Computers and Electronics in Agriculture 212 (2023) 108054

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

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

Original papers

Visual teach and generalise (VTAG)—Exploiting perceptual aliasing for scalable autonomous robotic navigation in horticultural environments

Jonathan Cox [a,*], Nikolaos Tsagkopoulos [a], Zdeněk Rozsypálek [b], Tomáš Krajník [b], Elizabeth Sklar [a], Marc Hanheide [a]

[a] University of Lincoln, Brayford Pool, Lincoln, LN6 7TS, United Kingdom
[b] Czech Technical University in Prague, FEE, Prague, 166 27 Prague 6, Czechia

ARTICLE INFO

Keywords:
Precision agriculture
Robotics
Autonomous navigation
Visual navigation
Deep learning

ABSTRACT

Nowadays, most agricultural robots rely on precise and expensive localisation, typically based on global navigation satellite systems (GNSS) and real-time kinematic (RTK) receivers. Unfortunately, the precision of GNSS localisation significantly decreases in environments where the signal paths between the receiver and the satellites are obstructed. This precision hampers deployments of these robots in, e.g., polytunnels or forests. An attractive alternative to GNSS is vision-based localisation and navigation. However, perceptual aliasing and landmark deficiency, typical for agricultural environments, cause traditional image processing techniques, such as feature matching, to fail. We propose an approach for an affordable pure vision-based navigation system which is not only robust to perceptual aliasing, but it actually exploits the repetitiveness of agricultural environments. Our system extends the classic concept of visual teach and repeat to visual teach and generalise (VTAG). Our teach and generalise method uses a deep learning-based image registration pipeline to register similar images through meaningful generalised representations obtained from different but similar areas. The proposed system uses only a low-cost uncalibrated monocular camera and the robot's wheel odometry to produce heading corrections to traverse crop rows in polytunnels safely. We evaluate this method at our test farm and at a commercial farm on three different robotic platforms where an operator teaches only a single crop row. With all platforms, the method successfully navigates the majority of rows with most interventions required at the end of the rows, where the camera no longer has a view of any repeating landmarks such as poles, crop row tables or rows which have visually different features to that of the taught row. For one robot which was taught one row 25 m long our approach autonomously navigated the robot a total distance of over 3.5 km, reaching a teach-generalisation gain of 140.

1. Introduction

The presence of robots in agriculture is gradually increasing, in recent years a greater number of robotic platforms have become commercially available to automate tasks such as harvesting, plant treatment, phenotyping, yield estimation, weeding, transportation etc. The market value for precision agriculture is increasing significantly from an estimated $3.67 billion in 2016 to $7.29 billion in 2021 and increasing annually at a rate of 14.7% (Oliveira et al., 2021). Robots have the potential to change agriculture entirely as they can assist humans by undertaking repetitive and arduous tasks and replace them in hazardous conditions such as operating under extreme heat. The adoption of robots in agriculture has accelerated further after the impact of the COVID-19 pandemic and the disruption in the availability of seasonal workers (Mitaritonna et al., 2020). Automating these tasks can help resolve some of these problems and bring benefits to farmers with reduced costs and higher yields ensuring that the growing presence of robots in agriculture is not a transient phenomenon.

The ability to autonomously navigate in its environment is essential for a robot to undertake its task. There are many methods used to navigate in autonomous robotics each having its own advantages and limitations depending on the requirements of the application and the available sensors. The most common sensors used for localisation and path traversal are Lidars, global navigation satellite system (GNSS) receivers, cameras (colour, infra-red (IR) and stereo), wheel encoders and inertial measurement units (IMUs).

In agricultural applications, the most commonly used sensors are GNSS receivers together with wheel encoders, however, both of these can introduce errors, wheel odometry suffers from accumulating errors

* Correspondence to: School of Computer Science, University of Lincoln, Brayford Pool, Lincoln, LN6 7TS, United Kingdom.
E-mail address: jcox@lincoln.ac.uk (J. Cox).

https://doi.org/10.1016/j.compag.2023.108054
Received 12 April 2023; Received in revised form 4 July 2023; Accepted 7 July 2023
Available online 1 August 2023

Fig. 1. The three robots used in our experiments, (a) Saga Thorvald in its Tall Arch configuration, (b) Clearpath Husky and (c) Agilex Hunter 2.0. The three robots are equipped with an Intel Realsense D435i camera mounted at different locations on each robot, on the Thorvald it is mounted at the top of the arch at a height of 1.95 m, on the Husky it is mounted at 0.5 m above the ground and on the Hunter 2.0 the camera was mounted at two heights 0.35 m and 0.8 m.

over time, especially in environments where the robot's wheels are prone to skid and slip, while the accuracy of civil-grade GNSS is anywhere from 3 m to 30 m (Gao et al., 2018). The accuracy of GNSS can be improved with the use of real-time kinematic (RTK) GNSS to subcentimetre accuracy (Nørremark et al., 2008), however, this accuracy comes with a high cost of receivers and requires the availability of nearby base station or the cost of a RTK data service and a method of sending the RTK correction data to the mobile robot by WiFi or mobile data.

The use of RTK GNSS is suitable for some agricultural robot applications, e.g. outside in open fields, however in other applications the robot has to operate in covered areas such as around buildings, metal polytunnels, tree canopies and drive through dense large crops which can attenuate or reflect the signal (Perez-Ruiz and Upadhyaya, 2012; Guo et al., 2018; Opiyo et al., 2021). Navigation inside polytunnels requires high precision localisation due to the confined spaces between the crop rows making traversing a large robot difficult, as can be seen in Figs. 1 and 5. Therefore the need for alternative navigation methods originates from the requirement for robots to operate in GNSS denied environments.

Navigation in GNSS denied environments can be achieved with onboard sensors such as laser range scanners and cameras in combination with Simultaneous Localisation and Mapping (SLAM) (Durrant-Whyte and Bailey, 2006; Bailey and Durrant-Whyte, 2006), visual SLAM (Taketomi et al., 2017) or Monte Carlo Localisation (MCL) (Dellaert et al., 1999). While these methods are suitable for localisation and mapping, they typically require sensors such as Lidar scanners, colour depth (RGB-D) cameras and stereo cameras, which are more expensive than monocular cameras. In addition, SLAM algorithms do not perform well in repetitive environments and are prone to failures caused by environmental changes (Cadena et al., 2016). An alternative approach to navigation, based on the teach-and-repeat paradigm (Furgale and Barfoot, 2010b) has shown to be able to perform well in changing environments using only an off-the-shelf monocular camera (Krajník et al., 2010). The robustness of teach-and-repeat systems to environmental changes was confirmed by extensive field tests (Paton et al., 2017).

Our approach is based on the idea of navigation without localisation in a metric map, where the robot only travels a known path from a start point to a goal. This has the benefit of not requiring a large metric map that is not scalable to commercial agricultural settings. In this paper, our aim is to extend this method to where the robot can generalise and travel a trajectory along untaught paths similar to the mapped one.

Our teach and generalise method takes advantage of scenes with high visual aliasing or visual repeatability and can be applied to a wide range of applications that have repetitive environments, such as corridors in offices and hospitals, isles in factories, supermarkets and warehouses, agricultural polytunnels, crop fields, vineyards etc. Our method has the advantage of only using low-cost monocular cameras over traditional navigation methods that use expensive Lidars, RTK GPS receivers and RGB-D cameras, leading to more affordable robotic platforms, making them more accessible to a broader range of users and scalable to new applications that require fleets of robots, such as logistics (Ravikanna et al., 2021). Finally, our method can significantly reduce the man-hours spent on creating a precise metric map for the whole environment where the robot is deployed.

We test our approach across three different robotic platforms and in multiple agricultural polytunnels, one at a test farm site and several others on a commercial farm. The robots are taught trajectories along a row and use the method to repeat the path along new, previously unseen rows. We evaluate the method by recording the number and locations of any interventions that may be required and the distance the robot can autonomously travel based on the distance taught, a generalisation gain. Our focus is on in-row navigation where our approach can navigate along the rows and not at the end of rows where other navigation methods can be used.

1.1. Related work

Within polytunnel environments, most path traversal methods use Lidars to detect the rows and poles and move the robot along the centre of the rows (Le et al., 2020; Xiong et al., 2020; Ponnambalam et al., 2020). The interest in vision based navigation has increased in recent studies to complement or overcome the shortcomings and expense of GNSS and Lidar based navigation (Aguiar et al., 2020).

1.1.1. Visual crop row following

Monocular cameras have been widely employed for crop row following in open fields as the crops can provide sufficient colour information and texture, RGB-D and stereo cameras can utilise additional information from the scene structure. The choice of the sensor depends on the application as the environment influences the decision. RGB-D and stereo cameras can complement traditional 2D segmentation methods which usually perform poorly under inconsistent lighting conditions, but they are preferred in applications where crops are large enough to be distinguished from the soil (English et al., 2014). RGB-D and stereo cameras find wide applicability in between-row navigation such as in orchards, vineyards or maize fields as those crops are large enough to provide adequate information (Aghi et al., 2021; Peng et al., 2022; Fei and Vougioukas, 2022; Cerrato et al., 2021; Luo et al., 2022) and in environments with the absence of crops where the camera detects the ridges and furrows of the crop rows (Song et al., 2022).

Vision based navigation in crop rows needs to distinguish the crops from the background this can be achieved by using segmentation and contour based methods (Guerrero et al., 2013; Romeo et al., 2012), classical computer vision techniques such as edge detection and colour segmentation (Luo et al., 2022; Zhou et al., 2021; Chang et al., 2022; Li et al., 2022) vegetation indexes using colour (García-Santillán et al.,

2017; Ahmadi et al., 2020, 2021) and IR cameras (García-Santillán et al., 2017; Åstrand and Baerveldt, 2005) or with deep learning networks (Lin and Chen, 2019; Chen et al., 2021; He et al., 2022; Bah et al., 2019; Adhikari et al., 2020; de Silva et al., 2022). Colour based approaches perform poorly in the presence of weeds and rely on manually tuned hyperparameters such as thresholding values, while deep learning methods are not crop agnostic and they require large amounts of labelled data which can be laborious to collect and not readily available.

After extracting suitable regions of interest, a common method is to trace a line on top of the identified crops (or between the crops in multi-row applications). This can be done either by extracting centroids from the contours of the segmented image and then performing standard regression fitting (Ahmadi et al., 2021; Ma et al., 2021) or by applying a Hough Transformation (Li et al., 2022; Winterhalter et al., 2018). Fitting arbitrary lines on extracted crop centroids has the advantage of navigating on curved paths (García-Santillán et al., 2017). Finally, the calculated angle and position of the extracted line in the image is used by a controller to send navigation commands to the robot.

1.1.2. Visual teach and repeat

Teach and repeat is a learning from demonstration technique similar to that used to programme stationary industrial robots to perform repetitive tasks. It is a well established concept in robotics, but visual teach and repeat (VT&R) and specifically in the context of navigation, is a keyframe based technique where the robot tries to localise and navigate based on past experiences using only a camera and odometry. As its name suggests it consists of two phases, the "teach" phase and the "repeat" phase. During the teaching phase an operator teleoperates the robot along a desired trajectory and the robot creates a topological map by saving the current image and wheel odometry at predefined distances as a graph of vertices and edges.

VT&R methods can be classified into pose based and appearance based approaches. Pose based approaches (Barfoot et al., 2012; Furgale and Barfoot, 2010a; Courbon et al., 2009; Ostafew et al., 2013; McManus et al., 2012; Clement et al., 2017) relies on relative pose estimation between the current and stored images this is based on matched features between these images. During the repeat phase, the system compares each frame from the live camera images with images stored in the vertices and localises itself by computing its current pose with each stored pose. After finding the closest vertex, it tries to move there by minimising the transformation between its current pose and the vertex's pose. Then it proceeds to the next vertex. During initialisation all the reference images are used in order to find the initial position of the robot, these approaches are computationally expensive.

Appearance based approaches first introduced by Chen and Birchfield (2009) and later by Krajník et al. (2010) use the same concept of a graph like map but without performing explicit localisation, it relies on comparing landmarks in the images captured with a monocular uncalibrated camera and wheel odometry. During the teaching phase, the operator drives the robot along the desired path and at fixed distance intervals a vertex is inserted into the map and stores the image and the distance travelled. During the repeat phase, the robot starts at approximately the same place with a suitable orientation to allow it to detect landmarks located in the first vertex, it starts navigating by replaying the velocity commands captured during the teaching phase and stops only when the total travelled distance matches the recorded one. As there is uncertainty about the robot's initial position and errors such as wheel slippage, reaching the end of the path by relying only on dead reckoning is insufficient. To reach the end of the path heading corrections are applied by using techniques such as feature matching the live camera images to the images stored in the map (Krajník et al., 2010; Chen and Birchfield, 2006; Churchill and Vardy, 2012; Erhard et al., 2009; Vardy, 2010; Majdik et al., 2013; Krajník et al., 2018, 2017), resulting in bearing navigation which optimises the heading or bearing of the robot. This method has benefits in computational cost, scalability in terms of distance and ease of implementation.

In recent years, a large part of computer vision research has been dominated by deep learning. It was shown that the handcrafted feature extractors could be outperformed by the learned ones (Krizhevsky et al., 2012). This development also has a significant impact on the field of visual navigation. Many modifications that exploit the advantages of machine learning were suggested to improve the robustness of VT&R (Camara et al., 2020; Broughton et al., 2021; Gridseth and Barfoot, 2022). One of the major issues for navigation is the deficiency of matching features caused by the environmental changes between the mapping and teaching phases. This issue is even more significant in agriculture, where the crops are growing, and the appearance of the environment can change rapidly. Multiple different approaches are used to tackle the changes in the environment. One way is to do continual mapping of the environment so that the map currently used for the navigation is not obsolete (Dayoub and Duckett, 2008; Churchill and Newman, 2013). Another way would be to learn what changes can happen from the long-term datasets and then use the learned model as an image descriptor, invariant to specific environmental changes (Krajník et al., 2017).

Some assumptions can be applied to the mobile ground robot for the VT&R framework, such as the robot is constrained to 2DoF as it can only move linearly forwards, backwards and rotate. Many of these frameworks use feature extractors but they do not use the matched features to estimate the 6DoF in the environment. The information usually obtained from the matched features is their displacement in the horizontal axis and the robot is then controlled to minimise this displacement. It was shown that it is possible to use the cross-correlation of the whole images to obtain similar information (Dall'Osto et al., 2021). Later, a fully-convolutional neural network was applied to improve the quality of cross-correlation (Rozsypálek et al., 2022b). The neural network is learned via contrastive methods and can produce a dense representation of the image, which is robust to seasonal and day/night image variations. Another advantage of this method is that it can be learned in a self-supervised fashion and is relatively efficient in terms of the needed amount of data. Additionally, the network is small, which is favourable for usage on a mobile robot, where real-time usage is crucial, and the performance of onboard computers is modest.

2. Method

Our new visual teach and generalise (VTAG) method adopts and pushes the work of the Bearnav framework, which is a complete VT&R navigation framework fully integrated in ROS (Krajník et al., 2018). Similarly, as with other VT&R frameworks, the deployment consists of two steps — teach and repeat. During the teaching phase, the operator drives the robot, and the framework constructs a map by saving reference images at predefined distance integrals by recording the camera and odometry data. Additionally, it records the velocity command data. During the repeat phase, the robot repeats the taught trajectory by replaying the pre-recorded velocity command, camera and odometry data and applies heading corrections based on the live camera feed.

2.1. Topological navigation framework

At our test farm, we can make use of a topological map of the site which comprises of nodes and edges connecting the nodes, as illustrated in Fig. 5(c). The nodes within this map correspond to predefined goal points, while the edges dictate the permissible paths and movement constraints for the robots. The edges in the map are associated with specific ROS actions, such as move_base for areas outside the tunnels and VTAG for regions within the tunnels. The utilisation of ROS actions enables transitioning between these two navigation methods, as depicted in Fig. 7 where the navigation mode shifts from VTAG to

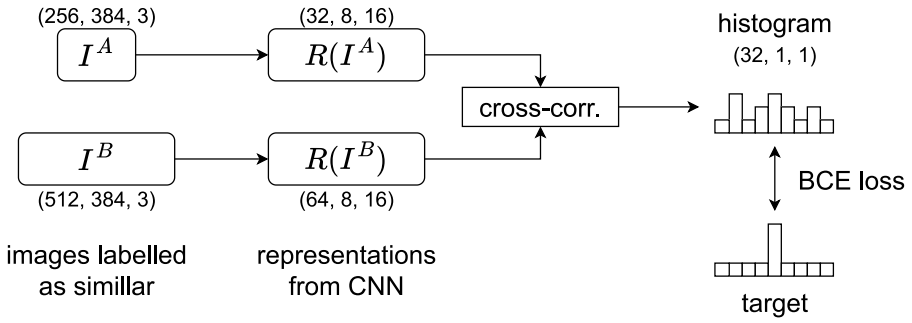

Fig. 2. Diagram depicting the training of the fully-convolutional Siamese network. The input are two different images labelled as similar. One of the images is cropped in width and both images are passed through CNN denoted R. The output of the CNN are the neural representations, which are further cross-correlated. Final histogram is used to compute the loss and the target is constructed on the fly, based on the position of the crop.

move_base at the end of the rows. It is important to highlight that this paper focuses on addressing in-row navigation and not traversing between rows or outside the polytunnels. A more detailed description of topological maps and analysis of such a robotic system for supporting farms and workers can be found in Zhu et al. (2023).

2.2. Teach and generalise

The VT&R frameworks generally cannot traverse different trajectories than the one traversed during the teaching phase. However, it can be highly tedious to create maps covering the whole environment, especially in agriculture, as autonomous robots are often deployed to cover vast areas. In this paper, we turn this challenge into an opportunity by aiming to teach only a small part of the site where the robot is deployed and use a single map to traverse the rest. This is possible because agricultural environments are usually very repetitive, and they often share visual cues, which can be followed to perform a successful traversal. In the past, such a visual cue was picked by humans, and the feature extractor was handcrafted. Our method aims to remove a human from the loop as much as possible and create a unified framework which is able to learn the most suitable features together with their extractor and descriptor.

Our method is an extension of Bearnav (Krajník et al., 2018; Rozsypálek et al., 2022a) which is based on a fully-convolutional self-supervised Siamese network (Bromley et al., 1993), which is used for the estimation of the image alignment during the repeat phase, a diagram of the architecture is shown in Fig. 2. We show that the training process can be altered so that the network is able to focus on the features specific to certain types of trajectory and environment, and thus apply the heading corrections on previously unseen paths. We exploit the capability of contrastive learning to automatically extract the most suitable features, without the aid of manually selecting features, based on the data feed in the training process.

Contrastive learning, in general, can update the parameters of the model, based on which data points are labelled as similar or dissimilar. The labelling is absolutely crucial and determines what kind of features are extracted by the model. Originally the fully-convolutional Siamese networks were used for image tracking — the part of the image containing the object in the consecutive video frames was labelled as similar, and the parts of the images not containing the object were labelled as dissimilar (Bertinetto et al., 2016). In this setup, the final extractor was robust to object variations in the image (rotation, partial occlusion, etc.). Further, it was shown that a similar method can be used to obtain image features robust to seasonal and day/night variations (Germain et al., 2019). More recent research showed that this approach is also suitable for deployment in robotics (Rozsypálek et al., 2022a). This particular model pre-trained on a large dataset is used as a starting point in our training pipeline.

Fig. 3 shows the CNN architecture, which is denoted as a function R in our Siamese architecture and is used to obtain a neural representation of the images. The training pipeline of the original model required a large number of images taken from the same position at different times (or environment state). All images which are taken at the same place with similar headings are then labelled as similar. Further, pair of images labelled as similar is loaded into the pipeline, and one of them is cropped. Both images are then passed through the same CNN, and their representation is obtained. Representation of the cropped image is significantly smaller and is used as a cross-correlation kernel. Finally, we know the exact location of the crop, so it is possible to create a target for the binary cross-entropy loss. This target has zero values at the positions where the crop is not located and non-zero values at the true crop location. In this paper, the training pipeline is alike, and the shape of the target can be seen in Fig. 4.

We push the framework presented in Rozsypálek et al. (2022a) even further, and we relax the constraint on the similarity of the position from which the image was taken. Our only requirement is that the image pairs are labelled as similar when they contain a visual cue on a consistent position. At first, this seems like a strong requirement. However, in repetitive agricultural environments, this can be ensured easily, for example by keeping the robot at a consistent distance from the crop row or polytunnel row structures during the data collection. The resulting model will learn to ignore the variances present in the dataset (i.e. plant details, background) and to focus on similarities (i.e. groups of plants, tracks in the dirt, consistent man-made objects such as polytunnel poles). It is necessary to keep in mind that by labelling images from different positions as similar, the neural network can lose some abilities presented in navigation stacks exploiting similar approach (Rozsypálek et al., 2023). For example, when one long crop row is traversed back and forth, and all the gathered images are labelled as potentially positive image pairs, the network will inevitably lose the ability to estimate the exact position in the row from the camera feed. Still, it can gain the ability to generalise and output valid heading corrections to traverse a different row.

2.3. Experimental setup

The experiments were conducted in a test farm at the Riseholme Campus of the University of Lincoln, UK and three polytunnel fields in a commercial farm in Norfolk, UK. The test farm consists of two 25 m long polytunnels each with five rows of strawberry plants mounted on tabletops (numbered r1 to r10) and six paths between these rows where the robots can traverse along (numbered r0.7, r1.5, r2.5, r3.5, r4.5, r5.3, r5.7, r6.5, r7.5, r8.5, r9.5, r10.3), the polytunnel and row numbers are shown in Fig. 5. The first commercial farm consists of twenty-two 120 m long polytunnels each split into two halves with seven 58 m rows where the robots can traverse. The second and third consist of rows of 65 m and 90 m long. Multiple traversals of the polytunnels were

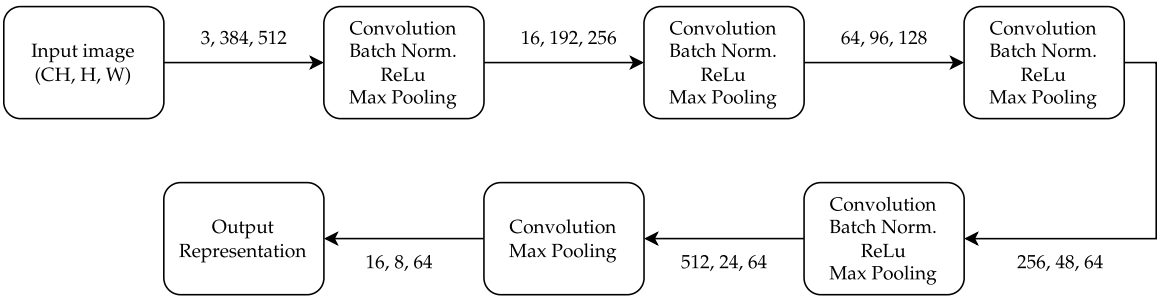

Fig. 3. Architecture of the convolutional neural network R used as the backbone of the Siamese architecture, which creates the embedding from the images captured by the camera.

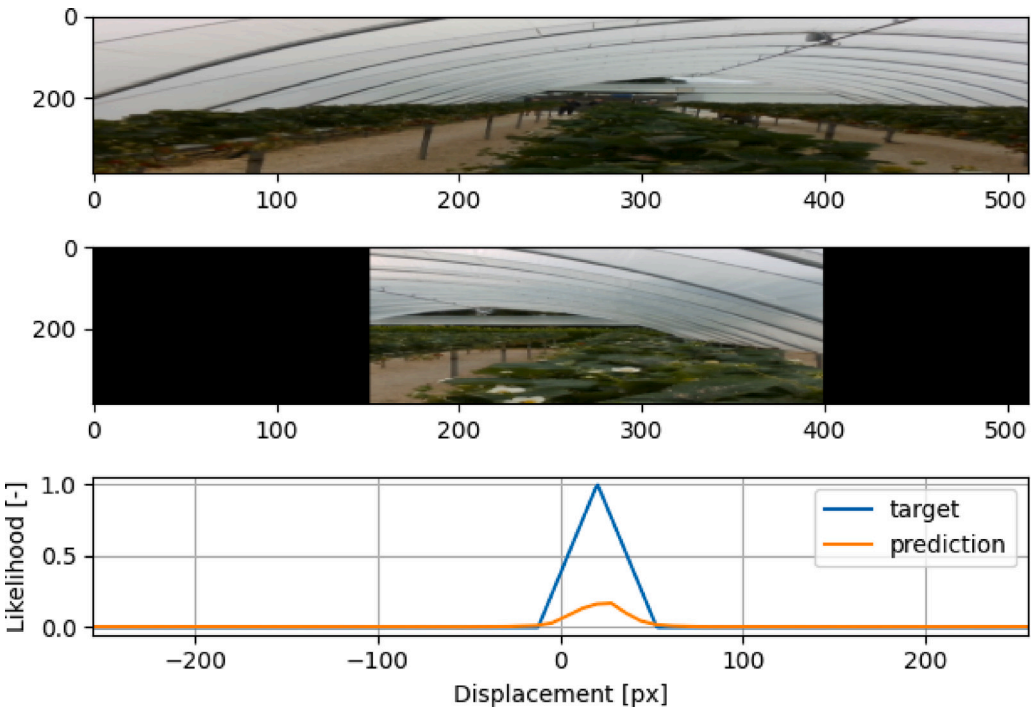

Fig. 4. An example of a data sample from the training process. At the top, we see the image captured by the robot during the traversal of a polytunnel. In the middle is a crop of the image, which was captured while traversing a different row in the same polytunnel. The crop position is chosen randomly on the fly during the training, and the target (visible at the bottom) matches the position of the crop. The proxy task for the network is to estimate the position of the crop given the top image. Even though the images are from different rows, they follow similar patterns (crop row, poles) the network can learn from.

performed, and the data gathered during these traversals are labelled as similar images for the training pipeline of the Siamese neural network. The resulting model is then used in the standard VT&R setup with one major difference — only one map (from a single polytunnel row) is used to traverse all the polytunnel rows.

To test the generalisability of the navigation system three robots were used for the experiments, a Saga Thorvald in its tall arch configuration, a Clearpath Husky and an AgileX Hunter 2.0, see Fig. 1. These represent a range of different ground robot types, from the Throvald with 4-wheel drive and 4-wheel steering, the Husky with 4-wheel differential drive and the Hunter 2.0 with 2-wheel drive Ackerman steering. All three robots are equipped with an IMU and an Intel Realsense D435i using only the RGB image stream with a resolution of 640 × 480 pixels. The cameras are mounted at different locations on the robots, on the Thorvald it is mounted at the top at a height of 1.95 m with a view looking over the top of the crop rows, on the Husky and Hunter 2.0 the cameras are mounted on the front of the robots at 0.5 m and 0.35 m from the ground respectively, this again tests the generalisability of the navigation system to different viewpoints in the environment, the camera views from the Thorvald and Hunter 2.0 are shown in Fig. 6. The Thorvald can only drive along the ten table top rows due to its tall arch design as it drives over the crop rows rather than in between the rows as the Husky and Hunter 2.0 can. Unlike the other robots, the Thorvald platform can take advantage of the RTK GNSS because the plants do not occlude the signal, and thus it is possible to use the RTK GNSS as ground truth for this robot.

In our study we refer to training as the data training of the Siamese network, teaching as driving the robot along a new path to create a map and repeating as the robot driving along a taught path. The Siamese network and feature extractor were trained only once on the Thorvald with the camera at a height of 1.95 m along all the rows at the test farm in Riseholme at midday (Fig. 6(a)). This network was used on all three robots at all the polytunnel sites without being retrained.

For the teaching phase, all three robots drove along a row in the polytunnels at a constant speed. After creating a map from a row the robot is guided to each row's start point, once at the start of the row VTAG's repeat mode is started to traverse the row. During the traversals, each frame from the camera's live feed is compared to the closest image in the map, based on the current odometry, to estimate the horizontal displacement and to correct the heading, by sending velocity commands to the robot. The robot configuration and VTAG parameters are shown in Table 1. Thanks to the small size of the neural

Table 1
The different configurations of the three robotic platforms and the VTAG parameters used in the experiments, node interval is the distance between nodes and recorded images in the map created by VTAG. The cameras used are Realsesne D435i but only the RGB images were enabled.

Parameter	Thorvald	Husky	Hunter 2.0
Steering Configuration	4 Wheel Steer	Skid Steer	2 Wheel Ackerman
Camera type	Intel RS D435i	Intel RS D435i	Intel RS D435i
Camera Height, [m]	1.95	0.5	0.35 & 0.80
Camera Resolution, [px]	640 × 480	640 × 480	640 × 480
Computer	Intel NUC 8	Asus PN50-E1	Intel NUC 11
CPU	Intel 8th Gen i5	AMD Ryzen 4700u	Intel 11th Gen i7
RAM	16 GB	8 GB	16 GB
Robot Size, L × W × H, [m]	2.00 × 1.50 × 2.00	0.99 × 0.67 × 0.70	0.98 × 0.75 × 0.38
Teach-Repeat speed, [m/s]	0.2	0.4	0.4
Node interval, [m]	0.7	0.7	0.7

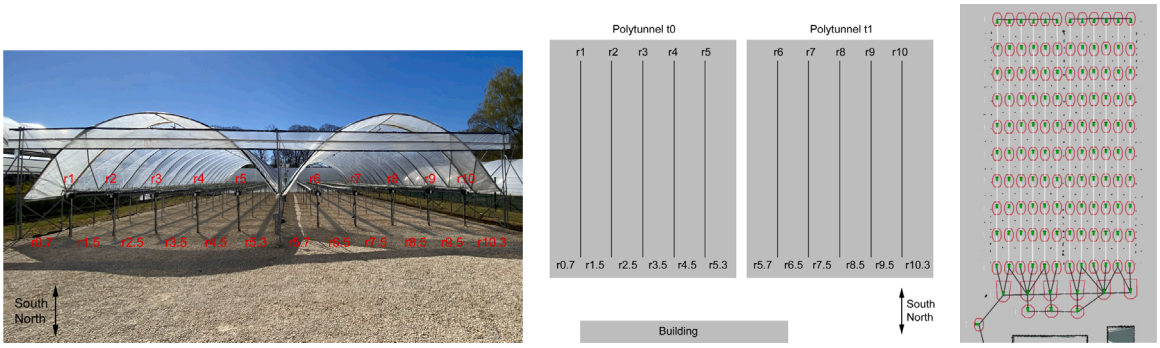

Fig. 5. The test site strawberry polytunnel at the University of Lincoln. (a) shows the two polytunnels and the row numbering system and orientation, the rows are 1.4 m wide (between the poles) and 25 m long. (b) a diagram of the top down view of the polytunnels. There are ten table top crop rows numbered r1 to r10 the rows in between the table top rows are numbered r0.7, r1.5, r2.5, r3.5, r4.5, r5.3, r5.7, r6.5, r7.5, r8.5, r9.5 and r10.3. (c) shows the topological map of the polytunnels showing the nodes and edges the robots can traverse when using topological navigation.

Fig. 6. Camera views from the robots at the test farm, the network was trained on images from the Thorvald (camera height 1.95 m) (a), VTAG was used on the Hunter 2.0 (camera height 0.8 m) (b) shows the taught row and (c) one of the repeated rows, which is visually different in position in the polytunnel, pole positions, roof height and background scene.

network, it is possible to achieve relatively high speed even though we do not employ GPU for image processing. With the Intel NUC 8th gen the processing speed is approximately 10FPS, and from our experience with VT&R navigation, it is necessary to process at least 3–5 images per metre, which puts the upper bound to the maximal speed of the robot. We were able to navigate the Husky robot with speeds over 1 m/s, but the space in the polytunnels is very tight, so, for safety reasons, we used lower speeds in the experiments.

3. Results

Our aim is to investigate the generalisability of the VTAG navigation framework in polytunnel environments. To evaluate this, we record the locations where human intervention is required, an intervention is where the operator intervened manually, with a joystick, to prevent the robot from hitting obstacles or where the robot drastically deviates from the desired trajectory. After an intervention, the robot was manually manoeuvred back to the centre of the row and VTAG resumed, occasionally the robot was driven back to the start of the row, which is reflected in the autonomous distance figures. As a total measure of this, we calculate the mean distance between interventions (MDBI), i.e. the total distance autonomously travelled divided by the number of interventions. Note that there is a minimal margin for error inside the polytunnel, and even a relatively small deviation in the lateral axis (~20 cm) can require an intervention. We also calculate a generalisation gain (or the teach/repeat ratio), the distance the robots travel using VTAG over the distance the robot was taught, in the case where the whole environment is mapped the gain would be 1. Another measure is the absolute trajectory error (ATE), which was only recorded

Table 2
Experiment results of our VTAG method used on three different robotic platforms (Thorvald, Husky and Hunter 2.0) in both our test farm and commercial farm sites.

Farm	Robot	Camera height [m]	Distance [m]		Interventions		MDBI [m]		Generalisation gain
			teach	traverse	end-row	in-row	end-row	in-row	
Test	Thorvald	1.95	25	418	20	0	21	n/a	17
Test	Husky	0.50	25	3,549	25	1	142	3,549	142
Test	Hunter 2.0	0.35	50	1,035	18	10	58	103	21
Test	Hunter 2.0	0.80	50	2,339	32	3	73	780	47
Commercial	Husky	0.50	116	2,146	0	0	n/a	n/a	19
Commercial	Hunter 2.0	0.80	606	6,016	9	20	668	301	10

on the Thorvald, where the repeated trajectory is compared to ground truth,

$$ATE_{RMSE}(\hat{X}, X) = \sqrt{\frac{1}{N}\|trans(\hat{x}_n) - trans(x_n)\|^2} \quad (1)$$

where $trans(x_n)$ is an individual 2D ground truth point corresponding to the crop row and $trans(\hat{x}_n)$ a 2D point of the demonstrated trajectory. The GNSS coordinates of the rows were used as ground truth or reference trajectory points, while the demonstrated trajectory was sampled every 0.2 m, using coordinate data from the RTK GNSS data. The experimental results for all three robots and farm sites are summarised in Table 2.

3.1. Thorvald

The Thorvald robot was taught to traverse the centre of row r8 north to south in our test farm polytunnel around midday and repeated the trajectory on all ten rows in both directions using VTAG framework the same afternoon. Fig. 7 summarises the results of the lateral, absolute trajectory and displacement errors for all ten rows repeated in both directions. The Thorvald is equipped with an RTK GPS receiver allowing a ground truth to be collected along the centre of each row traversed and compared against. The average lateral and absolute trajectory errors show an oscillation around the row with a mean error of 5–6 cm, while the robot did not fail to traverse a single row. The vision system was disengaging and switching back to move_base in the last 2.8 m as the table top crop row was no longer visible in the camera view. Fig. 7(c) shows the neural network's certainty of the output displacement estimations averaged over all the traversed rows, for the larger part of the row, the network is very confident, while towards the end of the row, the confidence drops significantly. This is mainly due to the decrease in the relevant features in the scene as the crop rows become less visible. The Thorvald travelled a total of 418 m with no in-row interventions. The only interventions were at the end of each row where the Timed Elastic Band (TEB) planner took over from VTAG given that the rows are only 25 m long VTAG successfully navigated the robot for 87% of all the row lengths.

3.2. Husky

The Husky robot was used at both our test farm and at a commercial farm. At our test farm, it was taught to traverse the centre of row r4.5 north to south around 11 am–12 pm. All twelve of the rows were repeated in both directions, a total of 144 row lengths (3.6 km) were repeated using this taught trajectory of a single 25 m row, in the results around two thirds of the repeat rows were conducted on the same day as the teaching, 12 pm–4 pm, and around one third was conducted three days later, 11 am–3 pm, using the same taught trajectory. Fig. 8(a) shows the locations of interventions in the polytunnel rows during the repeat phase. There were no interventions at the start of the repeat runs, almost all of the interventions are within 1–2 m of the end of the repeat runs at the end of the rows. We consider the end-of-row at 2.5 m from the end, shown as horizontal lines in Fig. 8. This is where in the camera view the distance between the last polytunnel poles is half the width of the image and the view is of the outside of the polytunnel with little view of the table top crop rows or the upright poles. One location requiring the intervention is halfway along row r10.3 this most likely because the visual features are different in the end rows than the middle rows, the end rows have a slightly different arrangement of poles and crop row tables which may have caused the neural network to misalign the images on this occasion. The Husky travelled a total of 3,548.5 m with 1 in-row intervention resulting in an in-row MDBI of 3,548.5 m and 20 interventions at the end-of-rows.

In the commercial farm polytunnels the Husky was taught to traverse the centre of one 58 m row (1.55 m wide) in both directions, around 10 am–11 am. The robot was then commanded to repeat 18 rows in both directions (one row twice) the same day from 11 am–4 pm. The Husky travelled a total of 37 row lengths, a distance of 2,146 m, successfully traversing all the rows without any intervention.

3.3. Hunter 2.0

The Hunter 2.0 robot was tested at both our test site and at the commercial farm. Firstly at our test site, the camera height was mounted at a height of 0.35 m and the robot was taught to traverse the centre of row r2.5 (Fig. 6(b)) both north and south in our test farm polytunnel, from 11 am–12 am. All twelve of the rows were repeated, the same day from 12 pm–4 pm in both directions using the respective north or south taught path a total of 48 row lengths. As with the Husky the majority of intervention locations during its repeat phase are near the end of the rows, as shown in Fig. 8(b). Some of the locations are further inside the rows (1–5 m) compared to the Husky and there are a few locations requiring intervention at the start of rows this is likely due to the lower camera position on the Hunter 2.0 where less of the table top rows are in the camera view. There are several locations in the end rows, r0.7, r5.3, r5.7 and r10.3, parts of these rows are visually different from the middle rows where the robot was taught. The Hunter 2.0 travelled a total of 1,034.5 m with 10 in-row interventions resulting in an in-row MDBI of 103.45 m and 18 interventions at the end-of-rows.

To investigate if the low camera position was causing interventions we raised the camera height to 0.8 m on the Hunter 2.0. In this configuration, the robot was taught r3.5 both north and south in our test site, from 10 am–11 am. All twelve of the rows were repeated twice the same day from 11 am–4 pm in both directions using the respective north or south taught path a total of 96 row lengths. As can be seen in Fig. 8(c) most of the interventions are at the ends of the row but there are fewer interventions in the rows, three interventions whereas eleven with the lower camera position. The Hunter 2.0 with the higher camera position travelled a total of 2,338.5 m with 3 in-row interventions resulting in an in-row MDBI of 779.5 m and 32 interventions at the end-of-rows, a large improvement over the lower camera height.

In the commercial farm polytunnels the Hunter was tested in three different polytunnel environments with row lengths from 58 m to 90 m. In all the polytunnels there are two row widths 1.55 m and 1.05 m the robot was taught to traverse both row types once in each polytunnel, around 9 am–11 am. Across all the polytunnels the robot repeated, the

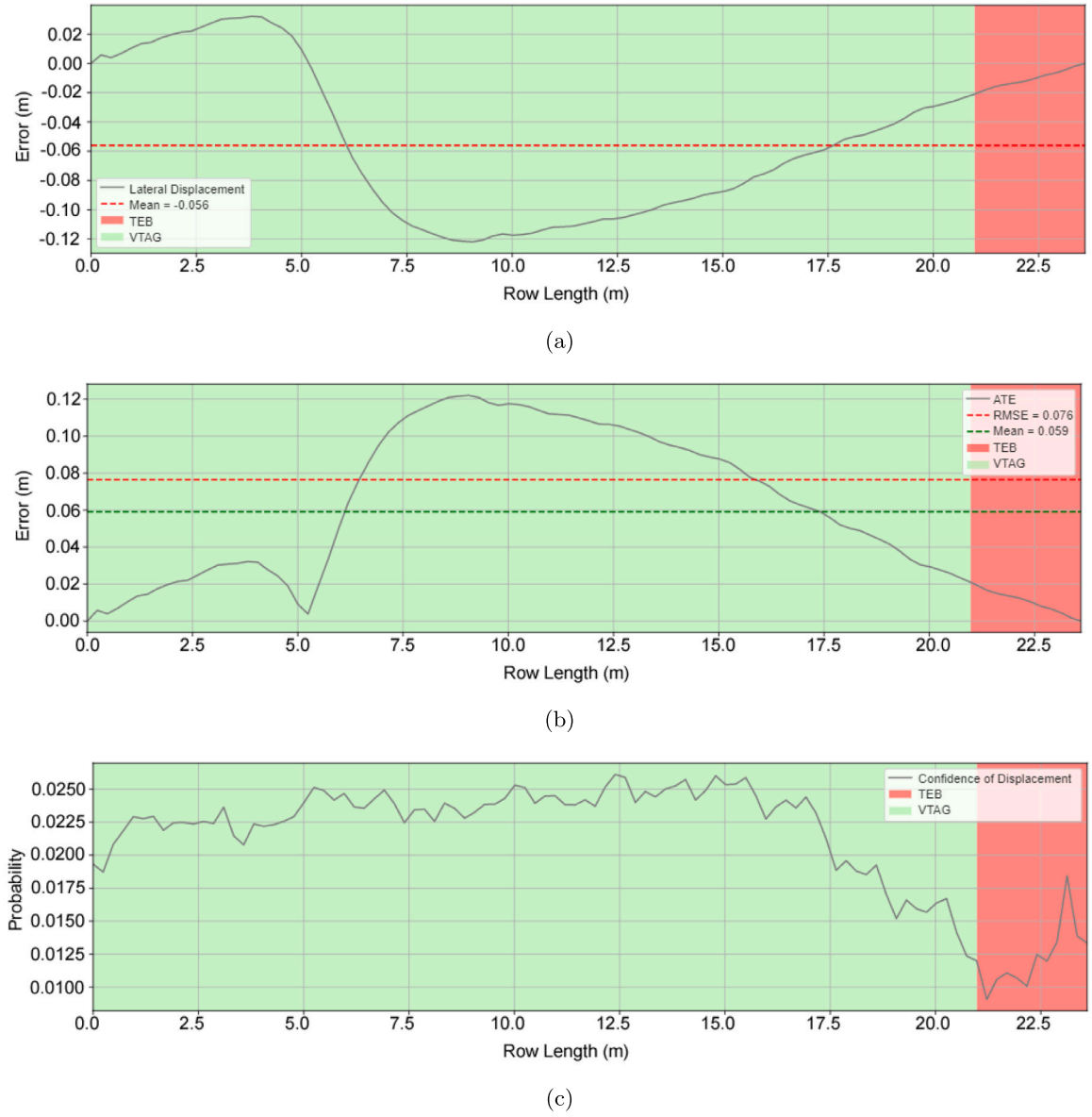

Fig. 7. The average lateral error (a), average absolute trajectory error (b) and average displacement confidence (c) averaged over all ten rows repeated in both directions using the Thorvald. Green highlights navigation with VTAG and red where navigation is taken over by move_base and the Timed Elastic Band (TEB) planner.

same day from 10 am–4 pm a total of 94 rows and successfully travelled 6,016 m with 20 in-row interventions resulting in an in-row MDBI of 300.8 m and 9 interventions at the end-of-rows.

4. Discussion and conclusion

We extend the classic visual teach and repeat to a novel teach and generalise approach for autonomous path traversal, where the robot learns how to traverse a path and is able to generalise on unseen paths. This approach only uses a single uncalibrated monocular camera and the robot's odometry, it has been tested in a strawberry farm polytunnel but it can be used in any visually repeating environment, such as corridors in offices and hospitals, isles in factories, supermarkets and warehouses, crop fields, vineyards etc.

The method is based on path repeatability convergent property introduced in Krajník et al. (2010) by applying only heading corrections. Visual displacement estimation can be challenging in agricultural environments due to the lack of distinct landmarks and the perceptual aliasing introduced by the scene's repeatability. This perceptual aliasing is turned in our favour by learning general scene representations using the self-supervised framework introduced in Rozsypálek et al. (2022a).

Our experimental results show our approach can successfully traverse along unseen polytunnel rows from only one taught row both in our test farm and in a commercial farm. Our approach is robust and generalisable across rows and between different robots with different steering methods and camera positions. Almost all of the locations where our approach requires intervention are at the end of the repeat run at the end of the polytunnel rows this is because the camera no longer has a view of the regular repeating structures such as the poles and table tops ahead of itself and now has a view of the open areas outside the polytunnels. As our experiments only taught the trajectories inside the rows this is to be expected as our approach takes advantage of the crop row structure to estimate displacement errors. At our test farm, a few locations requiring intervention are in the end rows of the polytunnels (r0.7 and r10.3, see Fig. 8), in places these rows have visual features which are different to the middle rows where the robots were taught, the end rows have a slightly different arrangement and number of poles and crop row tables. To improve the approach in our experiment new trajectories may need to be taught, one of a middle row

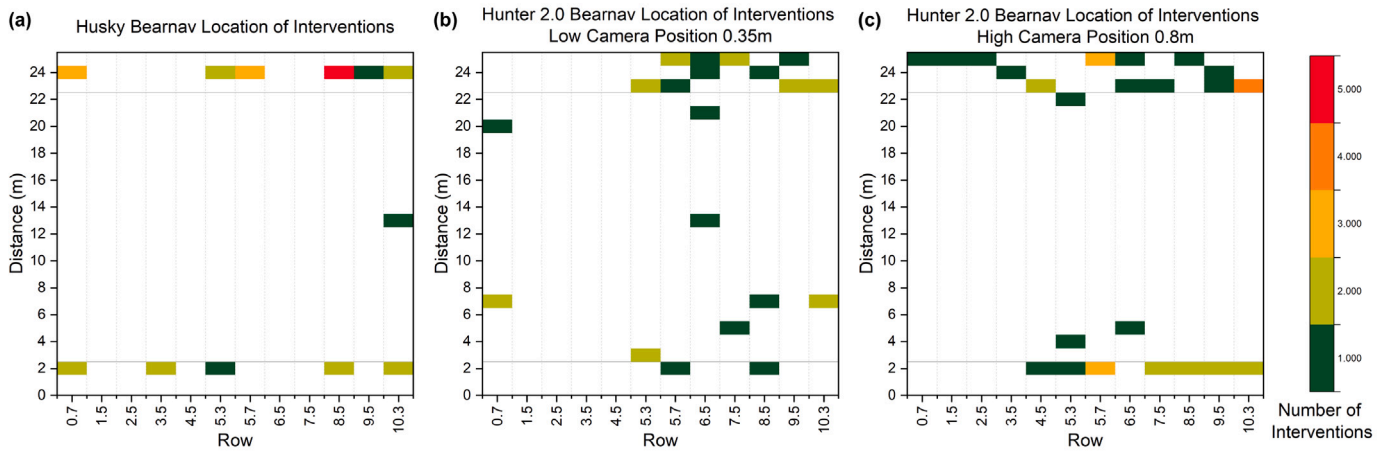

Fig. 8. Locations, within 1 m intervals, and frequency of interventions during the repeat phase in both directions using the (a) Husky and Hunter 2.0 in the polytunnels, (b) shows the results from the Hunter 2.0 with other lower camera position at 0.35 m above the ground and (c) with the camera at the higher position of 0.8 m. The horizontal lines at 2.5 m and 22.5 m shows the end-of-row this is where in the camera view the distance between the last polytunnel poles is half the width of the image, with a view outside the rows and little view of the repeating table tops and poles. Row numbering and orientation are the same as shown in Fig. 5.

and one of an end row and repeat the appropriate trajectory dependent on which row the robot is travelling along. It is also possible to use the images collected prior to intervention to improve the neural network's performance in the critical sections.

The generalisable of our approach extends to diverse robotic platforms and environments. The neural network was trained only on a single occasion, using images obtained from the Thorvald robot at our test farm, which has a significantly different camera view compared to the Husky and Hunter robots. Furthermore, the environment of the test farm (Fig. 6) in the images captured is considerably different to the commercial farm polytunnels (Figs. 1(b) and 1(c)).

The number of in-row interventions for the Hunter in the commercial farm is high due to the narrow rows. The rows are one-third the width of the rows the network was trained on at our test farm. The intervention rate for the Hunter robot could be potentially lowered by training the neural network using the dataset, which corresponds more closely to the view of the camera mounted on the robot, especially in the commercial farm where the number of in-row interventions is higher than at the end-of-row. It is expected that our method is unable to reliably guide the robot when the learned repetitive cue is not present in the camera view. There are multiple possible approaches to tackle this issue as it occurs at the end of the learned path. An advantageous property of our method is that the neural network can estimate the confidence of the visual cue detection, as shown in Fig. 7(c). When the confidence is low, the robot can switch to different behaviour and stop using the heading corrections estimated by the neural network. The quality of this behaviour switching can also be improved by using odometry in combination with the known length of the traversed row or indicating the end of the row by a fiducial marker.

The advantage of the presented approach is that the visual scene does not have to be consistent. The variations of the crops are a very challenging problem in agricultural environments. When a suitable dataset is provided, the neural network can learn a descriptor of the visual cue invariant to the variations provided in the data. Labelling image pairs containing tiny seedlings in one image and grown plants in the other image as positive pair is possible. The contrastive learning is then able to produce a descriptor of the plant (neural representation of the image), which is invariant to the plant's growing phase. A similar approach could be applied to multiple different variations of the environment and lighting conditions, resulting in a neural network, which is pretrained to be robust to given changes, and still a single map can be used to navigate the robot. As found previously by Krajník et al. (2018) and Rozsypálek et al. (2022a) the method was capable of handling different lighting conditions and environmental changes, the time of day the teach and repeats are conducted should not significantly affect the performance. However, with continual data gathering during the traversals and automation of the training process, the presented system can be suitable also for long-term deployment.

Our method can also have a broader impact not only in agriculture but also in other industries. The preliminary experiments show that this method is not bound only to natural environments and can be used in different repetitive environments such as corridors, warehouses, pathways or car parks. In the future, the robot could exploit multiple neural networks, each representing desired behaviour in specific environments such as centring inside a corridor, travelling along a pathway and keeping distance to a fence at the side of the pathway etc.

Our approach highlights an advantage over conventional visual SLAM methods, such as RTAB-Map (Labbe and Michaud, 2014). One notable limitation observed in visual SLAM methods is the occurrence of perceptual aliasing, which hampers the creation of reliable maps. Preliminary experiments conducted with RTAB-Map revealed subpar performance in terms of feature matching within the rows of the polytunnels. Consequently, this led to inadequate localisation of the robot, as most rows are visually similar to every other row. Furthermore, the maps generated by RTAB-Map exhibited a lack of scalability with respect to the environment's size, and a consistent map was not never created. Given these findings, the application of RTAB-Map was dismissed even before navigation attempts were made, emphasising the necessity of an alternative solution.

More teaching of trajectories would be required to perform row changing and movement to other locations outside of the polytunnels, such as to a warehouse building. However, this would require a lot of trajectories to be taught and selected by the robot to reach its intended goal, for example, in our test site every trajectory from r0.7 to every other row would need to be taught than from r1.5 to every other row and so on. Future work on a more suitable method would be to use the VTAG framework only inside the polytunnel rows and switch to another method for navigation outside the rows. SLAM methods would be suitable as visual aliasing is not a large problem outside the rows, this would overcome the limitations of both methods for navigating in these polytunnel environments.

CRediT authorship contribution statement

Jonathan Cox: Primary manuscript author, Experiments, Data analysis. **Nikolaos Tsagkopoulos:** Conceptualization, System integration, Data analysis. **Zdeněk Rozsypálek:** Neural network design, Conceptualization. **Tomáš Krajník:** Core hypothesis formulation. **Elizabeth Sklar:** Core hypothesis formulation. **Marc Hanheide:** Core hypothesis formulation.

Declaration of competing interest

Jonathan Cox reports financial support was provided by University of Lincoln. Nikolaos Tsagkopoulos reports financial support was provided by University of Lincoln. Zdenek Rozsypalek reports financial support was provided by Czech Technical University in Prague. Tomas Krajnik reports financial support was provided by Czech Technical University in Prague. Elizabeth Sklar reports financial support was provided by University of Lincoln. Marc Hanheide reports financial support was provided by University of Lincoln.

Data availability

Data will be made available on request.

Acknowledgments

Jonathan Cox (primary manuscript author, experiments, data analysis), Nikolaos Tsagkopoulos (conceptualisation, system integration, data analysis), Elizabeth Sklar (core hypothesis formulation) and Marc Hanheide (core hypothesis formulation) were funded by Innovate UK grant number 10028225, Collaborative Fruit Retrieval Using Intelligent Transportation and the Engineering and Physical Sciences Research Council [EP/S023917/1]. Tomáš Krajník (core hypothesis formulation) was supported by CSF, United States project 20-27034J and Zdeněk Rozsypálek (neural network design, conceptualisation) by OP VVV MEYS RCI project CZ.02.1.01/0.0/0.0/16 019/0000765. Thanks to James Heselden for the photo of the Thorvald robot.

References

Adhikari, S.P., Kim, G., Kim, H., 2020. Deep neural network-based system for autonomous navigation in paddy field. IEEE Access 8, 71272–71278.
Aghi, D., et al., 2021. Deep semantic segmentation at the edge for autonomous navigation in vineyard rows. In: 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems. IROS, pp. 3421–3428.
Aguiar, A.S., et al., 2020. Localization and mapping for robots in agriculture and forestry: A survey. Robotics 9 (4).
Ahmadi, A., Halstead, M., McCool, C., 2021. Towards autonomous crop-agnostic visual navigation in arable fields. arXiv preprint arXiv:2109.11936.
Ahmadi, A., et al., 2020. Visual servoing-based navigation for monitoring row-crop fields. In: 2020 IEEE International Conference on Robotics and Automation. ICRA, IEEE, pp. 4920–4926.
Åstrand, B., Baerveldt, A.-J., 2005. A vision based row-following system for agricultural field machinery. Mechatronics 15 (2), 251–269.
Bah, M.D., Hafiane, A., Canals, R., 2019. CRowNet: Deep network for crop row detection in UAV images. IEEE Access 8, 5189–5200.
Bailey, T., Durrant-Whyte, H., 2006. Simultaneous localization and mapping (SLAM): part II. IEEE Robot. Autom. Mag. 13 (3), 108–117.
Barfoot, T.D., et al., 2012. Exploiting reusable paths in mobile robotics: Benefits and challenges for long-term autonomy. In: 2012 Ninth Conference on Computer and Robot Vision. IEEE, pp. 388–395.
Bertinetto, L., et al., 2016. Fully-convolutional siamese networks for object tracking. In: Hua, G., Jégou, H. (Eds.), Computer Vision – ECCV 2016 Workshops. Springer International Publishing, Cham, pp. 850–865.
Bromley, J., et al., 1993. Signature verification using a "siamese" time delay neural network. In: Cowan, J., Tesauro, G., Alspector, J. (Eds.), Advances in Neural Information Processing Systems. Vol. 6. Morgan-Kaufmann.
Broughton, G., et al., 2021. Robust image alignment for outdoor teach-and-repeat navigation. In: 2021 European Conference on Mobile Robots. ECMR, pp. 1–6.
Cadena, C., et al., 2016. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. IEEE Trans. Robot. 32 (6), 1309–1332.
Camara, L.G., et al., 2020. Accurate and robust teach and repeat navigation by visual place recognition: A CNN approach. In: IEEE/RSJ International Conference on Intelligent Robots and Systems. IROS, pp. 6018–6024.
Cerrato, S., et al., 2021. A deep learning driven algorithmic pipeline for autonomous navigation in row-based crops. arXiv preprint arXiv:2112.03816.
Chang, C.-L., et al., 2022. Drip-tape-following approach based on machine vision for a two-wheeled robot trailer in strip farming. Agriculture 12 (3), 428.
Chen, Z., Birchfield, S., 2006. Qualitative vision-based mobile robot navigation. In: Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006, pp. 2686–2692.
Chen, Z., Birchfield, S.T., 2009. Qualitative vision-based path following. IEEE Trans. Robot. 25 (3), 749–754.
Chen, Z., et al., 2021. Navigation line extraction method for ramie combine harvester based on U-net. In: 2021 6th Asia-Pacific Conference on Intelligent Robot Systems. ACIRS, IEEE, pp. 1–7.
Churchill, W., Newman, P., 2013. Experience-based navigation for long-term localisation. Int. J. Robot. Res. 32 (14), 1645–1661.
Churchill, D., Vardy, A., 2012. An orientation invariant visual homing algorithm. J. Intell. Robot. Syst..
Clement, L., Kelly, J., Barfoot, T.D., 2017. Robust monocular visual teach and repeat aided by local ground planarity and color-constant imagery. J. Field Robotics 34 (1), 74–97.
Courbon, J., et al., 2009. Visual navigation of a quadrotor Aerial Vehicle. In: 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems. pp. 5315–5320.
Dall'Osto, D., Fischer, T., Milford, M., 2021. Fast and robust bio-inspired teach and repeat navigation. In: 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems. IROS.
Dayoub, F., Duckett, T., 2008. An adaptive appearance-based map for long-term topological localization of mobile robots. In: 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, pp. 3364–3369.
de Silva, R., Cielniak, G., Gao, J., 2022. Vision based crop row navigation under varying field conditions in arable fields. arXiv preprint arXiv:2209.14003.
Dellaert, F., et al., 1999. Monte Carlo localization for mobile robots. In: Proceedings 1999 IEEE International Conference on Robotics and Automation. Vol. 2. Cat. No.99CH36288C, pp. 1322–1328.
Durrant-Whyte, H., Bailey, T., 2006. Simultaneous localization and mapping: part I. IEEE Robot. Autom. Mag. 13 (2), 99–110.
English, A., et al., 2014. Vision based guidance for robot navigation in agriculture. In: 2014 IEEE International Conference on Robotics and Automation. ICRA, pp. 1693–1698.
Erhard, S., Wenzel, K.E., Zell, A., 2009. Flyphone: Visual self-localisation using a mobile phone as onboard image processor on a quadrocopter. J. Intell. Robot. Syst. 57 (1–4), 451–465.
Fei, Z., Vougioukas, S., 2022. Row-sensing templates: A generic 3D sensor-based approach to robot localization with respect to orchard row centerlines. J. Field Robotics 39 (6), 712–738.
Furgale, P., Barfoot, T., 2010a. Stereo mapping and localization for long-range path following on rough terrain. In: 2010 IEEE International Conference on Robotics and Automation. pp. 4410–4416.
Furgale, P., Barfoot, T.D., 2010b. Visual teach and repeat for long-range rover autonomy. J. Field Robotics 27 (5), 534–560.
Gao, X., et al., 2018. Review of wheeled mobile robots' navigation problems and application prospects in agriculture. IEEE Access 6, 49248–49268.
García-Santillán, I.D., et al., 2017. Automatic detection of curved and straight crop rows from images in maize fields. Biosyst. Eng. 156, 61–79.
Germain, H., Bourmaud, G., Lepetit, V., 2019. Sparse-to-dense hypercolumn matching for long-term visual localization. In: 2019 International Conference on 3D Vision. 3DV, pp. 513–523.
Gridseth, M., Barfoot, T.D., 2022. Keeping an eye on things: Deep learned features for long-term visual localization. IEEE Robot. Autom. Lett. 7 (2), 1016–1023.
Guerrero, J.M., et al., 2013. Automatic expert system based on images for accuracy crop row detection in maize fields. Expert Syst. Appl. 40 (2), 656–664.
Guo, J., et al., 2018. Multi-GNSS precise point positioning for precision agriculture. Precis. Agric. 19 (5), 895–911.
He, Y., et al., 2022. Automated detection of boundary line in paddy field using MobileV2-UNet and RANSAC. Comput. Electron. Agric. 194, 106697.
Krajník, T., et al., 2010. Simple yet stable bearing-only navigation. J. Field Robotics 27 (5), 511–533.
Krajník, T., et al., 2017. Image features for visual teach-and-repeat navigation in changing environments. Robot. Auton. Syst. 88, 127–141.
Krajník, T., et al., 2018. Navigation without localisation: reliable teach and repeat based on the convergence theorem. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems. IROS, pp. 1657–1664.
Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. ImageNet classification with deep convolutional neural networks. In: Pereira, F., Burges, C., Bottou, L., Weinberger, K. (Eds.), Advances in Neural Information Processing Systems. Vol. 25. Curran Associates, Inc.
Labbe, M., Michaud, F., 2014. Online global loop closure detection for large-scale multi-session graph-based SLAM. In: 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, pp. 2661–2666.

Le, T.D., et al., 2020. A low-cost and efficient autonomous row-following robot for food production in polytunnels. J. Field Robotics 37 (2), 309–321.
Li, X., et al., 2022. Robotic crop row tracking around weeds using cereal-specific features. Comput. Electron. Agric. 197, 106941.
Lin, Y.-K., Chen, S.-F., 2019. Development of navigation system for tea field machine using semantic segmentation. IFAC-PapersOnLine 52 (30), 108–113.
Luo, Y., et al., 2022. Stereo-vision-based multi-crop harvesting edge detection for precise automatic steering of combine harvester. Biosyst. Eng. 215, 115–128.
Ma, Y., et al., 2021. Autonomous navigation for a wolfberry picking robot using visual cues and fuzzy control. Inform. Process. Agric. 8 (1), 15–26.
Majdik, A.L., Albers-Schoenberg, Y., Scaramuzza, D., 2013. MAV urban localization from Google street view data. In: 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems. pp. 3979–3986.
McManus, C., et al., 2012. Visual teach and repeat using appearance-based lidar. In: 2012 IEEE International Conference on Robotics and Automation. pp. 389–396.
Mitaritonna, C., Ragot, L., et al., 2020. After Covid-19, will seasonal migrant agricultural workers in Europe be replaced by robots? CEPII Policy Brief 33, 1–10.
Nørremark, M., et al., 2008. The development and assessment of the accuracy of an autonomous GPS-based system for intra-row mechanical weed control in row crops. Biosyst. Eng. 101 (4), 396–410.
Oliveira, L.F.P., Moreira, A.P., Silva, M.F., 2021. Advances in agriculture robotics: A state-of-the-art review and challenges ahead. Robotics 10 (2).
Opiyo, S., et al., 2021. Medial axis-based machine-vision system for orchard robot navigation. Comput. Electron. Agric. 185, 106153.
Ostafew, C.J., Schoellig, A.P., Barfoot, T.D., 2013. Visual teach and repeat, repeat, repeat: Iterative Learning Control to improve mobile robot path tracking in challenging outdoor environments. In: 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems. pp. 176–181.
Paton, M., et al., 2017. I can see for miles and miles: An extended field test of visual teach and repeat 2.0. In: International Symposium on Field and Service Robotics.
Peng, C., Fei, Z., Vougioukas, S.G., 2022. Depth camera based row-end detection and headland manuvering in orchard navigation without GNSS. In: 2022 30th Mediterranean Conference on Control and Automation. MED, pp. 538–544.
Perez-Ruiz, M., Upadhyaya, S.K., 2012. GNSS in precision agricultural operations. In: Elbahhar, F.B., Rivenq, A. (Eds.), New Approach of Indoor and Outdoor Localization Systems. IntechOpen, Rijeka.
Ponnambalam, V.R., et al., 2020. Agri-cost-maps - integration of environmental constraints into navigation systems for agricultural robots. In: 2020 6th International Conference on Control, Automation and Robotics. ICCAR, pp. 214–220.
Ravikanna, R., et al., 2021. Maximising availability of transportation robots through intelligent allocation of parking spaces. In: Towards Autonomous Robotic Systems: 22nd Annual Conference, TAROS 2021, Lincoln, UK, September 8–10, 2021, Proceedings 22. Springer, pp. 337–348.
Romeo, J., et al., 2012. Crop row detection in maize fields inspired on the human visual perception. Sci. World J. 2012.
Rozsypálek, Z., et al., 2022a. Contrastive learning for image registration in visual teach and repeat navigation. Sensors 22 (8), 2975.
Rozsypálek, Z., et al., 2022b. Semi-supervised learning for image alignment in teach and repeat navigation. In: Proceedings of the Symposium on Applied Computing. SAC.
Rozsypálek, Z., et al., 2023. Multidimensional particle filter for long-term visual teach and repeat in changing environments. IEEE Robot. Autom. Lett. 8 (4), 1951–1958.
Song, Y., et al., 2022. Navigation algorithm based on semantic segmentation in wheat fields using an RGB-D camera. Inform. Process. Agric..
Taketomi, T., Uchiyama, H., Ikeda, S., 2017. Visual SLAM algorithms: A survey from 2010 to 2016. IPSJ Trans. Comput. Vis. Appl. 9 (1), 1–11.
Vardy, A., 2010. Using feature scale change for robot localization along a route. In: 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems. pp. 4830–4835.
Winterhalter, W., et al., 2018. Crop row detection on tiny plants with the pattern hough transform. IEEE Robot. Autom. Lett. 3 (4), 3394–3401.
Xiong, Y., et al., 2020. An autonomous strawberry-harvesting robot: Design, development, integration, and field evaluation. J. Field Robotics 37 (2), 202–224.
Zhou, Y., et al., 2021. Autonomous detection of crop rows based on adaptive multi-ROI in maize fields. Int. J. Agric. Biol. Eng. 14 (4), 217–225.
Zhu, Z., Das, G.P., Hanheide, M., 2023. Topological optimisation for multi-robot systems in logistics. In: SAC '23: Proceedings of the 38th ACM/SIGAPP Symposium on Applied Computing, March 27 - March 31, 2023, Tallinn, Estonia. ACM/SIGAPP, pp. 791–799.