-down (legal) action specification. The top level of the PA-hierarchy, the *Logical Reasoning Module (LRM)*, hence employs explicit first-order logical reasoning in order to compute the full set of equi-legal agent actions (constituting the Herbrand base of the LRM's logical programme) with respect to the currently configuration as interpreted via the bottom-up scene understanding.

The PA-hierarchy so constructed utilizes both neural and formal reason processes. However, there is a fundamental duality principle that suggests Logic Programmes are always capable of neuralization (cf. Hölldobler & Kalinke's equivalence between 3-layer NNs and logic programmes (LPs)). However, if rule-base is hierarchical (as it must be in a PA-hierarchy), then the above equivalence becomes that between the reasoning hierarchy and a functionally equivalent deep neural network. This means that the system can, in principle, be so constructed as to be able to learn adaptively (via

back-propagation) at all levels via an end-to-end neuralization of the PA hierarchy.

II. A HIERARCHICAL SENSORIMOTOR CONTROL SYSTEM FOR AUTONOMOUS VEHICLES

The driving agent model here proposed implements a biologically-analogous cortical system for *hierarchical sensorimotor control*, with physically connected (non-symbolic) bottommost layers and a top-most symbolic subsumption architecture. Long term (strategic) goals, in particular compliance with the highway code is enacted by the symbolic module. The module acts on the sub-symbolic (physical) layer by specifying desirable target areas, hence biasing low-level action selection. The symbolic module thus steers the behavior of the lowermost (physically-connected) layer which retains the final authority vetoing all but the safe tactical maneuvers that are moment-by-moment available (it can veto incorrect high-level requests).

Our architecture is inspired by the organization of the human brain's visual processing [1-3], with differing cortical loops permitting different agent learning modalities:

The *cerebellar loop* learns forward/inverse models of the vehicle/environment dynamics (used for motor control and adaptation to differing environments, as well as embodied simulation for training the dorsal stream to learn the value of novel short-term tactical-level maneuvers).

The *dorsal stream* has a convergence-divergence organization and learns compact representations of simple events that are used to construct simple episodes for developing short-term motor strategies (e.g., imagining other road users' possible behaviors and learning collision avoidance countermeasures).

The *symbolic level* learns long-term strategies with high-level action selection via reinforcement learning in an episodic simulation context.

Agent evolution is conducted via off-line learning utilizing wake-dream cycles to replace the various neural network building blocks.

The role of the dorsal stream is hence to recognize actions latent in the environment and to prepare motor plans accordingly [1]. The dorsal stream also has a role in conceptualizing episodes. Both of these capabilities are naturally implemented within our system via an auto-encoded convergence-divergence architecture.

Because of its intrinsically discrete and iterative nature, however, neuralization of the symbolic frontal cortex is less

straightforward. We thus first give a detailed account of this loop prior to discussing our symbolic neuralization strategy.

III. THE BIASING LOOP (FRONTAL CORTEX LOOP)

To implement complex symbolic rule-based behaviors such as legal action sequence-planning (e.g. overtaking), further layers are constructed on top of the dorsal stream that steer the agent low-level behaviors so as to produce legal action sequences for longer-term goals.

This is specified as a hierarchical PA-subsumption architecture that provides a unified framework for semantic annotated event logging, generation of legal priors for action selection via the basal ganglia (BG) loop and high-level motor babbling/top-down dream instantiation (in the offline system).

There is hence a unified architecture within the LRM incorporating a common symbolic/sub-symbolic interface that operates across 3 distinct symbolic/sub-symbolic information-flow modalities: *bottom-up semantic annotation*, *top-down legal intention biasing*, *top-down dream instantiation*.

While the sub-symbolic system operates via goal salience (i.e. defined regions of the motor cortex), which may be learned so as to optimize long-term strategical behaviors, the LRM can only make recommendations (as per the subsumptive ‘principle of lower-level veto’), such that the final choice is in charge of the lowest level motor control (the dorsal stream). The final authority is thus always in the responsibility of the dorsal stream physical loops.

A. The Logical Reasoning Module

The subsumptive Perception Action hierarchy embodied within the LRM consequently implements the symbolic (i.e. high-level representational) component of the architecture, being responsible for high-level scene interpretation & annotation, and for introducing legal biasing in intention (note that the highway code itself does not generally identify unique actions within a given road context, but rather gives rise to a degenerate, *equi-legal* set of action possibilities).

The LRM acts via a mixture of theorem proving-via-resolution and functional extrapolation in order to apply the HWC in unfamiliar scenarios, with the former constituting the highest level of PA subsumption. The road configuration is thus represented within the LRM as instantiated logical variables, irrespective of the LRM’s operational modality (as indicated, the LRM subsumption framework is constrained to have the capability to act reversibly, that is to say, in a *generative manner* via reverse PA logical-variable instantiation, such that hallucinated high-level legal road configurations are spontaneously generated alongside the corresponding legal intentionality in the offline dreaming process. The latter (although beyond the scope of this paper) is an instance of top-down exploratory PA motor babbling, in which theorem proving-via-resolution is applied to random instantiations of logical variables in order to establish self-consistent Herbrand (i.e. logically-self consistent) interpretations, i.e. scenarios consistent with the legal road protocols).

Thus, while the offline dreaming process is one of top-down symbolic grounding through the full PA subsumption architecture, it is conversely the case that run-time high-level

scene-description and annotation may be seen as a process of bottom-up symbolic abstraction. The two processes are hence the precise inverse of each other in the LRM’s design.

B. PA Subsumption Design Principle Adopted by the LRM

The criteria for the number of levels in the hierarchy is defined by the notions of subsumption and Percept-Action bijection. Application of the PA bijectivity criterion implies that we should, as far as possible, represent only those percepts that distinguish intentional actions on a given layer. This means that each intention must bring about a perceivable change in such a way that the total set of percepts is minimized with respect to the available actions (affordances), consistent with the highway code representation of *a priori* meaningful perceptual objects. In practical terms, application of this principle means that, for example, it is not possible to have two consecutive legal gaps within a lane, since a ‘legal gap’ in order to exist as a high level percept must be distinguishable by a correspondingly legally-definable intention (a legal gap is defined as a potential legal place of relative occupation for the Ego car within a given lane, and as such is not sub-divisible at the highest level of legal intentionality).

The notion of Subsumption in the LRM is thus related to the legal sub-structuring of high-level intentionality; in particular, where perceptual targets are fine-grained by sub-intentions, for which the same PA bijectivity condition also applies.

This bijectivity principle also extends to levels below that indicated by the HWC; however, the lowest intentional level defined by the HWC is that of the linearized road metric; this therefore dictates the interface point of the LRM with the rest of the system (equally, this is the symbolic/sub-symbolic cut off) as indicated in fig. 1.

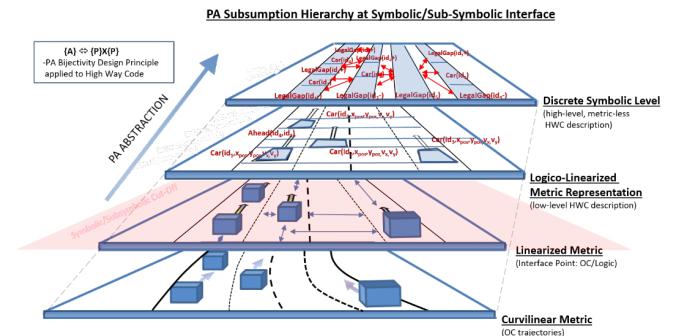


Figure 1. Run-time system PA hierarchy (OC refers to the optimal control trajectories existing at the physical layer)

From the hierarchical PA perspective, there are thus two distinct symbolic reasoning layers implicit in the Highway Code (because the HWC explicitly excludes both navigational considerations and motor processes from its remit, which would respectively extend the higher and lower levels of the hierarchy if present). The two levels are: the *discrete symbolic level* and the *logico-linear metric level*, as shown in Figure 1.

Consequently, legal-intention related configurations can only be defined in the above terms; they collectively represent the high-level semantic annotation (or equivalently, the high-level scene understanding) brought about by hierarchical PA considerations.

The LRM is therefore architected on two distinct layers (see Figure), with a perception/action interface specified between each level at the appropriate level of symbolic abstraction.

C. Interlayer Interface Structure of the LRM

The Highway Code refers to both discrete symbolic entities (cars, lanes, signs, gaps, etc.), as well as linearized-metric entities —i.e. metric entities expressed in terms of distance-to/time-to and distance-from/time-from other entities described in relation to the Ego Car.

At the high-level node, lane-wise road configurations are characterized in the LRM via a *logical-list* format: ordered in-lane lists of cars and gaps, with (the equivalent of) predicated assertions as to which cars/gaps are legally adjacent to which others.

At the immediately lower level of the LRM, the (symbolic/sub-symbolic) node is characterized by *annotated metrical bounding boxes* relating to legal transitions produced by a two-stage process, corresponding to the two stages of subsumption at the apex of the PA hierarchy listed above (the annotation aspect of the metric bounding boxes thus correlates to their high-level representation, illustrating the progressively grounded nature of symbols generated in a PA hierarchy).

Contextual metric information (distances to, and velocities of, other cars), received from the agent are hence converted into a non-metrical list of cars and gaps by means of linear extrapolation according the HWC protocols (i.e. assuming constant speeds and legally-specified reaction times). This list is passed to the second level of the LRM as the equivalent of declaratively-enacted predicate script, from which a set of high-level legal intentions with uniform priors are generated (they are uniform since road protocols do not distinguish between legal intentional possibilities *a priori*).

D. Bottom-up Communication from the Pre-LRM Layer

The bottom-up semantic annotation function of the runtime system thus involves communication through the various levels of the LRM in the form of abstractions of the perceptual data consistent with the outlined notion of perceptual subsumption:

At the symbolic/sub-symbolic interface layer (Linearized Metric Layer), geometric details such as the exact shape of the lanes are hence discarded, while the topology and linear distance (constituting a higher-level legal-symbolic parametrization) is retained. The speed and distance of individual objects in relation to the Ego car, and road configuration information in the form of lane numbering, width, lane marker types (e.g. whether lane change is allowed) etc. are passed to the LRM.

The net result of the bottom-up communication of road configuration, after processing by the logical-reasoning system, is thus a high-level symbolic representation of both the legal status of, and the legal possibilities with respect to, the current road configuration. This hence constitutes a semantic annotation of the road situation described with respect to a (legal) intentional frame, or equivalently the high-level scene interpretation.

E. Top-down communication from the LRM (Legal intention Grounding)

The logic-symbolic reasoning process, as well as providing the high-level interpretation of the road circumstances indicated above, also serves to provide a full set of Herbrand (i.e. logically-self consistent) interpretations of the future legal action possibilities (for example, whether it is legal to change lane in the current context).

These Herbrand sets are then grounded —i.e. propagated downwards (as instantiated hierarchical variables in the run-time mode)— through the perception-action subsumption hierarchy so that, at the point of interface they manifest as a set of *binary saliency indicators* attached to legally-designated areas in the linearized metric space.

The top-down communication from the LRM thus take the form of metrical bounding boxes augmented by discrete legal saliency indicators that are used to directly compute the motor cortex *biasing matrix*. This annotation thus simultaneously satisfies the requirements of perception-action bijectivity and legal self-consistency; in particular bijectivity allows the bounding box annotation to be directly interpretable at the motor cortex in regard to *action selection*.

Note that the top-down LRM logical annotation process is exhaustive, with a complete Herbrand-interpretation of the scene generated as the annotation output (this is a natural consequence of the logic program being applied recursively until an inferential fixed-point is arrived at).

This means that, in the event of incomplete input data, the system generates a full range of self-consistent ‘completion’ sets, which are effectively the equivalent of equally-weighted ‘possible worlds’ (in the modal logic sense) consistent with the input, composed of alternative groundings of predicate variables with the available constants.

F. Dreaming Initiation via Top-down Communication of Legal-Perceptual Priors (LRM Percept-Motor Babbling)

As a corollary, where no input is given to the LRM, there are no grounded logical road configuration variables asserted at the symbolic/sub-symbolic interface. In principle, this allows the LRM to initiate an offline learning via a dreaming process (i.e. high-level percept-action babbling) without any modification of the system’s subsumptive structure; exactly the same mechanism for legal biasing can be utilized for dreaming, since the Herbrand fixed-points in the absence of any assertion as to road configuration (i.e. no assertions relating to either road topology or to vehicle traffic using that topology) are simply a uniform set of possible worlds consistent with the legal constraints on the road configurations in general (the LRM’s logical axioms necessarily have only a nominal distinction between intentional-rules and environmental-consistency rules).

G. Action selection loops

Action selection within the system has is consequently organized in a hierarchical fashion. There are two distinct

action selection modules, acting at the symbolic (the LRM) and sub-symbolic (physical) levels of description. The levels are differentiated firstly by their differing inputs, and secondly by the differing timescales over which their decisions are made.

The higher-level action selection loop takes the outputs of the logical reasoning module (LRM) as its inputs. The high-level action selection module first assigns, at each time step, scalar weights representing the “desirability” of each of the LRM’s bounding boxes. These weights are learned, this way enabling the agent to learn long-term strategies.

Once the high-level action-selection loop has concluded its decision-making process, the conclusion can be passed down to the lower levels.

Neurally-inspired action selection within the agent hence takes the form of a computational model of the *basal ganglia*. In particular, it has been demonstrate that the basal ganglia could be performing a form of action selection known as ***multi-hypothesis sequential probability ratio testing (MSPRT)***. This algorithm sums evidence for each action over time, and finds the log likelihood that each channel is drawn from a distribution with a higher mean than the other channels. Once the log likelihood crosses a threshold, the action becomes selected. The threshold has to be tuned such that some predetermined error rate is permitted. Subject to a few assumptions, the algorithm can be shown to be optimal in decision time, given a particular error rate.

The MSPRT process is readily neuralizable, and so amenable to back-propagative learning in relation to the system as a whole. In order to produce fully end-to-end neuralization of the PA hierarchy it thus only remains to neuralize the LRM.

III. NEURALIZATION OF THE LRM

The direct translation of logic programs into artificial neural networks has a relatively long history. A standard approach to neuralisation of Horn clauses, using a local representation in which each (ground) atom corresponds to a single dedicated neuron, is exemplified by the Knowledge-Based Artificial Neural Network (KBANN) of Towell & Shavlik [7].

Networks of this type have been criticized as having a “propositional fixation”: a finite neural network can represent only a finite number of ground atoms, and can therefore represent a logic program only for a finite base. A language with first-order syntax but only a finite alphabet of symbols is equivalent to propositional logic, because any universally quantified (“for all X ”) clause can be translated into finitely many propositional clauses of the same form, one for each possible value of X . Thus, networks of the KBANN type cannot implement “true” first-order logic programs, only a finite fragment of first-order logic.

Holldobler et al. [8] give a neural method (in fact a precise duality) that replicates the immediate-consequence operator $\mathbf{T} \mathbf{P}$ of a true First-Order Logic program, to a desired degree of accuracy in a real embedding. However, this may involve a thousand or a million copies of a clause, one for each possible grounding of a variable X .

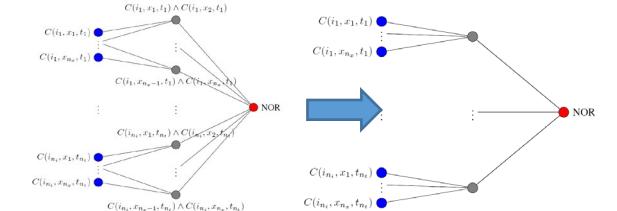
In neuralizing a logic program it is thus desirable to maintain the concept that there is a universal rule, rather than a thousand

unconnected rules, each of which might be only very weakly evidentially supported on its own and involves a combinatorial explosion in the number of neurons.

We thus construct a “neural logic programme parser” that simplifies the LRM Logical Programmes via a 3-fold strategy:

1. Appropriate thresholding considerations w.r.t. single predicate clauses potentially reduces the mid-layer neuronal budget by orders magnitude

=> it also naturally gives rise to a more pyramidal, CNN-like hierarchy



2. Assertion of facts can be accommodated straightforwardly to reduce input layer size.
3. Explicit observance of rule subsumption

Applying these three strategies very naturally results in a deep neural network structure, moreover one for which the layers intrinsically form a PA-hierarchy (since the underlying LRM logical rule base is constructed so as to respect PA bijectivity). Furthermore, there is *intrinsic parameter-sharing* amongst certain of the resulting network’s weights. Consequently, during training, all deep-learning tools appropriate for back-propagation within convolutional neural networks can be applied. Critically, FOL syntax is retained during training, irrespective of the network architecture of the sub-symbolic levels. We thus obtain an end-to-end trainable deep network for implementing a perception-action hierarchy based frontal cortex model within an autonomous driving context.

Acknowledgment: This work was supported by the EU 2020 project Dreams4Cars, grant number 731593

REFERENCES

- [1] P. Cisek, «Cortical mechanisms of action selection: the affordance competition hypothesis», Philos. Trans. R. Soc. B Biol. Sci., vol. 362, n. 1485, pagg. 1585–1599, set. 2007.
- [2] G. Pezzulo e P. Cisek, «Navigating the Affordance Landscape: Feedback Control as a Process Model of Behavior and Cognition», Trends Cogn. Sci., vol. 20, n. 6, pagg. 414–424, giu. 2016.
- [3] K. Meyer e A. Damasio, «Convergence and divergence in a neural architecture for recognition and memory», Trends Neurosci., vol. 32, n. 7, pagg. 376–382, lug. 2009.
- [4] D. Windridge, «Emergent Intentionality in Perception-Action Subsumption Hierarchies», Front. Robot. AI, vol. 4, ago. 2017.
- [5] D. Windridge e S. Thill, «Representational fluidity in embodied (artificial) cognition», Biosystems, vol. 172, pagg. 9–17, ott. 2018.
- [6] R. Bogacz e K. Gurney, «The basal ganglia and cortex implement optimal decision making between alternative actions», Neural Comput., vol. 19, n. 2, pagg. 442–477, 2007.
- [7] G. Towell and J. W. Shavlik, “Knowledge based artificial neural networks,” Artificial Intelligence, vol. 70, no. 4, 1994.
- [8] S. H’olldobler, F. Kurfess, and H.-P. St’orr, “Approximating the semantics of logic programs by recurrent neural networks,” Applied Intelligence, vol. 11, no. 1.

Following Social Groups: Socially Compliant Autonomous Navigation in Dense Crowds

Xinjie Yao¹, Ji Zhang² and Jean Oh²

Abstract—In densely populated environments, socially compliant navigation is critical for autonomous robots as driving close to people is unavoidable. This manner of social navigation is challenging given the constraints of human comfort and social rules. Traditional methods based on hand-craft cost functions to achieve this task have difficulties to operate in the complex real world. Other learning-based approaches fail to address the naturalness aspect from the perspective of collective formation behaviors. We present an autonomous navigation system capable of operating in dense crowds and utilizing information of social groups. The underlying system incorporates a deep neural network to track social groups and join the flow of a social group in facilitating the navigation. A collision avoidance layer in the system further ensures navigation safety. In experiments, our method generates socially compliant behaviors as state-of-the-art methods. More importantly, the system is capable of navigating safely in a densely populated area (10+ people in a 10 m × 20 m area) following crowd flows to reach the goal.

I. INTRODUCTION

The ability to safely navigate in populated scenes, e.g. airports, shopping malls, and social events, is essential for autonomous robots. The difficulty comes from the fact that people walk closely to the robot cutting ways in front of the robot or between the robot and the goal point. The safety margin for the robot to drive in crowded scenes is pushed to the minimum. In such a case, the navigation system has to trade-off between driving safely close to people and reaching the goal quickly. Furthermore, a previous study of socially compliant navigation [1] states three aspects in terms of the robot behaviors – *comfort* as the absence of annoyance and stress for humans in interaction with robots, *naturalness* as the similarity between the robot and human behaviors, and *sociability* as to abide by general cultural conventions. Among these three aspects, the first aspect essentially reflects safety of the navigation.

Previous studies on socially compliant navigation attempt to solve the problem with various methods, including data-driven approaches for human trajectory prediction [2], [3], potential field-based [4] and social force model-based [5] approaches. In particular, reinforcement learning-based methods use reward functions to penalizes improper robot behaviors eliminating the cause of discomfort [6], [7]. Inverse reinforcement learning based-methods learn from expert demonstrations [8]. These methods are hard to generalize due

¹X. Yao is with the Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology. Email: xyaoab@ust.hk

²J. Zhang and J. Oh are with the Robotics Institute, Carnegie Mellon University. Emails: zhangji@cmu.edu, jeanoh@cmu.edu

to that a large set of comprehensive expert demonstrations are hard to acquire.

The study of this paper is based on our previous work which uses deep learning in solving the socially compliant navigation problem [9]. This paper extends the work in two ways. First, we consider the findings from a previous study [10] that 70% of people walk in social groups. Crowd behavior can be summarized as flows of social groups, and humans tend to move along the flow. It is our understanding that the behavior of joining the flow that shares similar heading direction is more socially compliant, causing fewer collisions and disturbances to surrounding pedestrians. Our method recognizes social groups and selects the flow to follow. Second, we ensure safety with a multi-layer navigation system. In this system, a deep learning-based global planning layer makes high-level socially compliant behavioral decisions while a geometry-based local planning layer handles collision avoidance at a low-level.

The paper is further related to previous work on modeling aggregate interactions among social groups [10] and leveraging learned social relations in tracking group formations [11]. Our main contributions are a deep learning-based method for socially compliant navigation with an emphasis on tracking and joining the crowd flow and an overall system integrated with the deep leaning method capable of safe autonomous navigation in dense crowds.

II. METHOD

A. System Overview

Fig. 1 gives an overview of the autonomous navigation system which consists of three subsystems as follows.

- **State Estimation Subsystem** involves a multi-layer data processing pipeline which leverages lidar, vision, and inertial sensing [12]. The subsystem computes the 6-DOF pose of the vehicle as well as registers laser scan data with the computed pose.
- **Local Planning Subsystem** is a low-level planning subsystem in charge of obstacle avoidance in the vicinity of

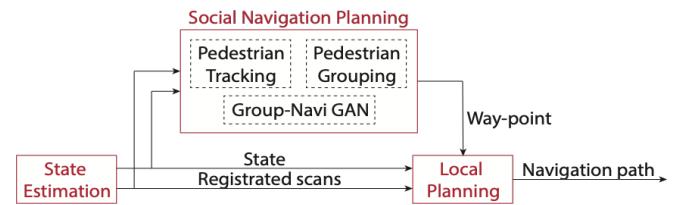


Fig. 1: Navigation software system diagram.

the vehicle. The planning algorithm involves a trajectory library and computes collision-free paths for the vehicle to navigate [13].

- **Social Navigation Planning Subsystem** takes in observations only consisting of pedestrians by subtracting the prior map. The subsystem tracks pedestrians in the surroundings of the vehicle, and then extracts the grouping information from the pedestrian walking patterns, with which, the subsystem generates way-points (as input of the Local Planning Subsystem), leveraging Group-Navi GAN, a generative planning algorithm in an adversarial training framework based on a deep neural network, Navi-GAN [9].

B. Group-Navi GAN

Following the extended social force model [10], we propose Group-Navi GAN, a framework to jointly address the safety and naturalness aspects at a group's level. Group-Navi GAN is inspired by our previous work Navi-GAN [9] which models social forces at an individual's level. An intention-force generator in the Group-Navi GAN deep network models the driving force as \vec{f}_i^0 for target agent i to move toward the goal. A group-force generator models the repulsive force from other pedestrians j as \vec{f}_{ij} and the interaction force from other group members as \vec{f}_i^{group} . The joint output of the intention-force generator and group-force generator defines the path for the robot to navigate.

In the group-force generator, a group pooling module first associates the target agent to a group based on the motion information (see Fig. 2). Then, the group pooling module computes path adjustments which essentially guide the robot to follow the group. We apply a support vector machine classifier [14] trained by [11] to determine if two agents belong to the same group. This uses the local spatio-temporal relation to cluster the agents with similar motions based on the coherent motion indicators, i.e. the differences in walking speed, spatial locations, and headings.

We use the following equation to aggregate the hidden state from h_j^t to $h_j'^t$,

$$h_j'^t = I_{ij}[s_i = s_j] * \cos(\theta_i - \theta_j) * h_j^t, \quad (1)$$

where $I_{ij}[s_i = s_j]$ indicates if two agents are in the same group,

$$I_{ij}[s_i = s_j] = \begin{cases} 1, & \text{if } i \text{ and } j \text{ are in the same group} \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

θ_i and θ_j are the agent headings. The resulting embedding H_i^t of hidden state $h_j'^t$ is computed as a row vector which consists of the maximum elements from all other agents. The embedding is further concatenated for decoding,

$$H_i'^t = [H_i^t, h_i^t, n_i] \quad (3)$$

where n_i is random noise drawn from $\mathcal{N}(0, 1)$.

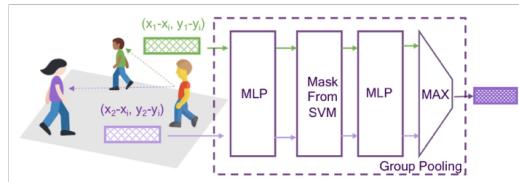


Fig. 2: Group pooling module in the Group-Navi GAN deep network. The input of the module is the relative displacements of the surrounding pedestrians w.r.t. the target agent. The module associates the target agent to a group based on the motion information and outputs path adjustments for the robot to follow the group.

III. EXPERIMENTS

A. Social Compliance Evaluation

We evaluate our method on two publicly available datasets: ETH [15] and UCY [16]. These datasets include rich social interactions in real-world scenarios. We follow the same evaluation methodology as the leave-one-out approach and the error metrics used in the prior work [3]:

- 1) *Average Displacement Error*: The average L2 distance between predicted way-points and ground-truth trajectories over the predicted time steps.
- 2) *Final Displacement Error*: The L2 distance between the predicted way-point and true final position at the last predicted time step.

We compare against a linear regressor that only predicts straight paths, Social-GAN(SGAN) [3], and Navi-GAN [9]. We use the past eight time steps to predict the future eight time steps. As shown in TABLE I, our method yields considerable accuracy improvements for some of the datasets where rich group interactions are prevalent. In particular, UNIV and ZARA1 have more than 70% of the pedestrians moving in social groups, and thus our model performs better. Our model performs slightly worse than the state-of-the-art approaches with the ETH and HOTEL datasets due to the lack of social group interactions. Further, our method assumes the existence of a goal point for each person in the dataset. Lacking precise goal point information results in a relative low accuracy. In the next experiments, we will show results with author-collected data where the strength of our method is more obvious.

Metric	Dataset	Group Percentage	Linear	SGAN [3]	Navi-GAN [9]	Group-Navi GAN
ADE	ETH [15]	18%	0.84	0.60	0.95	1.33
	HOTEL [15]	19%	0.35	0.48	0.43	0.39
	UNIV [16]	73%	0.56	0.36	0.85	0.29
	ZARA1 [16]	70%	0.41	0.21	0.40	0.21
	ZARA2 [16]	69%	0.53	0.27	0.47	0.30
AVG		50%	0.54	0.39	0.62	0.50
FDE	ETH [15]	18%	1.60	1.22	1.64	1.98
	HOTEL [15]	19%	0.60	0.95	0.74	0.93
	UNIV [16]	73%	1.01	0.75	1.36	0.68
	ZARA1 [16]	70%	0.74	0.42	0.66	0.40
	ZARA2 [16]	69%	0.95	0.54	0.72	0.85
AVG		50%	0.98	0.78	1.02	0.96

TABLE I: Social compliance evaluation of Group-Navi GAN and other baseline approaches. Two error metrics, Average Displacement Error and Final Displacement Error are reported (in meters) for $t_{obs} = 8$ and $t_{pred} = 8$. We manually count the number of pedestrians moving in social groups. Our method outperforms the prior work with the UNIV and ZARA1 datasets where social groups are richly available.

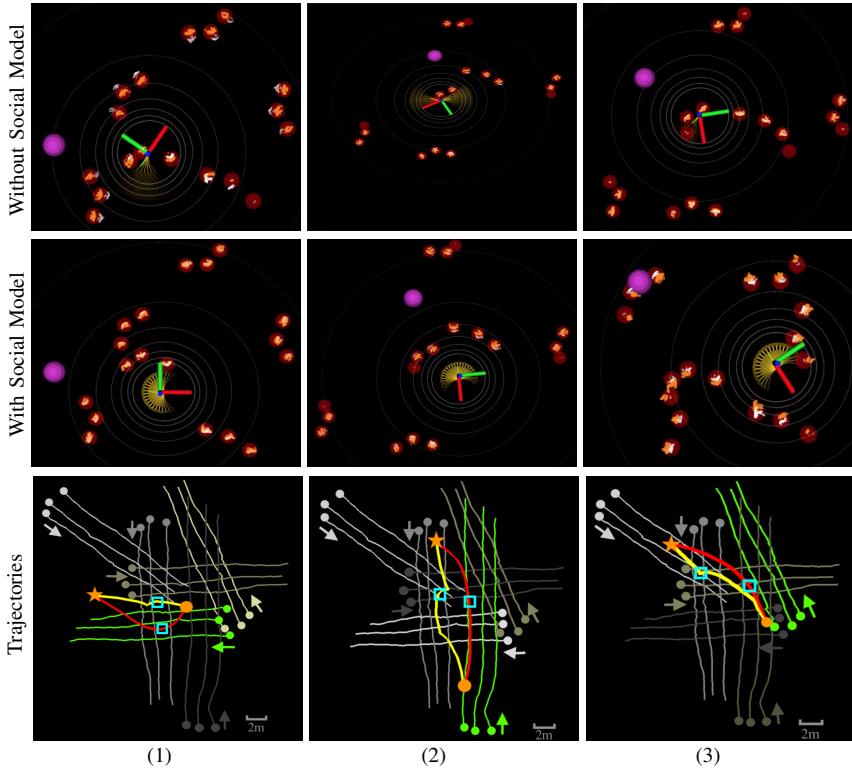


Fig. 3: Simulation results in a $10\text{ m} \times 20\text{ m}$ area. The tests involve 18 people walking in 6 groups. Each group moves in a different direction. The three columns present three representative cases. The first and second rows show screenshots of the simulation environment. The coordinate frame indicates the robot. The goal point is marked as the magenta dot. The red dots are the tracked pedestrians using laser scan data. The third row displays the trajectories of the pedestrians (gray and green) and the robot (yellow and red). The dots are the start points and the star is the goal point of the robot. When using Navigation without Social Model, the robot produces the yellow path. When using Navigation with Social Model, the robot follows the group in green color and produces the red path. A blue square is labeled on each robot path where the corresponding screenshot is captured on the first and second rows. Specifically, on the first row, the screenshots show the moments when the robot drives overly close to people due to not using the social model. On the second row, the screenshots are taken while the robot follows a group during the navigation.

B. Group Following Evaluation

We further evaluate the method with a robot vehicle as shown in Fig. 4. The robot is equipped with a Velodyne Puck laser scanner for collision avoidance and pedestrian tracking. Our method is evaluated in two configurations – Navigation with Social Model refers to the full navigation system as shown in Fig. 1, and Navigation without Social Model has the Social Navigation Planning Subsystem removed. The State Estimation Subsystem and the Local Planning Subsystem are directly coupled. The robot navigates directly toward the goal and uses the Local Planning Subsystem to avoid collisions locally.

We show results in both simulation and real-work experiments with pedestrian data collected by the robot. In simulation, we show scenarios with 18 people walking around the robot in 6 groups. In real-work experiments, we have 6 people walking in 2 groups. One group moves along the robot navigation direction and the other group moves in the opposite direction. The results are shown in Fig. 3 and Fig. 5. In each scenario, the robot selects a group to follow with the full navigation system (Navigation with Social Model). If using Navigation without Social Model, the robot drives directly toward the goal and results in interactions with

groups moving in other directions.

Finally, we conduct an Amazon Mechanical Turk (AMT) study to further understand the safety and naturalness of the robot navigation. A total of 466 participants evaluate the simulation and real-world results. As shown in Table II, $> 90\%$ of the participants consider Navigation without Social Model to be unsafe (with collisions) while the ratio reduces to $< 40\%$ using Navigation with Social Model. With the



Fig. 4: Experiment platform. A wheelchair-based robot carries a sensor pack on the top. The sensor pack consists of a Velodyne Puck laser scanner, a camera, and a low-grade IMU. The scan data is used for collision avoidance and pedestrian tracking. A laptop computer carries out all onboard processing.

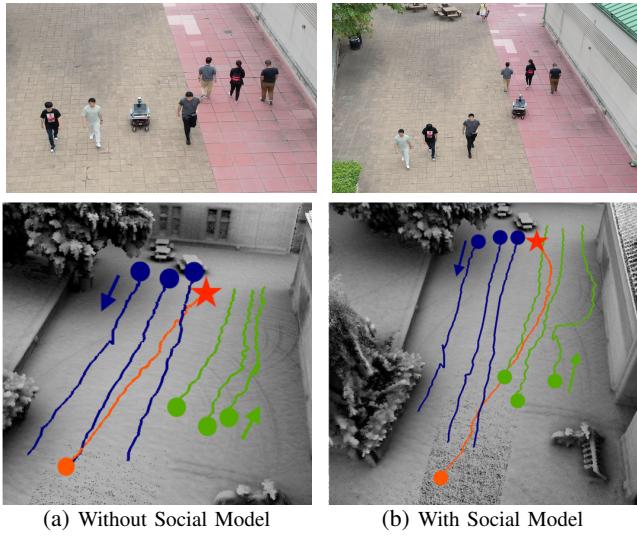


Fig. 5: Real-world experiments in a $10\text{ m} \times 20\text{ m}$ area. The first row shows photos of 6 people walking in 2 groups. One group moves along the robot navigation direction and the other group moves in the opposite direction. The second row shows the corresponding trajectories of the people (blue and green) and the robot (orange). Dots indicate the start points and the star indicates the goal point of the robot. In (a), when using Navigation without Social Model, the robot drives directly toward the goal point and results in cutting through the group on the left that moves against the robot. In (b), when using Navigation with Social Model, the robot follows the group on the right and avoids disturbances to the pedestrians.

real-world results, 95% of the participants report that the robot forces other pedestrians to change their paths if using Navigation without Social Model. When using Navigation with Social Model, the ratio reduces to 4%. The survey result validates that our method helps reduce disturbances to other pedestrians as well as improves safety of the navigation. A video of these results can be seen at www.youtube.com/watch?v=I_SkA9rmxYE.

IV. CONCLUSION

The paper proposes an autonomous navigation system capable of operating in dense crowds. In this system, a Social Navigation Planning Subsystem incorporating a deep neural network generates socially compliant behaviors. This

Metric	Scene	Without Social Model	With Social Model
Collision (Safety)	(1)	97%	42%
	(2)	92%	6%
	(3)	92%	36%
Path Change (Naturalness)	Avg	93%	28%
Path Change (Naturalness)	Real world	95%	4%

TABLE II: Results of survey study. A total of 466 participants evaluate the simulation results in Fig. 3 and the real-world results in Fig. 5. We can see that $> 90\%$ of the participants consider the Navigation without Social Model to have collisions. For Navigation with Social Model, the ratio reduces to $< 40\%$. Further, 95% of the participants report that the robot forces other pedestrians to change their paths if using Navigation without Social Model. When using Navigation with Social Model, the ratio reduces to 4%. The ratios reduce by 3 times in terms of collision and 20 times in terms of path change which validate that our method helps reduce disturbances to other pedestrians as well as improves safety.

involves a group pooling mechanism by inferring social relationships to encourage the autonomous navigation to join the flow of a social group sharing the same moving direction. We show the effectiveness of our method through quantitative and empirical studies in both simulations and real-world experiments. The result is that by joining the crowd flow, the robot has fewer collisions with people crossing sideways or walking toward the robot. Joining the flow also creates fewer disturbances to the pedestrians. As a result, the robot navigates in a safe and natural manner. Since this paper focuses on human-robot interactions at a group's level, extension of the work in the future can model interactions between groups and scattered individuals.

ACKNOWLEDGMENT

Special thanks are given to C.-E. Tsai, Y. Song, D. Zhao for facilitating experiments.

REFERENCES

- [1] T. Krusea, A. K. Pandeybc, R. Alamibc, and A. Kirschd, “Human-aware robot navigation: A survey,” *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1726–1743, Dec 2013.
- [2] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, F.-F. Li, and S. Savarese, “Social Istm: Human trajectory prediction in crowded spaces,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016.
- [3] G. Agrim, J. Justin, F.-F. Li, S. Silvio, and A. Alexandre, “Social gan: Socially acceptable trajectories with generative adversarial networks,” *arXiv preprint arXiv:1803.10892*, 2018.
- [4] F. Hoeller, D. Schulz, M. Moors, and E. F. Schneider, “Accompanying persons with a mobile robot using motion prediction and probabilistic roadmaps,” in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2007.
- [5] D. Helbing and P. Molnar, “Social force model for pedestrian dynamics,” *Physical Review E*, vol. 51, 1998.
- [6] Y. F. Chen, M. Everett, M. Liu, and J. P. How, “Socially aware motion planning with deep reinforcement learning,” *arXiv preprint arXiv:1703.08862*, 2018.
- [7] C. Chen, Y. Liu, S. Kreiss, and A. Alahi, “Crowd-robot interaction: Crowd-aware robot navigation with attention-based deep reinforcement learning,” *arXiv preprint arXiv:1809.08835*, 2019.
- [8] H. Kretzschmar, M. Spies, C. Sprunk, and W. Burgard, “Socially compliant mobile robot navigation via inverse reinforcement learning,” *The International Journal of Robotics Research*, vol. 35, no. 11, 2016.
- [9] C.-E. Tsai, “A generative approach for socially compliant navigation,” Master’s thesis, Pittsburgh, PA, June 2019.
- [10] M. Moussad, N. Perozo, S. Garnier, D. Helbing, and G. Theraulaz, “The walking behaviour of pedestrian social groups and its impact on crowd dynamics,” *arXiv preprint arXiv:1003.3894*, 2010.
- [11] T. Linder and K. O. Arras, “Multi-model hypothesis tracking of groups of people in rgbd data,” in *Proc. of the 17th Intl Conf. on Information Fusion*, 2014.
- [12] J. Zhang and S. Singh, “Random field topic model for semantic region analysis in crowded scenes from tracklets,” in *Journal of Field Robotics*, vol. 35, no. 8, 2018.
- [13] J. Zhang, C. Hu, R. Gupta Chadha, and S. Singh, “Maximum likelihood path planning for fast aerial maneuvers and collision avoidance,” in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2019.
- [14] C. Cortes and V. N. Vapnik, “Support-vector networks,” *Machine Learning*, no. 20, 1995.
- [15] S. Pellegrini, A. Ess, and L. V. Gool, “Improving data association by joint modeling of pedestrian trajectories and groupings,” in *Computer VisionECCV*, 2010.
- [16] L. Leal-Taixe, M. Fenzi, A. Kuznetsova, B. Rosenhahn, and S. Savarese, “Improving data association by joint modeling of pedestrian trajectories and groupings,” in *CVPR*, 2014.