Autonomous Navigation in Vineyards with Deep Learning at the Edge

Diego Aghi[1,2][0000−0001−5206−0943], Vittorio Mazzia[1,2,3][0000−0002−7624−1850], and Marcello Chiaberge[1,2][0000−0002−1921−0126]

[1] DET, Politecnico di Torino, Torino, Italy
[2] PIC4SeR - Politecnico Interdepartmental Centre for Service Robotic
`https://pic4ser.polito.it/`
[3] SmartData@PoliTo - Big Data and Data Science Laboratory Turin - Italy
`https://smartdata.polito.it/`
{diego.aghi, vittorio.mazzia, marcello.chiaberge}@polito.it

Abstract. With the rapid growth of the world population over the past years, the agriculture industry is asked to respond properly to the exponential augmentation of global demand for food production. In the past few years, autonomous agricultural field machines have been gaining significant attention from farmers and industries in order to reduce costs, human workload, and required resources. Nevertheless, achieving sufficient autonomous navigation capabilities requires the simultaneous cooperation of different processes; localization, mapping, and path planning are just some of the steps that aim at providing to the machine the right set of skills to operate in semi-structured and unstructured environments. In this context, the presented research exploits later advancement in deep learning and edge computing technologies to provide a robust and fully integrable local planner for autonomous navigation along vineyards rows. Moreover, the devised and tested platform necessitates only of low range and low-cost hardware with minimal power and bandwidth requirements. The machine learning algorithm has been trained and tested with acquired images during different field surveys in the north region of Italy. Then, after performing an optimization process, the overall system has been validated with a customized robot platform in the appropriate environment.

1 Introduction

Nowadays, with the continuous growth of the human population, agriculture industries and farmers have been facing the exponential augmentation of global demand of food production. According to the projections of growth established in 2017 by the United Nations [1], by 2050, the global population will be around 9.7 billion and it is expected to reach 11.1 billion in 2100. So, there is an incremental need of new techniques and technologies aimed at maximizing efficiency and productivity of every single land sustainably.

Over the years, precision agriculture [2] and digital farming [3] have gradually contributed with autonomous robotic machines and information collection to improving crops yield and resources management. Research on applications of mobile robotic systems in agricultural tasks has been increasing vastly [4]. However, despite the rising in investments and research activities on the subject, many implementations remain experimental and far from being applied on a large scale. Indeed, most of the proposed solutions require a combination of real-time kinematic GPS (RTK-GPS) [5,6] and costly sensors like three-dimensional multi-channel Light Detection and Ranging (LIDAR) [7]. Those solutions, besides being very expensive, are unreliable and prone to failure and malfunction due to their complexity. In [8], Riggio et al. proposed a low-cost solution based only on a single-channel LIDAR and odometry, but it is greatly affected by the type of canopy and condition of the specific vineyard. Instead, in [9], they proposed a vision based-control system using a clustering algorithm and Hough Transform in order to detect the central path between two rows. However, it is extremely sensitive to illumination conditions and intra-class variations.

On the other hand, several recent works [10,11] focus their efforts on finding an affordable solution for the generation of a global map with related way-points. However, path following inside vineyard rows is still a challenging task due to localization problems and variability of the environment. Indeed, GPS receivers require to function in an open area with a clear view of the sky [12] so, expensive sensors and solutions are needed in order to navigate through vineyards rows.

In this context, we present a low-cost solution for an autonomous navigation in vineyards rows using only a camera sensor and an edge computing platform. We have exploited latest advancement in deep learning and model optimization techniques to obtain a robust and power-efficient local planner to effectively navigate between vineyards canopies with only the raw RGB images coming from the front view of the robot.

The remainder of this paper is organized as follows. Section 2 introduces the materials and data used for this research. Section 3 and 4 give a detailed overview of the proposed methodology with the obtained experimental results followed by the conclusion and future works.

2 Materials and Data

In order to acquire a dataset for training and testing the network, we performed field surveys in two distinct rural areas in the north part of Italy: Grugliasco near the metropolitan city of Turin in the Italian region of Piedmont and Valle San Giorgio di Baone in the Province of Padua in the Italian region Veneto. The collected video samples present different types of terrains, wine quality and were acquired at different time of the day, with diverse meteorological conditions. Videos were shot at 1080p with a 16:9 ratio in order to have more flexibility during the data processing process.

On the other hand, as edge AI embedded computational platform, we selected NVIDIA Jetson Nano; it has been presented in June 2019 and targets applications where reducing the board size, power consumption, and price is important. For the hardware acceleration, it features a NVIDIA Maxwell GPU with peak performance of 472 GFLOPs and it can work in two power modes: 5W or 10W.

3 Proposed methodology

Our goal is to develop a real-time local motion planner with an ultra-light computational load able to overcome the problems faced by the GPS device when carrying out an autonomous navigation along a vineyard row.

Greatly inspired by Giusti et al. [13], we applied transfer learning [14] to train, with a custom dataset, a convolutional neural network (CNN) to classify the view of the front camera of the field machine in three distinct classes: left, center and right. Where, in a vineyard scenario, the class center describes the view of the camera when the vehicle is pointing at the end of the vineyard row, whereas the classes left and right are needed to indicate whether the vehicle is pointing at the left side or at the right side of the vineyard row respectively.

Successively, using the trained network predictions, we design a simple control system to route the path of the robot through vineyards rows. Moreover, we exploited latest advancement in model optimization techniques in order to obtain an efficient and lightweight network able to inference in real-time on a low-cost edge AI platform.

Finally, we integrated the deep learning model and the derived controller with Robot Operating System (ROS) framework. We evaluated the overall system in a vineyard near the city of Turin using a mobile platform with an ultrasound sensor to avoid possible collisions, a 720p USB camera and the Jetson Nano board to perform all required computations. The resulting system is a low-cost, power-efficient, and connection free local path planner that can be easily integrated with a global system achieving fully autonomous navigation in vineyards.

3.1 System workflow

The system workflow can be divided into four main phases as shown in Fig.1. Initially, a camera acquires images from the environment in front of the robot. Successively, those images are processed by the designed and optimized CNN that performs a multi-class classification locally and in real-time. After that, the control values are computed according to the class in which the classified image belongs to in order to properly drive the robot along vineyard rows. Finally, these values are sent to the robot platform using ROS that manages the signal communication to the actuators to perform the required motion.

3.2 Network architecture

We have carefully selected a deep learning known architecture that reaches high performance by also containing hardware costs. MobileNet [15] network, due to its efficient design, works reasonably fast on mobile devices and embedded systems without too much memory allocation. This choice allows us to meet our

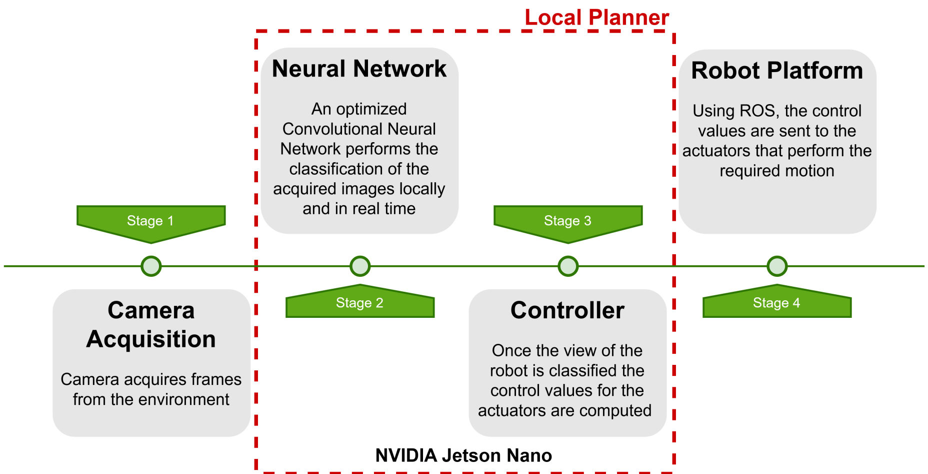

Fig. 1. An overview of the workflow of the presented system. Principal steps are reported in order with explanations of their role inside the global algorithm.

requirements of low weight and computational load and, therefore, to have a wider range of possible hardware to be employed.

We have modified the original final fully connected layers of the MobileNet by substituting them with two fully connected layers of 256 and 3 neurons, respectively. The resulting model is a CNN network with an overall depth of 90 layers and with just 3.492.035 parameters.

Moreover, we optimized the network model and we sped up the inference procedure by using TensorRT.

4 Experimental results and discussion

4.1 Dataset creation and pre-processing

We used a dataset of 33.616 images equally balanced along with the three previously introduced classes. To create the dataset, we took several videos in a variety of vineyards rows with a 1080p resolution camera in order to have more flexibility during the pre-processing phase. In particular, for the first video of the *center* class, we recorded rows with the camera pointing at its center. Whereas, for the other two videos, classes *left* and *right*, we registered with the camera rotated of 45 degrees with respect to the longitudinal axis of the row towards the left and the right side, respectively. Subsequently, we took each video as a streaming of images and we selected the best frame every six consecutive ones. Additionally, before being used to train the network, the images have been normalized and resized to the expected dimensions of the input images of the network.

(a) (b) (c)

Fig. 2. Three samples of the dataset used to train the network, one for each class. Fig.2(a) is an example of the left class, Fig.2(b) of the center class, and Fig.2(c) of the right class. Dataset samples have been collected with different weather conditions and at a different time of day. The resulting heterogeneous training set is aimed at giving generality and robustness to the model.

4.2 Training

As already introduced, we trained the network using a technique known as transfer learning [14]; instead of starting to train with weights randomly initialized, we used variables obtained with an earlier training session. In particular, we exploited weights obtained fitting MobileNet with the ImageNet classification dataset. Subsequently, we removed the last two fully connected layers and replaced them with a number of neurons equal to our number of classes. Using this technique, we were able to take advantage of previous low-level features, learned by the network, greatly reducing the number of images and epochs required for the training. Indeed, edges, contours, and basic textures are general-purpose features that can be reused for different tasks. In order to properly train, validate and test the model, we randomly divided the dataset into three subsets as follow: 70% for the training set, 15% for the validation set, and the remaining 15% for the test set. We trained the resulting network for only six epochs with a batch size of 64. To increment the robustness of the network and to overcome possible problems of overfitting we used different techniques such as dropout, weight decay and data augmentation with changes in zoom and brightness. Finally, we used RMSprop as an optimizer, accuracy as a metric, and cross entropy as a loss function.

4.3 Model Results

The implemented model has been trained and tested with the subdivision of the dataset introduced in section 4.2, giving an accuracy of 1.0 over the test set. Therefore, this model is the one employed for the navigation.

Moreover, in order to evaluate the robustness of the network over new scenarios, we performed an experimentation training the model only with a small part of the available dataset. So, we trained the architecture with just a vineyard type and tested the resulting model with five completely different scenarios with diverse wine quality and weather conditions. In particular, in Table 1, it is depicted how the dataset has been partitioned for training, validation, and testing for this second experimentation. The resulting split percentages (18, 8, 74) are due to the amount of images available for each region of the available dataset.

Table 1. Dataset division for the second evaluation of the model.

Subset	**%**	**Number of items**
Training set	18	6.068
Validation set	8	2.681
Test set	74	24.867

As shown in Table 2, an accuracy of 0.94 is achieved by the re-trained model in this second case. This is an optimal result considering the fact that the network has been trained with a very small dataset and it has been tested with a completely different vineyard scenario. This clearly demonstrates how transfer learning, for this specific task, is very effective at providing good generalization capabilities with also a small training set.

Table 2. Results of the second evaluation of the proposed model.

Class	**precision**	**recall**	**f1-score**
right	0.850	1.000	0.919
left	1.000	0.899	0.947
center	1.000	0.924	0.961
micro avg	0.941	0.941	0.941
macro avg	0.950	0.941	0.942
weighted avg	0.950	0.941	0.942

4.4 Optimization and deployment

As previously introduced, the employed network has been optimized using tensorRT. It allows us to tune the portion of available GPU memory dedicated to TensorFlow with a parameter called *per_process_gpu_memory_fraction* (PPGMF). In particular, after many attempts, in order to find the best configuration, we set the PPGMF to 0.75. TensorRT, besides not affecting the accuracy of the prediction, it gives a significant increment to the number of frames elaborated

Fig. 3. Test on the relevant environment with the Clearpath Robotics, Jackal model platform.

per second by our model. In fact, the control frequency using Tensorflow with a frozen graph was 21.92Hz, whereas, with the performed optimization, we reach 47.15Hz.

As far as the deployment is concerned, the system has been implemented in a ROS-oriented robot platform. The robot in which the local planner has been tested is an unmanned ground vehicle: the model Jackal from Clearpath Robotics (Fig.3).

The tests have been carried out in a new vineyard scenario, and the system proved to be able to perform an autonomous navigation along the vineyards row autonomously even with a low-resolution camera.

5 Conclusion and future works

We proposed a local motion planner for autonomous navigation along vineyards rows. We exploited latest advancement in deep learning optimization techniques in order to obtain a lightweight, power-efficient model able to run locally on a low-cost edge AI platform. The proposed system provides a considerably high control frequency that meets the requirements of a real-time autonomous navigation task, the predictions of the network are robust to changes in brightness, in vineyards dimensions and, eventually, to missing grapevines along the vineyard rows. Finally, the local planner that we proposed demonstrated to work exceptionally well in real conditions even if the images given as input have a very low resolution. However, as future works, an additional system with a proportional controller may be implemented in order to improve the controllability of the robot during the navigation.

Acknowledgments

This work has been developed with the contribution of the Politecnico di Torino Interdepartmental Centre for Service Robotics PIC4SeR (https://pic4ser.polito.it) and SmartData@Polito (https://smartdata.polito.it).

References

1. DESA, U.N. "World population prospects, the 2017 Revision, Volume I: comprehensive tables". In: *New York United Nations Department of Economic & Social Affairs* (2017).
2. Mulla, David J. "Twenty five years of remote sensing in precision agriculture: Key advances and remaining knowledge gaps." In: *Biosystems engineering* 114.4 (2013): pp. 358-371.
3. R Shamshiri, Redmond, et al. "Research and development in agricultural robotics: A perspective of digital farming." In: *Chinese Society of Agricultural Engineering* (2018).
4. Sharifi, Mostafa, and XiaoQi Chen. "A novel vision based row guidance approach for navigation of agricultural mobile robots in orchards." In: *6th International Conference on Automation, Robotics and Applications (ICARA).* (2015), pp. 251-255.
5. Guzmán,, R., et al. Autonomous hybrid GPS/reactive navigation of an unmanned ground vehicle for precision viticulture-VINBOT. In: *62nd German Winegrowers Conference At: Stuttgart* (2016).
6. Dos Santos, Filipe Neves, et al. "Towards a reliable robot for steep slope vineyards monitoring." In: *Journal of Intelligent & Robotic Systems* 83.3-4 (2016), pp. 429-444.
7. Astolfi, Pietro, et al. Vineyard Autonomous Navigation in the Echord++ GRAPE Experiment. In: *IFAC-PapersOnLine*, 51.11 (2018), pp. 704-709.
8. Riggio, Giuseppe, Cesare Fantuzzi, and Cristian Secchi. "A Low-Cost Navigation Strategy for Yield Estimation in Vineyards." In: *2018 IEEE International Conference on Robotics and Automation (ICRA)* (2018), pp. 2200-2205.
9. Sharifi, Mostafa, and XiaoQi Chen. "A novel vision based row guidance approach for navigation of agricultural mobile robots in orchards." In: *2015 6th International Conference on Automation, Robotics and Applications (ICARA)* (2015), pp. 251-255.
10. Santos, Luís, et al. "Path Planning approach with the extraction of Topological Maps from Occupancy Grid Maps in steep slope vineyards." In: *2019 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)* (2019), pp. 1-7.
11. Zoto, Jurgen, et al. "Automatic Path Planning for Unmanned Ground Vehicle Using UAV Imagery." In: *International Conference on Robotics in Alpe-Adria Danube Region* (2019), pp. 223-230.
12. Ma, Changlin, et al. "Gps signal degradation modeling." In: *Proceedings of International Technical Meeting of the Satellite Division of the Institute of Navigation.* (2001).
13. Giusti, Alessandro, et al. "A machine learning approach to visual perception of forest trails for mobile robots". In: *IEEE Robotics and Automation Letters* 1.2 (2015), pp. 661-667.
14. Long, Mingsheng, et al. "Deep transfer learning with joint adaptation networks". In: *Proceedings of the 34th International Conference on Machine Learning-Volume 70* (2017), pp. 2208-2217.
15. Howard, Andrew, et al. Mobilenets: EffiHoward, Andrew GHoward, Andrew Gcient convolutional neural networks for mobile vision applications. Preprint at https://arxiv.org/abs/1704.04861 (2017)