ScienceDirect

IFAC PapersOnLine 52-30 (2019) 102–107

End-to-end Learning for Autonomous Crop Row-following [★]

Marianne Bakken [*,**] **Richard J. D. Moore** [*] **Pål From** [**]

[*] *SINTEF Digital, Oslo, Norway (email: marianne.bakken@sintef.no)*
[**] *Norwegian University of Life Sciences (NMBU), Ås, Norway*

Abstract: For robotic technology to be adopted within the agricultural domain, there is a need for low-cost systems that can be flexibly deployed across a wide variety of crop types, environmental conditions, and planting methods, without extensive re-engineering. Here we present an approach for predicting steering angles for an autonomous, crop row-following, agri-robot using only RGB image input. Our approach employs a deep convolutional neural network (DCNN) and an end-to-end learning strategy. We pre-train our network using existing open datasets containing natural features and show that this approach can help to preserve performance across diverse agricultural settings. We also present preliminary results from open-loop field tests that demonstrate the feasibility and some of the limitations of this approach for agri-robot guidance.

Keywords: Robot vision; Robot navigation; Machine learning; Agricultural robotics; Mobile robots

1. INTRODUCTION

Automating agricultural practices through the use of robots (i.e. agri-robots, Fig. 1) is a key strategy for improving farm productivity and achieving sustainable food production to meet the needs of future generations. However, modern food production techniques have resulted in diverse growing environments – from greenhouses and polytunnels to open fields (Fig. 2) – presenting a significant technological challenge for the development of generally-useful agri-robots.

In order for autonomous agri-robots to be a realistic and cost-effective alternative for the end-user (i.e. farmers), they must overcome the following challenges:

(1) Accurate navigation to maximise efficiency and avoid damaging crops.
(2) Flexibility to support various environments, crop types, and environmental conditions.
(3) Minimal setup and installation cost.
(4) Safe and reliable operation, including intelligent response to unexpected conditions or events.

Accurate and flexible navigation can be achieved with external localisation systems such as D-/RTK-GNSS (e.g. Perez-Ruiz and Upadhyaya (2012)), but such systems require a network of base stations to provide real-time correction data as well as a precise map of crop locations and are therefore expensive to install. Visual-inertial navigation (V-INS) or visual- or lidar-based SLAM systems do not rely on external hardware and have been demonstrated on board agri-robots (e.g. Le et al. (2019)), but they also require a precise map of crop locations and can suffer from coordinate frame drift in agricultural settings due to the typically long non-overlapping trajectories and self-similar environments. To meet the above challenges, it is therefore interesting to investigate local (i.e. onboard) navigation solutions that provide direct guidance relative to crop locations and have no external hardware dependencies.

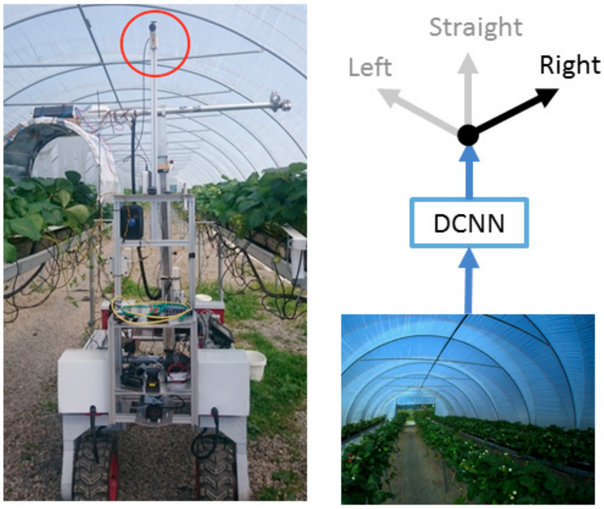

Fig. 1. The mobile robot recording setup: A forward-facing, wide-angle video camera (red circle) mounted above the robot was used to capture images for offline training as well as online field tests. A DCNN was trained to predict view orientation for autonomous row following in agricultural environments.

Existing local sensing approaches typically aim to segment the scene into vegetation and non-vegetation classes based on either 2D image data from RGB (Bakker et al. (2008); Jiang et al. (2010)) or NIR (Halmetschlager et al. (2014)) cameras, or 3D data from stereo systems (Kise et al.

[★] This work was funded by The Norwegian Research Council, grant number 259869.

2405-8963 Δ 2019, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved.
Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2019.12.505

(2005)) or scanning LIDAR (Biber et al. (2012)). The vehicle's heading or lateral offset from the preferred trajectory can then be computed by leveraging the typically linear layout of crop plantations (Kise et al. (2005); Bakker et al. (2008); Jiang et al. (2010); English et al. (2015)). However, 3D methods do not perform well when crops are too sparse or too dense, and 2D methods traditionally employ handcrafted features, or features that are specific to a particular crop type and/or cultivation method, and thus do not generalise well to other agricultural settings.

Much recent work has shown that deep convolutional networks (DCNNs) are able to learn optimised image features for a wide range of classification and segmentation tasks even in poorly structured scenes, so it is therefore interesting to investigate whether DCNNs can enable more general solutions for crop row following than traditional approaches. However, DCNN-based approaches require feature appearance variation to be captured at training time, which means data capture and annotation is a time-consuming and costly process.

Our hypothesis is that by pre-training a DCNN on available data sets with non-specific vegetation features, e.g. forest trails (Giusti et al. (2015)), we can reduce the amount and diversity of agricultural-specific training data required and additionally improve the ability of the network to generalise to other agricultural use cases. Our preliminary work (Bakken et al. (2018)) has shown that a network trained on forest trail data can be relatively simply adapted for crop row following in various settings by fine-tuning with a small amount of agricultural-specific training data. Here we expand on this work to compare the performance of our pre-trained and fine-tuned network with a network trained fully on agricultural data to show that our approach retains better generalisation capability.

Additionally, since our ultimate aim is to establish whether this approach can be used successfully to provide guidance for an autonomous agri-robot in diverse settings, we implement our approach on board a test platform for field testing. We propose to use an *end-to-end* learning strategy to train our DCNN to output control commands for our autonomous vehicle directly from RGB image input data, following on from the work of Giusti et al. (2015) and Loquercio et al. (2018), who showed that end-to-end learning can be successfully employed to overcome the problem of designing control policies for autonomous platforms in scenes with widely varying appearances. Here we present preliminary results from initial open-loop field trials that demonstrate the feasibility and some of the limitations of our approach.

2. RELATED WORK

Recently, there has been an increasing amount of work on learning control policies directly from RGB images using deep neural networks. Reinforcement learning in particular has seen great success in game settings, but requires an extremely large number of training examples, which are usually not possible to collect in real environments. Transfer of networks trained purely on simulated data to real-world scenarios has been successfully demonstrated by Sadeghi and Levine (2016), but with such diversity in environments and tasks, building a full simulation environment for all agricultural applications is not feasible.

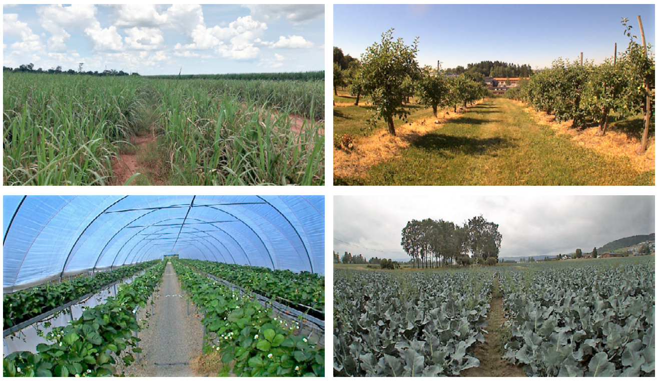

Fig. 2. Diverse agricultural scenes with crop rows (t-b, l-r): sugarcane, apple, strawberry, broccoli. Rows are often not easily identifiable and can change rapidly in appearance – presenting a difficult challenge for autonomous robots. We use the strawberry polytunnel case as a controlled environment for initial testing, but have designed our approach to generalise to other agricultural scenes.

Supervised learning approaches require fewer samples compared to reinforcement learning, but the samples must be labelled and/or the networks pre-trained on datasets such as ImageNet (Deng et al. (2009)) or Pascal VOC (Everingham et al. (2010)). However, features from the agricultural domain are not well represented by such datasets and so large amounts of new data would need to be collected and annotated, which is both time-consuming and expensive.

Supervised end-to-end learning of high-level control policies directly from RGB input has shown great promise in alleviating the difficulty of annotating training data (e.g. guidance of aerial robots, Loquercio et al. (2018)). However, for autonomous control of aerial robots it is often not practical to acquire accurate ground truth labels for data captured in flight, thus Loquercio et al. (2018) use a car driving dataset from Udacity[1] to train a drone to follow roadways. Their approach uses a network with regression output layer that gives a continuous output and can be taught a wide range of control policies, but requires an expert driver for data capture. In a recent publication, Kaufmann et al. (2018) trained a similar network for drone racing by carrying a drone around a race course to collect training data, which did not require expert steering, but depended on additional sensors and an offline state estimation for data labelling. A simpler data collection approach was employed by Giusti et al. (2015), who collected an extensive dataset for prediction of view orientation on forest trails with a head-mounted three-camera rig, which gave a built-in labelling of orientation (left/straight/right). Based on this, they trained a view orientation classifier, which was used to compute yaw control of a drone from RGB images only. Smolyanskiy et al. (2017) developed this approach further by experimenting with different network architectures and adding lateral control, which showed improved performance and indicated good generalisation

[1] Available at https://github.com/udacity/self-driving-car

capabilities within the trail domain. This approach allows ground truth commands to be easily generated during data capture, but is limited somewhat by the type of steering commands that can be learned.

We expect that the IDSIA trail dataset [2] from Giusti et al. (2015) better captures natural features such as vegetation and soil than for instance the Udacity city driving dataset. Thus, we select this dataset for pre-training our DCNN.

3. METHODOLOGY

We propose to apply the the principles from the trail following method of Giusti et al. (2015) to crop row-following in agriculture, and use trail data for pre-training to reduce the amount application-specific agricultural training data needed. In our preliminary study (Bakken et al. (2018)), this showed promising results on a very limited polytunnel dataset. In this paper, we expand our dataset substantially to polytunnels from several different locations and seasons. We have also extended the label generation procedure and the network architecture to work for regression, in order to provide a continuous output angle. To assess our network's ability to generalise beyond one single location and setting, controlled experiments are performed with networks trained on differing amounts of polytunnel training data and tested on locations not seen during training. We also assess the performance of our approach for guidance of an autonomous agri-robot with open-loop field tests.

3.1 Network architecture

Our approach is based on Giusti et al. (2015), using the VGG network architecture (Simonyan and Zisserman (2014)) with three output classes and dropout on fully-connected layers. We also implemented a regression network based on this architecture, with one continuous output value from the last layer instead of a three-class output. An overview of our network architecture is shown in Fig. 3.

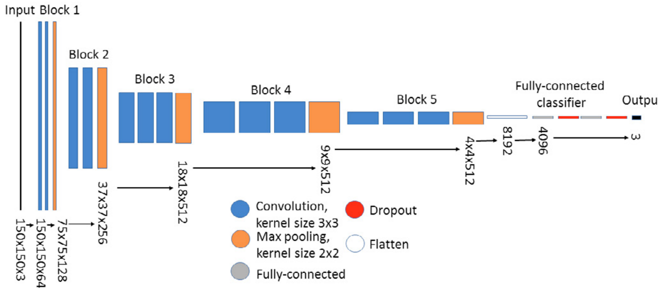

Fig. 3. Network architecture for VGG16 with dropout on fully-connected layers and 3 output classes. For the regression version of the network, the output size is one instead of three.

3.2 Datasets and label generation

Trail data set The IDSIA Swiss Alps trail data set from Giusti et al. (2015) consists of several kilometres of trail recordings. The images are recorded with a rig of three cameras looking left, straight, and right that provides the ground truth labels for the three viewpoint classes. The data contains different kinds of trails, and also has some road sections. Most recordings are from the same season, probably late autumn, and have very little green vegetation. See Fig. 4 for example images. As in Smolyanskiy et al. (2017), we used the folders 003, 008, and 010 for testing (*trails test*), and the remainder for training (*trails training*).

Agricultural datasets We recorded a new data set for row following in strawberry polytunnels, with a recording approach similar to that described for the trails data set. The data was captured from five different polytunnels, totalling 3 km of recordings at 5 fps. Our recording setup was a Basler Ace camera with Sunex 190 degree field of view (FoV) fisheye lens. For most recordings, this was mounted approximately 2 m above ground level with a downward tilt of 25 degrees. Some recordings were performed with lower height and less tilt for more variation. Video sequences were recorded travelling straight along the centre of each row, either on board a mobile agricultural robot (Fig. 1) or by hand.

The strawberry polytunnel dataset is divided into three subsets for training and testing: *single polytunnel* consists of data from one row within a single polytunnel at a single point in the growth cycle; *diverse polytunnels* includes data from other rows within the same tunnel as *single polytunnel* as well as three additional tunnels; and *polytunnel test* consists of data from a separate location and season and is used only for testing. For each tunnel, data has been recorded in both directions. Example three-class images from two of five different polytunnels are shown in Fig. 4.

Label generation By employing a wide-FoV camera instead of a fixed rig of three cameras as in Giusti et al. (2015), we are able to extract virtual camera views from arbitrary angles after the recordings were made. This gives much more flexibility than a fixed rig, and makes it possible to train a continuous regression output. Our procedure for extracting virtual camera views was integrated directly into the Keras image augmentation pipeline, such that roll, pitch, and yaw angle offsets could be specified and corresponding virtual views extracted directly at training and test time. In our regression training setup, we specify a fixed roll and pitch and three random yaw angles between -27 and 27 degrees with a 140 degree FoV, per full-FoV image. For classification, three fixed yaw angles of -27, 0 and 27 was used per full-FoV image.

3.3 Training procedures

Our network was pre-trained on the *trails* dataset with additional regularisation, followed by fine-tuning the fully-connected layers of the DCNN (Fig. 3) on our own polytunnel data. A full description of the training setup as well as experiments with hyperparameters and regularisation is given in Bakken et al. (2018). For comparison, we also performed training from scratch on polytunnel data only, using the same setup as with the *trails* dataset, but with a slightly smaller learning rate $1 \cdot 10^{-6}$, until a loss plateau was reached.

For our regression network, the loss function was changed to mean-squared error. The weights in all other layers from

[2] Available at `http://people.idsia.ch/~giusti/forest/web/`

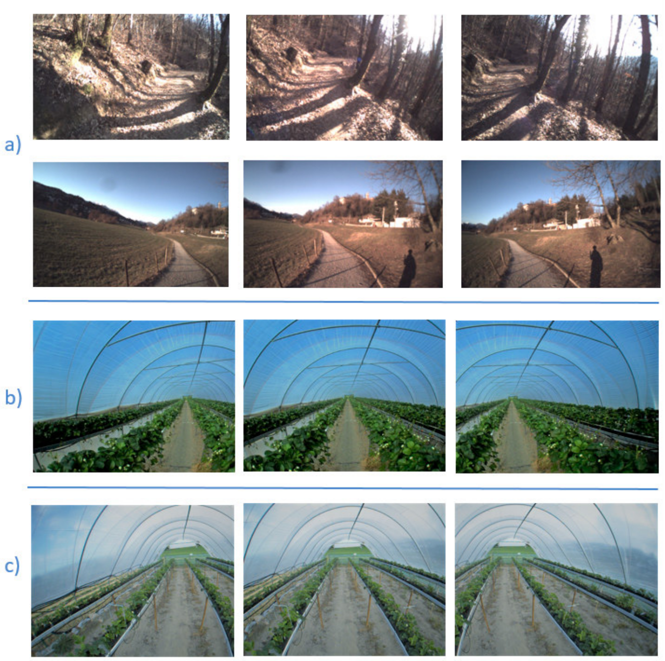

Fig. 4. Example left, straight and right class images from the different datasets: a) *trails*, and two different tunnels from our strawberry polytunnel datasets: b) *single polytunnel* and c) *polytunnel test*

the trails pre-training were kept (since the trails data has only discrete labels), but the fine-tuning was run on our own data with regression labels. The learning rate was slightly lower than for the classification network, $4 \cdot 10^{-7}$, and the the training was run until a loss plateau was reached.

3.4 Classification experiments

We compared the classification performance of our network under two different training regimes:

(1) trained using only data from the *single polytunnel* set, and
(2) pre-trained on the *trails training* dataset and then fine-tuned using the *single polytunnel* dataset.

Both trained networks were then tested for classification accuracy (against left/right/straight ground truth steering angles) on two data sets (see section 3.2):

(1) the *polytunnel test* set, containing an unseen polytunnel at a different phase in the growing cycle (significantly different vegetation density) to that of the training set, and
(2) the *trails test* set containing unseen forest trails.

The same experiments were repeated with training data from the *diverse polytunnels* training set.

3.5 Preliminary field trials

Preliminary field trials were performed in a strawberry tunnel similar to the *polytunnel test* set (not seen during training) with the same robot and camera setup as for data collection. We integrated our DCNN into a ROS node that received a live image stream from the camera and predicted steering angles in real-time. Our ROS node executed on a laptop CPU on board the mobile robot platform, with a rate of 9 Hz. During this preliminary testing, we operated our system open loop and the robot was steered manually at a speed of 0.4 m/s along a slalom path between crop rows to allow qualitative performance analysis on live data.

4. EXPERIMENTAL RESULTS

Here we present both quantitative results from offline analysis as well as qualitative results from online field testing. We first compare the steering angle classification accuracy of our network pre-trained on forest trail data and fine-tuned on agricultural data with a network trained fully on agricultural data. We then investigate the usefulness of this approach for steering an autonomous robotic platform performing crop row following, and compare discrete classification network output with continuous regression output.

4.1 Classification accuracy in diverse settings

The results from the classification experiments (described in section 3.4) are summarised in Table 1.

Table 1. Classification accuracy for different training regimes and test cases.

Training data set	**Classification accuracy (%)**	
	Polytunnel test	Trails test
Single polytunnel	84.0	31.5
Trails + single polytunnel	78.5	85.7
Diverse polytunnels	99.5	48.1
Trails + diverse polytunnels	97.9	78.6

When trained using only data from the *single polytunnel* set, our DCNN performed well in other polytunnel environments, despite the unseen variation in tunnel appearance, camera angle, and vegetation density. However, the same trained network was not able to transfer at all to the general vegetation scenes present in the *trails test* dataset. On the other hand, by pre-training on the *trails training* set and fine-tuning on *single polytunnel* data, we were able to also achieve reasonable performance in all polytunnel environments whilst preserving good performance for general vegetation scenes. Increasing the diversity and amount of polytunnel data using for training (*diverse polytunnels*) enabled our DCNN to improve its classification performance on the general vegetation scenes in the *trails test* set, but still fell well short of the performance of the network pre-trained on *trails* data. These results suggest that the features learned by the network pre-trained on *trails* data are more general than those learned by the network trained only on a specific agricultural setting (*single polytunnel*), and that this approach should therefore generalise more readily to diverse agricultural settings.

4.2 Preliminary field trials

To test the robustness of our fine-tuned network to real world conditions, we implemented our DCNN on board a mobile robot (section 3.5) and drove it through the tunnels found in the *polytunnel test* set, which were not presented

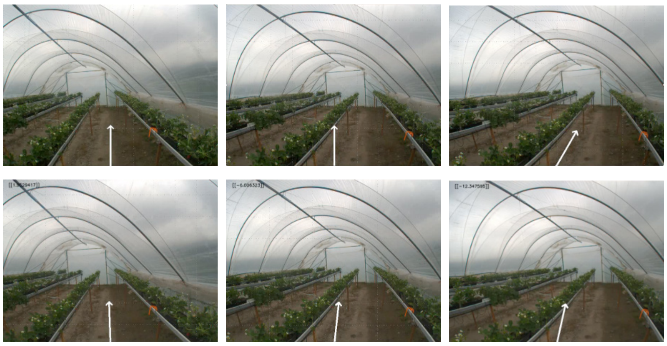

Fig. 5. Snapshots from ROS visualisation video showing predicted steering angle from a classification network (top) and a regression network (bottom) as the robot is turning left. The classification output does not respond to the moderate angle in the middle image. The regression output gives some response to moderate angles, but underestimates both the moderate and the large angle.

during training and included significant seasonal and other differences. Quantitatively, the network fine-tuned on *diverse polytunnels* showed very good performance when tested on the *polytunnel test* set, which contains only three possible steering angles ([-27, 0, 27] degrees, as for the trail data in Giusti et al. (2015)) (Table 1 lower). However, qualitative analysis (Fig. 5) from field tests showed that the sensitivity of the network to smaller steering angle disturbances was not sufficient for autonomous crop row following.

4.3 Regression for continuous steering

To address the poor sensitivity of our network, we re-implemented the top-most layer in our network to give single continuous output (section 3.1). When fine-tuned on the *diverse polytunnels* training set and tested on the unseen *polytunnel test* set, we achieved an RMSE of 5.8 degrees compared to ground truth viewing angles. Qualitative assessment of the field test data shows a much improved sensitivity to steering angle disturbances compared to the classification network, but the regression network has a tendency to underestimate the viewing/steering angle.

5. DISCUSSION

It is important for our use case that the agri-robot is capable of adapting to a new environment with minimal setup effort and cost. It is therefore not feasible to collect training data across all locations, seasons, and conditions in order to fine-tune the network for each new setting. Our hypothesis for this work was that pre-training on a general dataset containing a mix of appropriate features would reduce the amount of training data needed from the specific use case, and furthermore that generic agricultural features could be obtained by pre-training on available data sets containing general vegetation scenes, e.g. forest trails.

5.1 Diverse agricultural settings

The overall classification accuracies for both the specific agricultural setting (polytunnels) and more general setting (forest trails), presented in Table 1, support our hypothesis that features extracted from general vegetation scenes are applicable for agricultural use cases and appear to be more readily generalisable to diverse settings than those obtained from a specific agricultural setting. However, further research is required to prove or disprove our hypothesis that pre-training on a general dataset reduces the amount and/or diversity of training data required from the particular use case. In our results presented here (Table 1), the same quantity of polytunnel training data (*single polytunnel* or *diverse polytunnels*) was used for both training of the standalone network and fine-tuning of the pre-trained network, and in fact the standalone network performed better on the *polytunnel test* set. This is not surprising, as training and testing on the same setting (although with differences, see section 3.2) can lead to overfitting and inflated performance measures, and perhaps indicates that our *polytunnel test* could have contained more diversity or perhaps that polytunnel environments contain enough visual cues beyond those of the crops themselves that seasonal variations are not as important.

In future work we plan on expanding our research to more diverse agricultural settings with less structure (e.g. Fig. 2). The benefit (or not) of pre-training on a general setting should be more evident from these test cases.

5.2 Autonomous control

Our open-loop field trials reaffirm our conclusions from the offline polytunnel tests: that viewing/steering angle prediction performs well even for seasons and locations not experienced during training. However, we also identified some important limitations to our initial classification approach, which to some extent has been alleviated by changing to a regression output with continuous angle. The precision of the steering angle is not yet satisfactory, and some adjustments of the training setup are required to improve this. Further field testing will be performed to evaluate this simple yaw-angle based control policy. A natural next step could be to consider adding lateral control as in Smolyanskiy et al. (2017), which requires at least an additional camera for the recording setup. Alternatively, one could learn steering commands directly as in Loquercio et al. (2018), but this increases the complexity of recording training data. To ensure safe and reliable operation for such an end-to-end training approach, we will also investigate methods to recognise if the current environment is outside its scope of operation, and present a confidence measure along with the predicted steering commands. For our test case, the robot is driving at a very slow speed (0.4 m/s), and a processing rate of 9 Hz is more than sufficient for closed-loop execution. However, a GPU could be used for DCNN inference to reduce processing time and accommodate faster driving speeds in future applications.

6. CONCLUSIONS

We have presented an approach for predicting steering angles for an autonomous, crop row-following, agri-robot using only RGB image input. Our approach employs a deep convolutional neural network (DCNN) and an end-to-end learning strategy to learn steering angles from images labelled with different viewpoints. We leveraged existing open datasets to pre-train our DCNN with naturalistic features, which improved generalisation capabilities compared to training from scratch on data from a specific agricultural setting. Experiments on existing forest trail datasets and our own datasets from an agricultural setting have demonstrated the accuracy of our approach and its ability to generalise to environments and seasonal conditions not experienced during training. Our online field testing on board an agri-robot operating in a strawberry polytunnel demonstrated the feasibility of this approach for autonomous robot guidance, but also revealed some limitations for steering sensitivity, which will be addressed in future work. Our approach promises a flexible alternative to traditional 2D- and 3D-based onboard guidance schemes and with lower setup costs than external-localisation solutions.

7. FUTURE WORK

Our continuing work will focus on investigating the factors affecting the response of our network to seasonal and environmental variations; investigating the performance of our network on diverse agricultural settings; and implementation on board our autonomous agri-robot for closed-loop field testing.

REFERENCES

Bakken, M., Moore, R., and From, P. (2018). End-to-end learning for autonomous navigation for agricultural robots. *ICRA 2018 Workshop on Robotic Vision and Action in Agriculture*. URL `https://research.qut.edu.au/future-farming/wp-content/uploads/sites/3/2018/06/End-to-end-Learning-for-Autonomous-Navigation-for-Agricultural-Robots.pdf`.

Bakker, T., Wouters, H., Van Asselt, K., Bontsema, J., Tang, L., Müller, J., and van Straten, G. (2008). A vision based row detection system for sugar beet. *Computers and electronics in agriculture*, 60(1), 87–95.

Biber, P., Weiss, U., Dorna, M., and Albert, A. (2012). Navigation system of the autonomous agricultural robot bonirob. In *Workshop on Agricultural Robotics: Enabling Safe, Efficient, and Affordable Robots for Food Production (Collocated with IROS 2012), Vilamoura, Portugal*.

Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., and Fei-Fei, L. (2009). ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*.

English, A., Ross, P., Ball, D., Upcroft, B., and Corke, P. (2015). Learning crop models for vision-based guidance of agricultural robots. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, 1158–1163. IEEE.

Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., and Zisserman, A. (2010). The pascal visual object classes (VOC) challenge. *International Journal of Computer Vision*, 88(2), 303–338.

Giusti, A., Guzzi, J., Cire, D.C., He, F.l., Rodríguez, J.P., Fontana, F., Fässler, M., Forster, C., Schmidhuber, J., Caro, G.D., Scaramuzza, D., and Gambardella, L.M. (2015). A Machine Learning Approach to Visual Perception of Forest Trails for Mobile Robots. *IEEE Robotics and Automation Letters*, PP(99), 1–1. doi:10.1109/LRA.2015.2509024. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7358076`.

Halmetschlager, G., Prankl, J., and Vincze, M. (2014). Probabilistic near infrared and depth based crop line identification. In *Workshop Proceedings of IAS-13 Conference on*, 474–482.

Jiang, G.Q., Zhao, C.J., and Si, Y.S. (2010). A machine vision based crop rows detection for agricultural robots. In *Wavelet Analysis and Pattern Recognition (ICWAPR), 2010 International Conference on*, 114–118. IEEE.

Kaufmann, E., Loquercio, A., Ranftl, R., Dosovitskiy, A., Koltun, V., and Scaramuzza, D. (2018). Deep drone racing: Learning agile flight in dynamic environments. *2nd Conference on Robot Learning (CoRL 2018), Zurich, Switzerland*.

Kise, M., Zhang, Q., and Más, F.R. (2005). A stereovision-based crop row detection method for tractor-automated guidance. *Biosystems engineering*, 90(4), 357–367.

Le, T.D., Ponnambalam, V.R., Gjevestad, J.G.O., and From, P.J. (2019). A low-cost and efficient autonomous row-following robot for food production in polytunnels. *Journal of Field Robotics*, (April), 1–13. doi:10.1002/rob.21878. URL `http://doi.wiley.com/10.1002/rob.21878`.

Loquercio, A., Maqueda, A.I., Del-Blanco, C.R., and Scaramuzza, D. (2018). DroNet: Learning to Fly by Driving. *IEEE Robotics and Automation Letters*, 3(2), 1088–1095. doi:10.1109/LRA.2018.2795643. URL `http://ieeexplore.ieee.org/document/8264734/`.

Perez-Ruiz, M. and Upadhyaya, S.K. (2012). GNSS in precision agricultural operations. In *New Approach of Indoor and Outdoor Localization Systems*. InTech.

Sadeghi, F. and Levine, S. (2016). CAD2RL: Real Single-Image Flight without a Single Real Image. doi:10.15607/RSS.2017.XIII.034. URL `http://arxiv.org/abs/1611.04201`.

Simonyan, K. and Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. *CoRR*, abs/1409.1. doi:10.1016/j.infsof.2008.09.005. URL `http://arxiv.org/abs/1409.1556`.

Smolyanskiy, N., Kamenev, A., Smith, J., and Birchfield, S. (2017). Toward Low-Flying Autonomous MAV Trail Navigation using Deep Neural Networks for Environmental Awareness. *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 4241–4247. doi:10.1109/IROS.2017.8206285. URL `http://arxiv.org/abs/1705.02550`.