Deep Trail-Following Robotic Guide Dog in Pedestrian Environments for People who are Blind and Visually Impaired - Learning from Virtual and Real Worlds

Tzu-Kuan Chuang*,1, Ni-Ching Lin*,1, Jih-Shi Chen[1], Chen-Hao Hung[1], Yi-Wei Huang[1], Chunchih Teng[1], Haikun Huang[2], Lap-Fai Yu[2], Laura Giarré[3], and Hsueh-Cheng Wang[1]

***Abstract*— Navigation in pedestrian environments is critical to enabling independent mobility for the blind and visually impaired (BVI) in their daily lives. White canes have been commonly used to obtain contact feedback for following walls, curbs, or man-made trails, whereas guide dogs can assist in avoiding physical contact with obstacles or other pedestrians. However, the infrastructures of tactile trails or guide dogs are expensive to maintain. Inspired by the autonomous lane following of self-driving cars, we wished to combine the capabilities of existing navigation solutions for BVI users. We proposed an autonomous, trail-following robotic guide dog that would be robust to variances of background textures, illuminations, and interclass trail variations. A deep convolutional neural network (CNN) is trained from both the virtual and real-world environments. Our work included major contributions: 1) conducting experiments to verify that the performance of our models trained in virtual worlds was comparable to that of models trained in the real world; 2) conducting user studies with 10 blind users to verify that the proposed robotic guide dog could effectively assist them in reliably following man-made trails.**

I. Introduction

We wish to enable independent navigation for people who are blind or visually impaired (BVI). We proposed a solution in the form of a robotic guide dog with physical human-robot interaction to extend the capability and reliability of a white cane. The robotic guide dog could travel indoors or outdoors on various terrains of general pedestrian environments and used a vision-based learning approach to autonomously follow man-made trails.

A. Navigation Aids for the Blind and Visually Impaired

There is a great need to develop navigation aids, given that there are 286 million BVI people in the world, according to the World Health Organization [1]. In the last two decades, assistive technologies have been developed, including those with functionalities of navigation, localization, and obstacle avoidance using several types of non-visual feedback, such as voice and vibration [2]. To date the most commonly used navigation aid for BVI people is a white cane, given its low-cost and reliability [3]. Recently, Wang et al. [4] developed a system that works along with a white cane. It uses an RGB-D camera, embedded computer, and haptic device to provide feedback to avoid contact with other pedestrians walking through a crowded environment or detect a target object such as an empty chair. We refer the readers to the reviews of Wang et al. [4] for other wearable solutions for safe navigation.

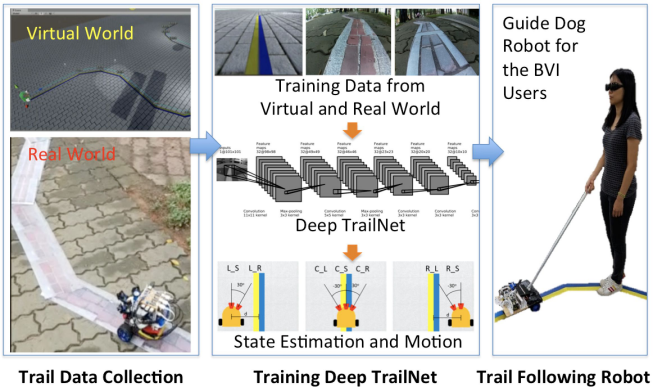


Fig. 1: We proposed a trail-following guide dog robot powered by a deep convolutional neural network (CNN). The network was trained in both the real and virtual worlds to estimate the robot states and generate robot motions. The robot guide dog was shown to assist 10 blind users complete trail-following tasks.

A guide dog is another navigation aid that assists BVI users navigate without contacting other pedestrians; they also provide companionship for the users. Nevertheless, according to the International Guide Dog Federation (IGDF) [5], guide dogs are still not available for most BVI users who need the service in many regions due to the high cost of service animal training and the pairing of dogs and users. Furthermore, some environments are not service animal friendly.

There have been numerous efforts to develop navigation robots that assist blind people. Different forms of robots are capable of executing distinct tasks in various environments. Kulyukin et al. [6] designed a wheeled robot using a laser range finder and radio-frequency identification (RFID) sensors to navigate around indoor environments with pre-installed RFID tags. Gharpure and Kulyukin [7] proposed a shopping robot for blind people, which guides the user in a store and informs them of the prices of commodities. However, it only operates indoors. For outdoor robots, Rasmussen et al. [8] used stereo camera with a tilting range finder as an input sensor. From the input data, the robot is capable of navigating along outdoor trails intended for hikers and bikers. Another form of robot was proposed by Ulrich and Borenstein [9]. Called "GuideCane", it is a robot combined with a white cane. When the GuideCane's ultrasonic sensors detect an obstacle, the

*T. Chuang and N. Lin contributed equally to this work. [1]Department of Electrical and Computer Engineering, National Chiao Tung University, Taiwan. Corresponding author email: hchengwang@g2.nctu.edu.tw

[2]Department of Computer Science, University of Massachusetts at Boston, USA

[3]University of Modena and Reggio Emilia, Modena, Italy

embedded computer determines a suitable direction of motion, which steers the GuideCane (and user) around it. Therefore, it does not require any training of the user. However, the weight of the robot might be too heavy for everyday use.

B. Autonomous Trail/Lane Following

One challenge for blind people is to follow a trail/lane in a specific environment. A variety of methods have been proposed for a range of scenarios. Rasmussen et al. [8], [10] used tilting LiDAR and omnidirectional camera data to navigate along outdoor trails intended for hikers and bikers. Using image processing techniques for appearance and structural cues, the system achieved good accuracy and robustness for trail-following over challenging scenarios with varying tread textures, border vegetation, illumination, and weather conditions. Siagian et al. [11] proposed a vision-based mobile robot navigation system that is capable of navigating along a road by detecting a set of lines extrapolated from the contour segments. The heading and lateral position of the robot are maintained by centering the vanishing point in its field of view. It was tested to work even in busy college campus environments, including challenges of occlusion by pedestrians and obstacles, non-standard road markings and shapes, and shadows. In a survey of lane/road detection by Hillel et al. [12], the relevant approaches and algorithmic techniques are categorized into functional building blocks, including image cleaning, feature extraction, model fitting, time integration, and image-to-world correspondence. [13]–[15] develop autonomously traverse within a lane using low-level features processing pipeline of line detection, ground projection, lane filtering, lane control, and finite state machine (FSM).

Recently, deep Convolutional Neural Networks (CNN) have been used to achieve autonomous trail or lane following. Giusti et al. [16] tackled autonomous forest or mountain trail-following using a single monocular camera mounted on a mobile robot, such as a micro-aerial vehicle. Unlike the previous literature, they focused on trail segmentation and used low-level features to develop a supervised learning approach using a deep CNN classifier. The trained CNN classifier was shown to follow unseen trails using a quadrotor. Deep driving [17] categorizes the autonomous driving work into three paradigms. *Behavior Reflex* is known as a low-level approach for constructing a direct mapping from the image/sensory inputs to produce a steering motion. This is done by means of a deep CNN trained by labels generated from human driving along a road or in virtual environments. *Mediated perception* is the recognition of driving-relevant objects, e.g., lanes, traffic signs, traffic lights, cars, or pedestrians. The recognition results are then combined into a consistent world representation of the cars and immediate surroundings. *Direct perception* falls between mediated perception and behavior reflex. It proposes to learn a mapping from an image to estimate several meaningful states of the road situation, such as the angle of the car relative to the road and the lateral distance to lane markings. With the state estimation, other filters or FSMs and controllers can be applied. The forest-trail-following vehicle in [16] belongs to behavior reflex, whereas the Duckietown falls into the direct perception paradigm.

Fig. 2: Man-made trails in pedestrian environments. From left to right: 1) a real-world indoor environment in a hospital in Taiwan; 2) guiding trail in a school for the blind in Italy; and 3) the Freedom Trail in Boston, USA.

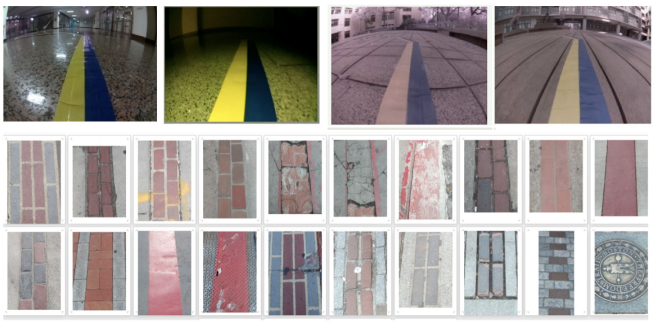


Fig. 3: Trail variations. Top: man-made yellow-blue trails applied to ground with different textures; Bottom: there are more than 20 trail textures on the Boston Freedom Trail, resulting in a challenging interclass variation problem.

C. Challenges

The current trail-following solution for BVI users is to use a white cane to detect and follow tactile trails on the ground. However, building and maintaining such tactile trail infrastructures is expensive and may cause inconvenience to other pedestrians or wheelchair users. Man-made trails (as shown in Fig. 3) that are detected by vision-based algorithms could provide low-cost as an alternative to white canes and tactile trails. Nevertheless, there are still challenges to overcome, as in the previous works. Camera observations may vary when man-made trails are deployed on various background textures under different illuminations or shadows. Interclass variations, such as the textures of the Freedom Trail, also make it a challenging problem for feature-based approaches or RANSAC to solve. Recent deep learning approaches have been successful in lane/trail-following, such as the scenarios in a highway autopilot or in an unstructured forest. But to our knowledge, pedestrian environments have not been well studied yet. Therefore, a robust learning-based approach that enables trail-following is needed. We will bypass the direct perception paradigm of detecting features or object in the scene, and carry out both behavior reflex (mapping an image to a robot action) and mediated perception (mapping an image to a robot state, such as heading and lateral distance to the trail).

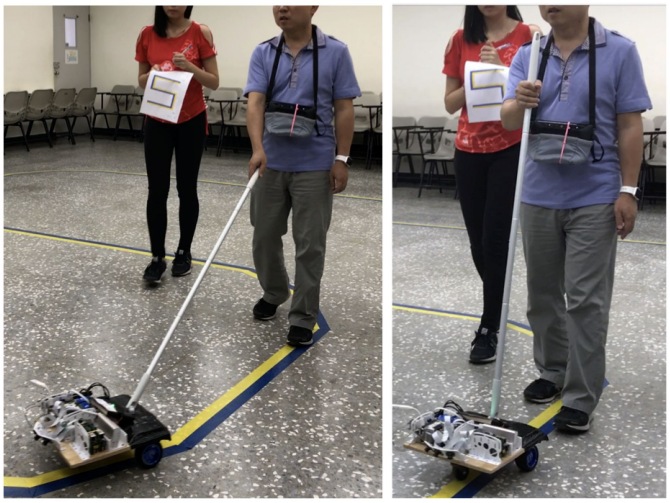


Fig. 4: The cane-like rods and hand grasp. Left: Similar to using a white cane, the forehand gesture allows the user to explore his surrounding environment with the rod extended. Right: The user holds the rod upright in a crowded scenario.

D. Contributions

We summarize our contributions as follows:

1) A robotic guide dog that extends the reliability of a white cane and is able to autonomously follow various man-made trails for BVI people in pedestrian environments.
2) Deep trail-following models trained using data from real-world and virtual environments, which are robust to various background textures, illumination variances, and interclass variations.
3) A user study with 10 BVI users, who were able to complete trail-following tasks.

II. The Proposed System

A. Robotic Guide Dog Design Considerations

Because of its reliability, the white cane is the most commonly used mobility aid in the BVI community; therefore, we wished to maintain its reliability and extend its capability. There have been many wearable solutions, including our previous work [4]. A wearable device has advantages, such as small size and light weight, but its computation capability and battery life are highly limited. A robotic guide dog can carry a heavier computation device and batteries. More importantly, a robotic guide dog provides reliable physical human-robot interaction, so we designed the cane-like rod and hand grasps shown in Fig. 4. The design was inspired by the use of a white cane, where a user either reaches farther away for exploration or holds the rod upright in a crowded environment.

We wish to learn the mapping between an image of center camera obtained from the robot's heading and lateral distance of the trail and its desired motion during prediction. The settings of 3 cameras make it easier to collect 3 observations of headings at the same time for training. For example, the images collected from the left camera, representing the observations of 30 degrees left of the robot heading during prediction, are labelled "Turn Right." We collect data with the robotic guide dog, train it offline with a workstation equipped

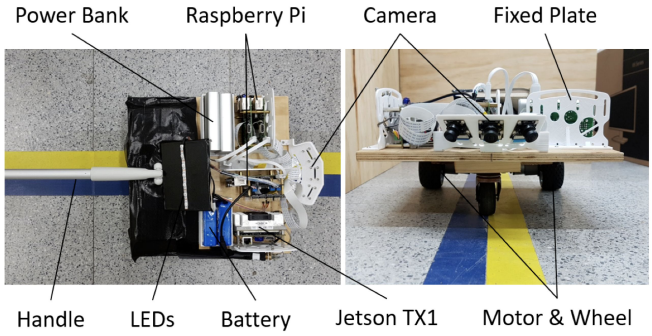


Fig. 5: The proposed robotic guide dog. There are three cameras, three Raspberry Pi2s, and one Jetson TX1 embedded computer onboard. During data collection, all three cameras are used, whereas only one camera input is used while performing prediction/inference.

NVidia GTX 1080 GPU, and then load the trained network onto the embedded system for prediction.

The proposed robotic guide dog is shown in Fig. 5. Its components include three Raspberry Pi fisheye cameras; each camera is connected to a Raspberry Pi 2 Model B embedded computer through its Camera Serial Interface (CSI) port. The three cameras installed on the robotic guide dog with headings of -30, 0, and 30 degrees are used for training data collection. One of the three Raspberry Pi2s is connected to DC motors with pulse width modulation control using Duckietown software, whereas the other two are responsible for ROS data logging only. We use an NVIDIA Jetson TX1 embedded system with 4 GB of GPU/CPU memory for onboard prediction. The dimensions of the robot are $34\ cm \times 30\ cm \times 18\ cm$ (length, width, and height); it weights about four kilograms, including batteries. The wheels of the robot are designed to drive on indoor or outdoor terrains. Complete with high-torque motors, it is designed to support the handle tension between the robot and user.

B. Behavior Reflex for Motion Primitives

The system workflow is shown in Fig. 6. The deep CNN model of behavior reflex directly maps the input image into three output classes as motion primitives: "Turn Left", "Go Straight", and "Turn Right". To keep the robotic guide dog following the trail, the left camera is labeled as "Turn Right", the middle camera as "Go Straight", and the right camera as "Turn Left." A sigmoid function with a gain of 9 was used in Eq. 1 to map the prediction probability of the deep CNN model to the trim of the angular velocity of the robotic guide. Only the "Turn Left" prediction probability P_{tl} and "Turn Right" prediction probability P_{tr} were considered in calculating the trim of the angular velocity using Eq. 2. With the trim from Eq. 2 and a fixed linear velocity of 0.38, we applied the inverse kinematics method in [13]–[15] to obtain the motor rates for the robotic guide dog. Those settings in Eq. 1 have been found to work empirically.

$$T(x) = \frac{9}{1 + e^{\ x}} \tag{1}$$

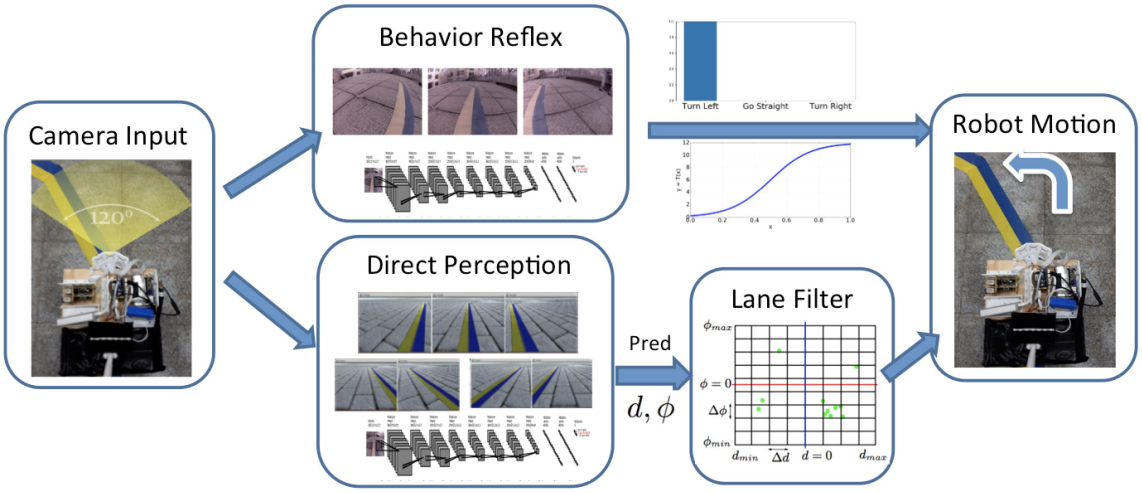


Fig. 6: The proposed system workflow. We trained the deep CNN models in two ways: 1) the "behavior reflex" maps the three prediction probabilities to turn left, go straight, and turn right motions, and 2) the "direct perception" trains a model with seven output classes corresponding to various robot states of lateral distance d and heading ϕ. The states are then voted into a lane filter [15] and lane controller to align with the trail.

$$trim = 1.2(T(P_{tl}) - T(P_{tr})) \tag{2}$$

C. Direct Perception for State Estimation

In direct perception, we wished to estimate the robot state using the lateral distance d and heading ϕ directly from a camera input. Again, we took advantage of the three cameras of the robotic guide dog and collected images as it followed the trails using three lateral distances $-d$, 0, and d. With the $-d$ lateral distance, the input images from the middle and right cameras were labeled as "L_S" and "L_R" respectively, as shown at the bottom left of Fig. 8. For the 0-lateral-distance scenarios, as shown in the bottom middle of Fig. 8, the input camera images from the left to right cameras were marked as "C_L", "C_S", and "C_R" respectively. With d lateral distance, the images from left camera were labeled as "L_R" and those of middle camera are labeled "L_S," as shown in the bottom right of Fig. 8. The images from the left camera with $-d$ distance and the right camera with d distance were ignored. Such settings resulted in seven output classes in the deep CNN model of direction perception. We utilized prediction probability of the seven output classes to generate votes on the measured likelihood of lateral distance and heading angle based on the lane filter in [13]–[15]. The measured likelihood resulted in estimates of lateral distance and heading angle for the next controller stage by considering the current votes and last controlling motion using a nonparametric Bayes filter. The controller stage based on the lane controller in [13]–[15] converted the estimates of lateral distance and heading angle to a linear velocity and angular velocity trim for the robot.

III. Deep Trail-Following Networks

A. Real-World Training Datasets

We designed two types of man-made trails: the yellow-blue trail (YB) and the Freedom Trail (FT). The YB trails are reusable yellow and blue tapes, around 9.6 cm in width. The FT trails were made by printing the textures of the Boston Freedom Trail on A4-sized pages around 21 cm in width. The real-world training datasets were collected by manually operating the robotic guide dog following the YB and FT trails with various backgrounds and illumination conditions, in both indoor and outdoor environments, as shown in Fig. 7. With additional data augmentation by flipping the collected dataset, the real-world training dataset of the YB trail for behavior reflex and direct perception contained 18,000 and 42,000 images respectively.

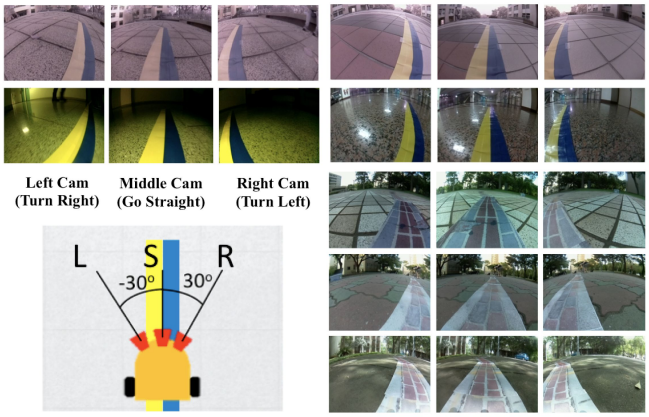


Fig. 7: Data collection in real world. We prepared two types of trails (yellow-blue trail and the Freedom Trail) by printing them on A4-sized pages and then attached them together in one- or two-meter lengths. We then stitched them together and placed them in natural environments. Top two rows: camera inputs of the yellow-blue trail. Bottom-right: camera inputs of the Freedom Trail.

B. Virtual-World Training Datasets

Deep learning has rapidly developed to address a variety of problems, but such an approach has relied upon massive

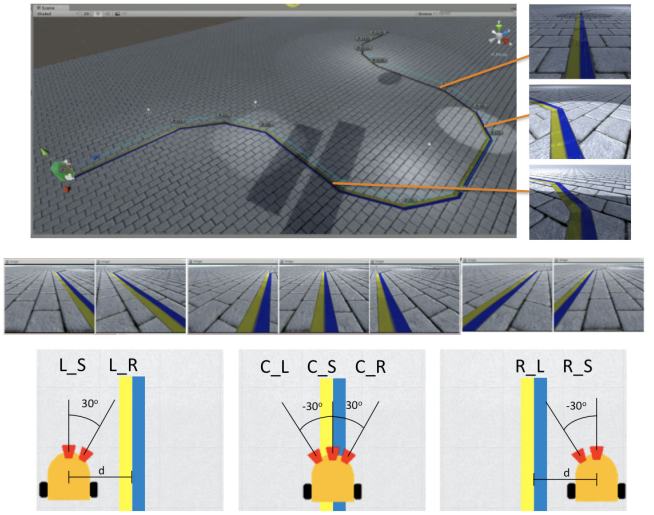


Fig. 8: Data collection in the virtual world. We rendered camera inputs from a virtual world including a background texture with some regions covered by light and shadow. Nine virtual cameras were placed at lateral distances $-d$, 0, and d and headings -30, 0, and 30 degrees, with two outward cameras not used, resulting in seven classes of training data.

amounts of human-annotated training data, which became a bottleneck for adapting to new applications. References [17] and [18] used synthetically rendered datasets and showed improved results. We wished to obtain the training data for trail-following in a wide range of appearances and backgrounds. As shown in Fig. 8, we set up seven virtual cameras and followed a YB trail under different light conditions and shadows. The images of the seven virtual cameras: L_S, L_R, C_L, C_S, C_R, R_L, and R_S are shown in Fig. 8.

To simulate real-world scenarios in virtual environments, we employed background textures similar to those that occur in natural scenes, such as brick and wood. Illumination variances, spotlighting, and shadows generated by objects in the air resulted in different illuminations on the trails, as shown in the rightmost column of Fig. 8. To navigate automatically in virtual environments, we set waypoints along the trail so that the motions of the cameras could be estimated depending on the current position and next waypoint. The input images of the cameras were transferred as Robot Operating System (ROS) compressed image messages to crop the training samples. We also augmented the training data by flipping the collected dataset, the virtual-world training dataset of the YB trail for behavior reflex and direct perception contained 9,600 and 11,200 images, respectively.

C. Deep CNN Model Architectures

We used two model structures to perform deep trail-following for the BVI: CaffeNet [19] and TrailNet [16]. CaffeNet consists of four convolution layers, four max-pooling layers and three fully connected layers; the input channel is $3 \times 227 \times 227$, and the size of the Caffe [20] platform is about 220 MB. TrailNet is a relatively small network with five convolution layers, three max-pooling layers, and three fully connected layers. The gray-level input channel is $1 \times 101 \times 101$, and the size of TrailNet is 845 KB, which is significantly smaller than CaffeNet. We also tested the RGB inputs for the TrailNet input channel $3 \times 101 \times 101$ as TrailNet-Color to execute the recognition task.

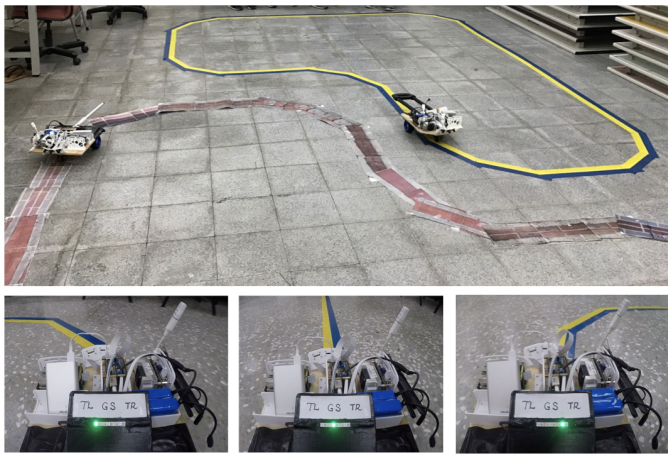


Fig. 9: Top: We test the trained models on the proposed robotic guide dog in one YB trail and one FT trail. Bottom: The LED lights are designed to indicate the deep CNN prediction results and estimate onboard computation latency.

IV. Model Evaluations

A. Model Deployment in the Robotic Guide Dog

We deployed the trained models from both the real and virtual environments for onboard prediction on the proposed robotic guide dogs. The confusion matrix of the validation showed a high accuracy of classification (all greater than 0.97). The proposed robotic guide dogs were then tested in an environment with one loop of YB and the other of FT trails, as shown in Fig. 9. Each loop was around 10 meters long including 6 turns. We carried out testings of 5 trials, each required the robots to stay on the trails for 20 loops. There are 5 trails on YB and 5 trails on FT lines, and all trials were completed. We also inspected the LED lights on the robots, which indicated the onboard computation latency. Our system run at 6 frames per second, and we found that the maximum latency should be smaller than 200 ms from empirical testings.

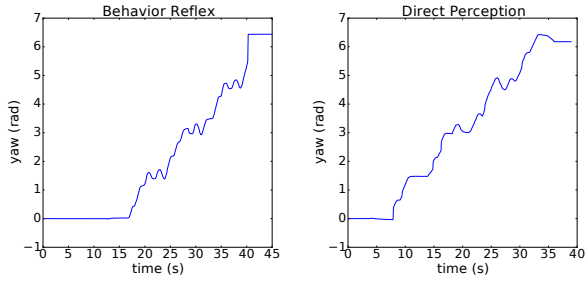
B. Virtual Environment Evaluations

A virtual environment offers a safe and reproducible experiment in dynamic environments, such as train stations. Such an evaluation is usually difficult to carry out in real environments due to safety considerations. We constructed a 3D model of Kenmore station in Boston, MA, in which to conduct our model evaluation (see Fig. 10). The virtual environment included a few virtual pedestrians walking around, which caused shadows. At that time, we did not implement a "Stop" command when the virtual camera encountered a virtual pedestrian. The trained deep CNN models tested in virtual environments are shown in the supplementary video.

C. Behavior Reflex vs. Direct Perception

We further tested the model performances with the behavior reflex and direct perception settings. We designed another YB trail as a rectangular loop with 4 turns (2π), around 10.12

Fig. 10: Virtual prediction in Kenmore station. Left: Kenmore station with people walking randomly. Right: Motion for virtual prediction: Top: Go straight; Middle: Some virtual pedestrians walking around during a left turn; Button: A right turn under different lighting conditions.



(a) Behavior reflex paradigm. (b) Direct perception paradigm.

Fig. 11: Heading (yaw) estimated by a Google Tango device mounted on the robotic guide dog over time in a rectangle loop. (a) Behavior reflex paradigm; (b) Direct perception paradigm. We accumulated Δyaw from t to $t+1$, and found that the direct perception paradigm was more stable.

meters in length. A Google Tango device was mounted on the robotic guide dog for the purposes of obtaining visual inertial odometry (VIO) as ground truth. We show the yaw-versus-time curves of the behavior reflex and direct perception paradigms in Fig. 11a and in Fig. 11b, respectively. The robot heading changes, estimated by the yaw of the VIO, are used to indicate the stability of the paradigms. The accumulated Δyaw from t to $t+1$ used as measure, and a lower value indicates the desired overall gentler turns. The measures were found 9.77 in the behavior reflex paradigm, and 8.68 in the direct perception paradigm. Both paradigms tend to cause some orientation errors compared with the ideal case (6.28 in a rectangular loop with 4 turns), but the direct perception paradigm shows smaller measure and is more stable than behavior reflex.

V. User Studies

We conducted user studies with 10 BVI users, who did not have prior experience with our system. All were recruited through an association of the blind community in Taiwan. All 10 participants were BVI (nine were blind and one was visually impaired, who had a very limited field of view and could not see the experimental trail). The user study was conducted in a $4.5\ m \times 4\ m$ classroom, with an S-shaped (two

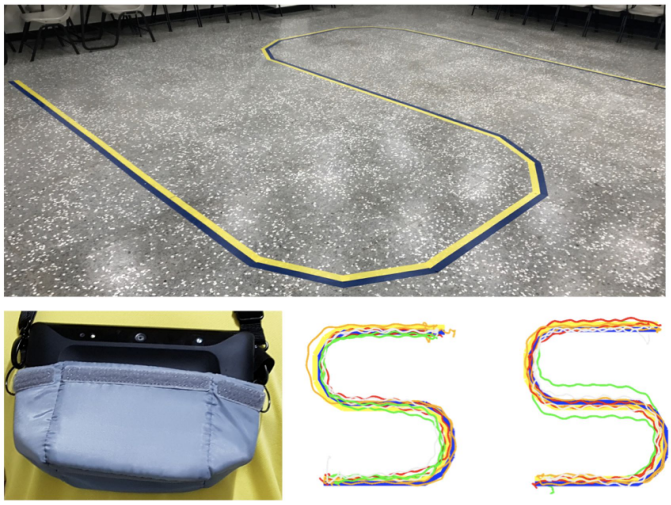


Fig. 12: Top: The experiment environment is designed as a S-type map made of a yellow-blue trail. Bottom left: A Google Tango device is hung around the neck of a participant to record the user's movements. Bottom, middle, and right: The trajectories of users' movements for the forehand and upright grasps, respectively.

left and two right turns) experimental trail with a total length of $15.4\ m$ (see Fig. 12). Each participant was introduced to the tasks and signed a consent form. They were instructed to follow the robotic guide dog by holding a cane-like rod in two hands, as shown in Fig. 4. The robotic guide dog was set to the "behavior reflex" paradigm. We also instructed the participants to respond to the experimenter when they encountered a turn. A Google Tango device was hung around each user's neck to capture and record their movements, average walking speed, and time to completion. We also used a questionnaire to capture the users' subjective experience.

Each participant followed and walked twice along the experimental trail. They used a forehand grasp for their first attempt and an upright grasp for the second attempt, which included four left turns and four right turns in total. The trajectories of the forehand and upright grasps are shown in Fig. 12. All participants finished the trail-following and correctly responded to the eight turns, except one user who tended to hold the rod too tightly, thereby pulling the robotic guide dog away from the trail. There were three incorrect verbal responses: one during a right turn and two during straight segments, caused by the drifting of the robot motions.

All 10 participants completed the questionnaires after the experiments. The 5-point scales questionnaires were designed to reflect the following aspects of a mobility aid: 1) real time, 2) portable, 3) reliable, 4) cost-effective 5) friendly, 6) interaction, 7) speed, and 8) mental map, where 1-5 were suggested by Dakopoulos and Bourbakis [2]. For example, the question "Do you feel reliable about the guiding robot?" for the reliable aspect. The results in Fig. 13 show that participants appreciated real-time guidance, reliable feedbacks, and friendly interface without pre-training. Some participants wanted the robotic guide dog to have stronger interaction and faster speed, which would be a trade-off on the portability considerations (weights of the robot). We set the cost of the

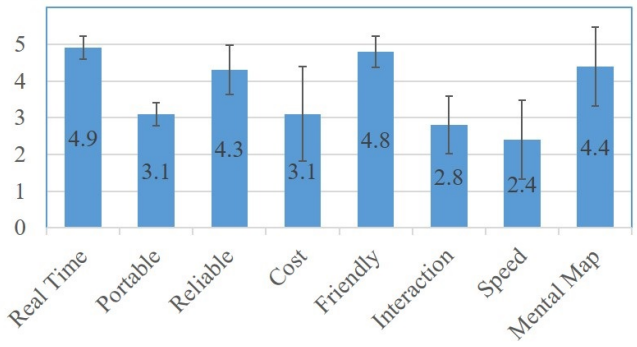


Fig. 13: Questionnaires results of the 5-point scales questionnaires.

guide dog robot to USD 500, which was acceptable for most users. There is a high variation in the number of participants who were able to build and draw a mental map; some could not. Overall, the participants said the reliability was high, which aligned well with our design considerations.

VI. CONCLUSIONS

We demonstrated a reliable robotic guide dog system, which was able to learn and follow various man-made trails. The proposed deep CNN approaches of learning from both real and virtual worlds overcame the challenging scenarios of a vision system in a real-world environment, including illuminations, shadows, different background textures, and interclass variations. We also tested the trained model within the framework of a virtual environment, which made it possible to consider safety and advance the robustness in dynamic environments, such as a train station. The user questionnaires indicated that the 10 BVI users favored the real-time, reliable, friendly aspects of our proposed robotic guide dog. In future work, we will further examine and visualize the trained CNN models with different settings of training data. We will also incorporate system recovery from failure, and test the system in realistic environments, such as paved pathway.

ACKNOWLEDGMENT

The research was supported by Ministry of Science and Technology, Taiwan (grant numbers 105-2218-E-009-015 and 105-2511-S-009-017-MY3). We are also grateful for the help by Santani Teng, Robert Katzschmann, Ilenia Tinnirello, and Daniele Croce.

REFERENCES

[1] World Health Organization. [Online]. Available: http://www.who.int/mediacentre/factsheets/fs282/en/

[2] D. Dakopoulos and N. G. Bourbakis, "Wearable obstacle avoidance electronic travel aids for blind: a survey," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 40, no. 1, pp. 25–35, 2010.

[3] R. Tapu, B. Mocanu, and E. Tapu, "A survey on wearable devices used to assist the visual impaired user navigation in outdoor environments," in *Electronics and Telecommunications (ISETC), 2014 11th International Symposium on*. IEEE, 2014, pp. 1–4.

[4] H.-C. Wang, R. K. Katzschmann, S. Teng, B. Araki, L. Giarré, and D. Rus, "Enabling independent navigation for visually impaired people through a wearable vision-based feedback system," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 2017, pp. 6533–6540.

[5] International Guide Dog federation (IGDF). [Online]. Available: https://www.igdf.org.uk/

[6] V. Kulyukin, C. Gharpure, J. Nicholson, and G. Osborne, "Robot-assisted wayfinding for the visually impaired in structured indoor environments," *Autonomous Robots*, vol. 21, no. 1, pp. 29–41, 2006.

[7] C. P. Gharpure and V. A. Kulyukin, "Robot-assisted shopping for the blind: issues in spatial cognition and product selection," *Intelligent Service Robotics*, vol. 1, no. 3, pp. 237–251, 2008.

[8] C. Rasmussen, Y. Lu, and M. Kocamaz, "A trail-following robot which uses appearance and structural cues," in *Field and Service Robotics*. Springer, 2014, pp. 265–279.

[9] I. Ulrich and J. Borenstein, "The guidecane-applying mobile robot technologies to assist the visually impaired," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 31, no. 2, pp. 131–136, 2001.

[10] C. Rasmussen, Y. Lu, and M. Kocamaz, "Appearance contrast for fast, robust trail-following," in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*. IEEE, 2009, pp. 3505–3512.

[11] C. Siagian, C.-K. Chang, and L. Itti, "Mobile robot navigation system in outdoor pedestrian environment using vision-based road recognition," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 564–571.

[12] A. B. Hillel, R. Lerner, D. Levi, and G. Raz, "Recent progress in road and lane detection: a survey," *Machine vision and applications*, vol. 25, no. 3, pp. 727–745, 2014.

[13] Duckietown. [Online]. Available: https://duckietown.mit.edu/

[14] Duckietown NCTU. [Online]. Available: https://duckietown.nctu.edu.tw/

[15] L. Paull, J. Tani, H. Ahn, J. Alonso-Mora, L. Carlone, M. Cap, Y. F. Chen, C. Choi, J. Dusek, Y. Fang, D. Hoehener, S.-Y. Liu, M. Novitzky, I. F. Okuyama, J. Pazis, G. Rosman, V. Varricchio, H.-C. Wang, D. Yershov, H. Zhao, M. Benjamin, C. Carr, M. Zuber, S. Karaman, E. Frazzoli, D. D. Vecchio, D. Rus, J. How, J. Leonard, and A. Censi, "Duckietown: an Open, Inexpensive and Flexible Platform for Autonomy Education and Research," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 2017.

[16] A. Giusti, J. Guzzi, D. C. Cireşan, F.-L. He, J. P. Rodríguez, F. Fontana, M. Faessler, C. Forster, J. Schmidhuber, G. Di Caro, *et al.*, "A machine learning approach to visual perception of forest trails for mobile robots," *IEEE Robotics and Automation Letters*, vol. 1, no. 2, pp. 661–667, 2016.

[17] C. Chen, A. Seff, A. Kornhauser, and J. Xiao, "Deepdriving: Learning affordance for direct perception in autonomous driving," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 2722–2730.

[18] M. Johnson-Roberson, C. Barto, R. Mehta, S. N. Sridhar, K. Rosaen, and R. Vasudevan, "Driving in the Matrix: Can virtual worlds replace human-generated annotations for real world tasks?" in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 2017, pp. 746–753.

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105. [Online]. Available: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

[20] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the ACM International Conference on Multimedia*. ACM, 2014, pp. 675–678.