

# Strawberries Detection Using a Heterogeneous Multi-Processor Platform

Samuel Brandenburg<sup>1</sup> and Pedro Machado<sup>1,2</sup> and Nikesh Lama<sup>1</sup> and T.M. McGinnity<sup>1,3</sup>

**Abstract**—Over the last few years, the number of precision farming projects has increased specifically in harvesting robots and many of which have made continued progress from identifying crops to grasping the desired fruit or vegetable. One of the most common issues found in precision farming projects is that successful application is heavily dependent not just on identifying the fruit but also on ensuring that localisation allows for accurate navigation. These issues become significant factors when the robot is not operating in a prearranged environment, or when vegetation becomes too thick, thus covering crop. Moreover, running a state-of-the-art deep learning algorithm on an embedded platform is also very challenging, resulting most of the times in low frame rates. This paper proposes using the You Only Look Once version 3 (YOLOv3) Convolutional Neural Network (CNN) in combination with utilising image processing techniques for the application of precision farming robots targeting strawberry detection, accelerated on a heterogeneous multiprocessor platform. The results show a performance acceleration by five times when implemented on a Field-Programmable Gate Array (FPGA) when compared with the same algorithm running on the processor side with an accuracy of 78.3% over the test set comprised of 146 images.

**Index Terms**—YOLOv3, Strawberry Detection, FPGA, Xilinx, Vitis-AI, DPU

## I. INTRODUCTION

The majority of the UK's land is used for agriculture production [1]. The production of agricultural commodities aids in the effort to be self-sufficient as production sustains food levels in the country [1]. In order to support such mass production, each year, the agriculture industry requires the recruitment of hundreds of skilled labours to operate in the country's farmland [2]. A substantial amount of workers are required given the size of the operation, and that most of the work comprises of manual labour. As such, manual workers spend a large portion of their day performing hours of repetitive task, such as picking fruit and vegetables. However, due to legislation changes in England, it has become more difficult for farmers to find the workforce they need for their fields [2]. It is important to note that the shortage of field workers was present before Brexit [3]. Labour shortages ultimately increase the amount of food waste when crops are not collected before they spoil. Thus, shortening the amount

of supply to a growing demand due to continued population growth [4].

Several universities and private companies have invested in smart farming to help reduce labour shortages and increase the production of agricultural commodities [5]. While the automation of agriculture presents opportunities to counterbalance the labour shortage, smart farming also has the potential to be more effective than traditional farming equipment [4][6]. For example, heavy machinery such as tractors, which is used in traditional farming, has been shown to decrease the number of crop yields due to soil compaction [4]. In the UK Robotics and Autonomous Systems (RAS) white paper, it is argued that small intelligent farming robots can aid in the reduction of waste, improve the economy, decrease environmental harm and increase food sustainability [4]. Therefore, there is a great need for innovation, that could be potentially met by using smart farming robots capable of performing at a commercial level. The soft fruit sector could especially benefit from automating harvest collection on high yield fruit like strawberries.

However, a robot capable of picking strawberries must meet certain requirements to be successful. The first requirement is detecting and localising the fruit. Detection and localisation are essential because it takes visual data as input. The collected visual information is then analysed using a variety of different methods. A standard and suitable design for detecting fruit use colour sensors that take the colours in the image and analyses the pixels by colour and shape. Calculations are conducted within the colour space to determine if there are any items in the picture. Once one or more strawberries are identified, a localisation strategy is needed. Localisation is the process of identifying the location of each located fruit. Once the fruit has been properly localised, trajectory planning is required.

Trajectory planning is the second essential task and equally as challenging as detecting the desired crop. For this, a pre-planning approach may be more suitable for reaching fruit located under challenging positions verse doing the detection, localisation and end-effector manipulation all at once. The project measures successful trajectory planning based on three main criteria's; starting with how well it can manoeuvre around without being blocked or entangled by vegetation. In the case where such criteria can not be met, the robot should proceed with the next detected strawberry. The second objective is to have the gripper positioned within grabbing distance of the desired fruit. The last objective measures robot control and perception by ensuring that nearby crops are not damaged.

The next task combines grasping the fruit and removing it

<sup>1</sup>Samuel Brandenburg, Pedro Machado, Nikesh Lama, T.M. McGinnity are with Computational Neuroscience and Cognitive Robotics Group, School of Science and Technology, Nottingham Trent University, Clifton Campus, United Kingdom. samuel.brandenbourg2016@my.ntu.ac.uk and {pedro.batpistamachado, nikesh.lama02, martin.McGinnity}@ntu.ac.uk

<sup>2</sup>Pedro Machado with the Sundance Multiprocessor Technology Ltd., Chesham, Buckinghamshire, UK. pedro.m@sundance.com

<sup>3</sup>T.M. McGinnity is also with the Intelligent Systems Research Centre, Ulster University, Magee Campus, Northern Ireland, UK. tm.mcGinnity@ulster.ac.uk

from its peduncle. The challenges are to remove the peduncle and grasp the fruit without inflicting any damage to it. Removing the peduncle wrong leads to the fruit being exposed, causing early spoilage. The other crucial part of the task is grasping the strawberry with a suitable amount of pressure. Applying too much pressure can lead to bruising on the strawberry. Two methods to determine how much pressure to apply are the use of visual and pressure sensors. When a strawberry ripens, its' colour becomes darker; therefore, the robot can begin adjusting what grip is suitable by the degree of ripeness. In the case of uneven ripeness, a pressure sensor on the gripper would be the most suitable means of determining the amount of pressure to apply.

The overall smart farming process can be accomplished by improving farming vision, tactile sensing and perception. In this paper a vision system is achieved through the use of YOLOv3 operating on the Xilinx ZCU104 development board <sup>1</sup>, a heterogeneous platform fitted with the Xilinx Multi-Processor ZU7EV which is analysed for its suitability for identifying and localising crops such as strawberries.

A brief literature review is presented in section II, the methodology in presented in section III, preliminary results are discussed in section IV and the conclusions and future work are given in section V.

## II. RELATED WORK

The UK employs up to 29,000 seasonal pickers and generates over 160,000 tons of fruit each year [7]. One can see that agriculture requires a tremendous amount of people to produce adequate returns. However, it is essential to note that while such a high number of people are employed there is still a shortage of workers. The shortage of workers is, in fact, an invaluable opportunity to maximise production through automating soft fruit harvesting like strawberries. One significant aspect of automating the collection of strawberry is that they are grown in most places in the world [7]; thus, an automated harvest would bolster the efficiency of strawberry agriculture world-wide. Currently, there is an extended amount of research conducted on fruit and vegetable picking robots. These research efforts have brought about significant advances in fruit-picking robots, such as the Argobot<sup>2</sup>, which uses 24 end-effectors to pick strawberries, and Sweeper, which is a robot designed to collect yellow peppers [8]. While there are many robots capable of locating and grasping said fruit or vegetable, none are at a commercial standard [5] as of yet. These robots have not yet achieved a better harvesting performance than the methods currently commercially used for fruit or vegetable collection [7].

Any robot task with crop picking is faced with the questions of what methods will the robot use for information gathering on its environment. For instance, will the system implement computer vision [9], and if so, what methods would it use to process visual information; would

it use a neural network or a machine learning algorithm? One group focused on designing an algorithm that used multi-template matching for identifying cucumbers in natural surroundings[10]. Bao *et al.* [10] proposed the use of an image that serves as a reference point for incoming images. The reference point slides across an image and is then compared with the template to identify labels in picture [10]. Their system used 65 templates to detect mature Radit cucumbers. The method proposed by Bao *et al.* [10] was designed to overcome the issue of misidentifying the cucumber due to its complex environment[10]. The author states other methods which focus on the feature extraction of the cucumber such as the shape, or the colour, struggle with accurate detection. The cucumber, unlike the strawberry, is green, which makes it harder to differentiate between the vegetable and the leaf [10]. Nonetheless, the author reported that their implementation was able to achieve 98% accuracy [10]. In spite of the results being certainly interesting, the work was done in 2016, and new emerging methodologies should be explored. In this case, template matching can still be improved to distinguish crop despite occlusion and lighting[2]. However, template-matching shows promising results when differentiating between subtle differences.

Other projects have been done on creating a smart farming robot for apple extraction [11]. Red apples, unlike cucumbers, are easier to distinguish from leaves. However, automating apple farming presents challenges in visually recognising maturity because of uneven maturity [11]. The author proposed a visual method using red, green, and blue (RGB) sensors to identify mature apples. One clear difference in the previous project on cucumber recognition was their visual processing method. They used a multi-template technique for visualisation. However, the Ni *et al.* [11] project utilised an image processing algorithm based on an RGB colour scheme. The colour scheme is popular for colour models because a large array of colours can be created from varying the degrees of red, green and blue values. The system was broken into separate tasks. The first task was acquiring visual data to process, which was done by taking snapshots of the apple tree. Once the image was acquired, the image was then analysed based on its RGB traits [11]. Image processing based on colours play a crucial role in identifying objects of interest from noise [11]. Ni's method is beneficial because it is intuitive and able to adapt to changes in colour. However, it was asserted that Ni's method of intensifying the colour spectrum to identify apples might not be as effective in image processing if there are subtle differences with its environment[10]. Furthermore, it may be beneficial to add supplemental vision techniques to increase the accuracy rate and reduce false positives. False positives mislead the system to believe it has the correct prediction [11]. The most critical similarity is that apples and strawberries share the same colour. Therefore, image processing based on RGB sensors may prove essential for a foundational understanding of the farming environment for strawberries.

The red and green colour scheme for image processing was utilised in a smart farming project focused on automat-

<sup>1</sup>Available online, <https://www.xilinx.com/products/boards-and-kits/zcu104.html>, last accessed 30/07/2020

<sup>2</sup>Available online, <https://www.agrobot.com/>, last accessed: 31/07/2020

ing cherry tomato harvesting [12]. Similar to the previous projects, the main challenge was identifying mature cherry tomatoes [12]. Feng *et al.* [12] also used a colour scheme method for image processing, but instead of using RGB, they used R-G. The author stated that a red and green model would help differentiate the targeted tomatoes from its background as a consequence of the colour intensity formed by having only red and green to represent their colour spectrum. The article reported having an 83% harvest success, yet the robotic arm would sometimes collide with the branches, therefore, in its attempts to grab a tomato [12]. Ultimately, the colour model can be used in different ways to reduce the amount of white noise in image processing [12]. The proposed method was reported useful for tomato picking, and it could be useful for collecting apples. It is unlikely that it would be useful in cucumber harvesting due to cucumbers sharing the same colour with their background. Intensifying the colour spectrum may only create more false positives identifying leaves as cucumbers. The proposed method could be beneficial for strawberry picking as they are sometimes grown in clusters. Strawberries also ripe unevenly, so identifying the mature strawberries could be more challenging when closely confined with other berries [5].

Another group provided two research papers on strawberry picking. The first report used an RGB Depth camera and two infrared sensors to detect the depth of the object in its environment. The visual component of their robot system has two abilities: detection of individual strawberry, the localisation of the fruit for its separation. The process of filtering out pixel noise is used to assist in identifying strawberries by using a colour threshold[5]. The author stated that image processing based on RGB provides fast performance for live detection. It also discussed cases where the robot failed due to the vision system. They reported that their vision system was impaired by occlusions, duplicate detections, inaccurate localisation, and segmentation failures [5]. Obstruction of the camera view appears to be the main issue in each smart farming project. It appears that it is more challenging when harvesting strawberries, probably, due to strawberries being small and easily covered by leaves and easily hidden from the robot's end-effector. Another unsuspected issue is duplicate detections of the same fruit. The research that covered cucumber detection used template-matching however, it appeared that objects did not obscure the camera view from taking sufficient images during testing. Nonetheless, a multi-template system may help mitigate duplicate detection. Xiong *et al.* [7] attempted to improve their computer vision algorithm by creating a subsystem that created a threshold for adapting to colour changes specifically to sunlight. The subsystem did its calculations on 2-dimensional images and allowed for simplicity in their algorithm and fast performance. However, segmentation issues caused localisation error while processing the image. Xiong *et al.* suggested that it could potentially be resolved by doing 3-dimensional image processing[7]. However, the approach would require more complex solutions, for instance, a deep

learning algorithm [7].

In addition to computer vision, it is important to consider what sensors can be used to gauge distance from the targeted crop. These are interlinked with the overall performance of the smart farming robot, and research shows that different sensors can be invaluable in simplifying distance calculations [2]. The use of different sensors is important because it can assist the trajectory planning for the chosen end-effector. Once the end-effector is designed, and its range of motion has been determined, a trajectory planning algorithm can be tested [13]. All of these separate components working together are needed to plan a successful grasping attempt. An important note is how trajectory planning and design of the end-effector is based on the nature of the crop and raises important questions to answer when attempting a grasping application. Each robot is faced with visualising the targeted crop, using a well-designed end effector suited for grasping the harvest, and trajectory planning to the target crop. Previous work [13] shows that soft fruit requires more care when handling them, unlike apples, to prevent bruising.

The size of the fruit and the way it grows may present additional difficulties when attempting to grasp the item. A good example is comparing how cucumbers grow compared to strawberries. Cucumbers grow in a way that makes them easy to reach because they usually are not obscured by vegetation. ; however, strawberries can grow in quite an unpredictable and hidden fashion[7]. Therefore, it is essential first to understand the nature of how strawberries will grow and what planning can be done in advance to increase performance. An essential aspect to consider when analysing the challenges to strawberry picking is their environment. Strawberries planted in greenhouses may grow in a more organised manner, so the fruits are less likely to be hidden from the view of the robot[14]. As a result, it makes it easier for a robot to find and collect each strawberry. However, the same cannot be said about strawberries grown in a field, as they would require more probing around obstacles to find and grasp them[7]. A low-efficiency rating will cost the agriculture industry unnecessary loss, thus defeating the purpose of smart farming and addressing the needs of the robot efficiently complete the tasks.

Another critical challenge is that strawberries ripe unevenly thus further complicating grasping the soft fruit. If the end-effector grasps the strawberry with too much pressure, the fruit can be damaged[9]. Gauging the right amount of force must be adaptable to the ripeness of each fruit. One way to determine if a fruit is ready to collect is based on its ripeness which is reflected in its colour. The robot must adapt to the different ranges of red in strawberries to differentiate the pre-mature fruit from the mature ones. Finally, strawberries are small in size and sometimes grow in clusters. Strawberry picking is considerably challenging for a robot[7], because the robot has to distinguish each fruit from the other while locating the mature strawberry in a cluster of different colours. Failure to accurately identify the right berry may result in picking the wrong one or damaging nearby strawberries. Therefore, such a task would not be feasible

without computer vision.

### III. METHODOLOGY

Our review on automated robots clearly shows that while each picking task is different, the robot’s visual sensory largely impacts its success. Although one could argue that the end-effector and gripping strategy plays an equal part in the success of precision farming, it is evident that without a reliable and robust way to visually survey the crops environment, the following task may fail. Thus the contribution to our paper analyses how YOLOv3 operating on the Xilinx ZCU104 development board could improve strawberry classification—thus providing an essential function for fruit picking robots.

A fruit grasping algorithm includes five distinct phases, namely, 1) fruit classification: use the CNN algorithm to classify fruits in the image; 2) pose estimation: estimate the best pose to pick the object; 3) path searching: search the best path assuming the current position of the objects in the scene; 4) path planning: predict moving objects trajectory and estimate the optimal path while avoiding collisions and 5) grasping: process grasping the target fruit.

The fruit grasping steps are represented in Figure 1.

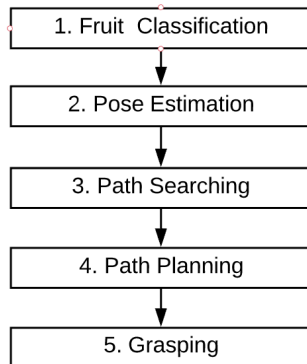


Fig. 1. Fruit grasping steps

The focus of this paper is on phase 1: the fruit classification. The objective during the classification stage is to improve strawberry detection in 6 key areas. These areas comprise of improving localisation accuracy, building model durability to light variances, enabling background adaptability, identifying partially hidden strawberries, increasing accuracy confidence, and achieve real-time processing speed. To accomplish such feats, the project utilises the deep learning network YOLOV3, which will operate on a Xilinx ZCU104 development board.

Our custom YOLOV3 model was trained on 1454 images of strawberries obtain from various sources. The selection of images played a key part in ensuring our model could adapt to light and background variances. This is also important when training the network to identify strawberries partially engulfed by vegetation. To reduce the resource demands for training, the images were resized to 416x416. Reducing the size of the images allowed our machines to process training

better, but it also meant that there was a loss of detail in the images. The model was trained up to 30k epoch’s<sup>3</sup> which iterated by 10k. After testing each model trained at varying epochs, it was evaluated that the model trained at 20k epochs produced the best accuracy; Therefore, this model was chosen to operate on the Xilinx ZCU104.

The Xilinx ZCU104 development board, powered by a powerful Xilinx Zynq UltraScale+ ZU7EV MPSoC, was used to implement and accelerate the YOLOv3 at the edge. The Xilinx Zynq UltraScale+ ZU7EV MPSoC is characterised by including a 64-bit quad-core ARM Cortex A53 as the Processor System (PS) for running a standard embedded Linux Operating System, a Graphical Processing Unit (GPU) for accelerating the graphics processing, Field-Programmable Gate Arrays (FPGA) for accelerating the state-of-the-art AI algorithms and Real-Time Processing Unit (RTU) for processing events in real-time<sup>4</sup>. Only the PS and FPGA units were used.

An Intel RealSense D435i camera<sup>5</sup> is used to capture RGB and depth image frames that are forwarded to the MPSoC. The classification is accelerated on the FPGA side, and the remaining steps are completed on the processor system, which then controls the mechanical arm to pick the fruit. The overall approach is shown in Figure 2.

Resource consumption in embedded systems and the lack of fruit and vegetable datasets are two drawbacks with using deep learning for crop classification. Thus the reasons why traditional image processing techniques have been widely used[15]. However, deep learning algorithms used for object detection are more adaptable to variance in light and are able to detect sophisticated patterns in data. While using deep learning for precision farming trades off speed for recognising complex images, utilising an optimised AI model on a Multi-Processor System-on-Chip (MPSoC) could help balance the scale between speed and accuracy. To do this, the Vitis-AI tool was used to quantise the 32-bit floating-point model into an 8-bit processing model. Quantising the model reduces its overall complexity while ensuring very little accuracy is lost<sup>6</sup>. The quantising process is then followed up by compiling the model which enables the model to utilise the board’s deep processing unit (DPU).

The Xilinx Deep Learning Processor Unit (DPU) is a programmable engine that was designed for accelerating convolutional neural networks. The DPU contains a register module, a data controller module, and a convolution computing module. There is a specialised instruction set for the DPU, which enables it to accelerate several state-of-the-art convolutional neural networks (CNNs)<sup>6</sup>.

The DPU was configured with the Xilinx recommended

<sup>3</sup>One epoch is when an entire dataset is passed forward and backwards through the neural network only once.

<sup>4</sup>Available online, <https://www.xilinx.com/products/silicon-devices/soc/zynq-ultrascale-mpsoc.html>, last accessed 31/07/2020

<sup>5</sup>Available online, <https://www.intelrealsense.com/depth-camera-d435i/>, last accessed 31/07/2020

<sup>6</sup>Available online, <https://www.xilinx.com/products/intellectual-property/dpu.html>, last accessed 31/07/2020

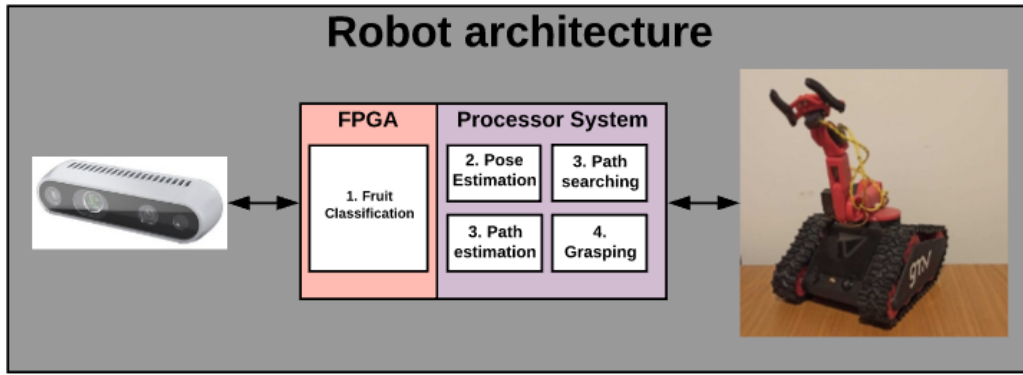


Fig. 2. High Level Block Diagram

parameters<sup>7</sup>.

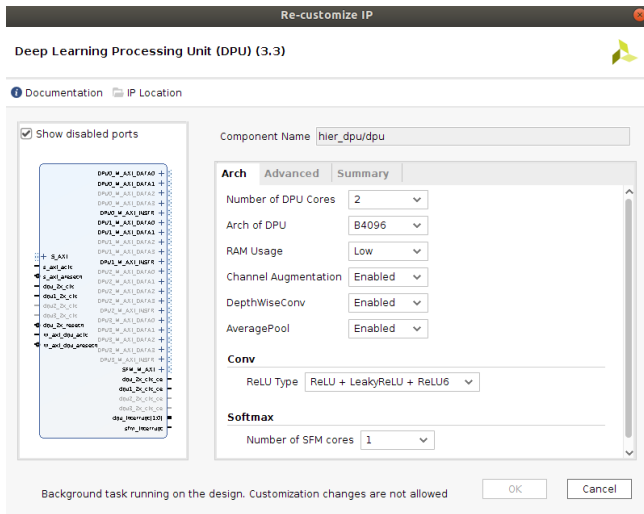


Fig. 3. DPU configuration with 2 cores of the B4096 architecture, low RAM usage, channel augmentation, Depth Wise convolution, ReLU type: ReLU + LeakyReLU + ReLU6 and 1 SFM core

The Vitis-AI development environment was designed for AI inference targeting the Xilinx hardware platforms, the including ZCU104 used in this project. It consists of optimised IP (e.g. the DPU), tools, libraries, models, and example designs<sup>8</sup>.

Table III shows a list of resources and tools used in the project to perform the classification using YOLOv3 accelerated on the FPGA side.

The report generated by Xilinx Vivado shows the utilisation of:

- 40% of Look-up-Tables (LUT): used to buffer data
- 8% of LUT distributed RAM (LUTRAM): used as small data buffers;

<sup>7</sup>Available online, [https://www.xilinx.com/support/documentation/ip\\_documentation/dpu/v3\\_2/pg338-dpu.pdf](https://www.xilinx.com/support/documentation/ip_documentation/dpu/v3_2/pg338-dpu.pdf), last accessed: 30/07/2020

<sup>8</sup> Available online, <https://www.xilinx.com/products/design-tools/vitis/vitis-ai.html>, last accessed 31/07/2020

TABLE I  
LIST OF EQUIPMENT AND TOOLS

Resources	Type	Description
GMV/Sundance Dennis robot prototype	Dennis robot prototype fitted with the Sundance VCS-junior system prototype	The Sundance VCS-junior system prototype is light version of the Xilinx ZCU104 designed to control small robots.
Xilinx ZCU104 development board	Xilinx Zynq UltraScale+ ZU7EV MPSoC	FPGA used for accelerating the classification of strawberries.
Intel RealSense D435i sensor	Camera	Camera used to capture colour and depth image frames.
YOLOv3 algorithm	AI-Model	A single shot algorithm that localises and classifies objects from one input.
Xilinx Vivado	Xilinx EDA tool	used for implementing the DPU on the FPGA side and connect it to the PS side using the AXI bus.
Xilinx DPU	Xilinx Deep Processing Unit	used for accelerating YOLOv3 on the FPGA side.
Vitis-AI quantizer	Xilinx tool	Converts 32-bit floating-point weight into a fix-point 8 bit integer
Vitis-AI compiler	Xilinx tool	Plots the AI model to an instruction set to be used by the DPU
Deep Learning processor unit	Xilinx tool	An optimised engine which can be programmed for deep neural networks use.

- 37% of flip-flops (FF): used to describe logic circuits;
- 57% of built-in RAM (BRAM): store data;
- 55% of Digital Signal Processing (DSP): used to process signals inside the FPGA;
- 1% of Global Clock Buffer (BUFG): used to buffer signal and assure that all the logic circuits receive the same clock signal within the acceptable tolerances;
- 13% of Phase Locked Loop (PLL): clock controller is used compensate the clock signal;

The resources utilisation shows that about half of the FPGA side is available to perform acceleration of algorithms from the next phases of fruit picking (i.e. pose estimation until grasping).

#### IV. PRELIMINARY RESULTS

Two tests were conducted to determine how well the model could identify strawberries in a farm setting. The first test assessed the detection percentage over the validation set of a 146 images. The validation set contained images of strawberries not limited to a farming environment. When testing the model threshold was set to 70% this parameter reduces the risk of false positives. As a result, the model was able to detect at an impressive 78.3% over the validation set. The second test used a youtube video collected on a strawberry farm<sup>9</sup>. It is important to note that the full video does not contain strawberries on frame; however, this allowed us to gather insight on how well the model would perform in adverse environments. The video contained a total of 7439 frames. Our experiment analysed the video and created two filters. The first was to check how many frames did our model identify a potential strawberry. In this case, the model detected strawberries in 3371 frames. Of this, 2346 frames were identified as having a low confidence score ranging from .30 percent to .60 percent. When identifying these frames, it was evident that the model, even at a low confidence rate had few false positives; however, the model was not able to capture a high degree of all the strawberries in any frame. Furthermore, the network struggled to identify closely coupled strawberries. Another key area was localisation accuracy; this area focuses on how well was the strawberry encapsulated in the bounding box. Unfortunately, results were volatile; therefore, measuring the localisation accuracy was unreliable. Certain frames showed that the strawberry was located within the centre of the bound box, while other images of strawberries were partially encapsulated. In some cases, the network would falsely classify the ground or a leaf with red coating. Due to the current localisation results, other methods such as segmentation by colour should be used to calibrate the bounding box and help reduce false positives. The second metrics to evaluate was how fast the network detected on the ZCU104 Development board after being accelerated by the DPU.

TABLE II  
FRAME RATES OBTAINED RUNNING YOLOV3 IN DIFFERENT ARCHITECTURES

Model	Frames per second (FPS)	Device
YOLOv3 on the processor system	2.66	Raspberry PI 3 model B
YOLOv3 accelerated on the FPGA	13.05	ZCU104 with no modifications
YOLOv3 [16]	30	Workstation fitted with a NVIDIA Quadro P2200 and Intel Xeon Gold with 20 hyper-threaded cores and 64 GB of DDR3

Table IV displays the frames per second achieved running YOLOV3 on different processors. The YOLOv3 achieved

<sup>9</sup>Available online, <https://www.youtube.com/watch?v=-br6dVv9yDs>, last accessed 15/08/2020

30 FPS when running on the powerful workstation and accelerated by a powerful Graphics Processing Unit (GPU) which is not suitable to be installed on robots and has more processing resources than embedded processors such as the 64-bit quad-core ARM cortex a53 which is found in the ZCU104. When running the YOLOv3 on the Xilinx ZU7EV Processor System, 2.66 FPS was achieved whereas, the ZCU104 Processor System accelerated by the DPU, described on the FPGA side achieved the impressive 13.05 FPS<sup>4</sup> demonstrated when detecting a strawberry. It is important to highlight that the maximum power consumption was 20W as opposed to the typical 350W consumed by the workstation described in Table IV.

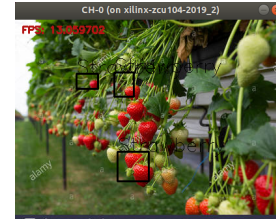


Fig. 4. Classification of a strawberry using the YOLOv3 running on the MPSoC

The network performed surprisingly well when identifying strawberries in different lighting. While true, the networks detection on the bases of lighting was not consistent. The network tended to identify strawberries that were well lighted better than the ones cascaded by a shadow.

In this evaluation, the model showed that different backgrounds did not affect the network. This could be seen in the video as the background was everchanging. Another key aspect is that the video was not used in the training data.

Another property that displayed that the model began to generalise is that it showed signs that it was capable of detecting partially hidden strawberries. Generalisation was displayed at different ranges of distances proving that the model was beginning to become quite robust. Furthermore, our model was able to detect partially hidden strawberries. While the model was able to identify strawberries hidden in vegetation, there is also the issue of false-positive.

#### V. CONCLUSION AND FUTURE WORK

Currently, the project is proceeding to the second development stage, which poses estimation, while improving classification by increasing the training dataset. Therefore, the current achievements of this work is the classification of strawberries using YOLOV3 on a ZCU104 development board which is accelerated using a DPU. Hence, achieving fast and accurate detection at 13.05 FPS within its' testing stage. However, as with many deep learning algorithms, their accuracy is dependent on the quality and volume of data. Thus, there is still a need to improve the accuracy of the classifications with our model. This especially true when detecting strawberries in complex environments like greenhouses or farmland. Nonetheless, the current results show that our model can adapt to varying degrees of light and



Fig. 5. Classification of strawberries over four image frames extracted from a video footage

can accurately detect strawberries in ideal positions. The key takes away is to include more data, especially images that contain strawberries in clusters. As such, it is expected that problems will arise when testing the algorithm in different settings—another foreseeable issue adapting the algorithm to detect strawberries in clusters. More research and testing would need to be conducted to find a potential solution.

Future work will include increasing the model's FPS. This will be done by utilising the Xilinx optimiser tool to prune and finetune the model. Most neural networks have redundant layers that are not essentially needed. The Xilinx optimiser tool allows an engineer to remove specified layers to improve detection time. The removal of layers, however, does reduce the models' accuracy slightly, but finetuning the model can help restore accuracy. Thus, achieving faster detection with little accuracy lost. Future work will also include using a colour segmentation algorithm to help calibrate the bounding boxes over the strawberry and assist with estimating the position of the strawberry. Once these key conditions have been met, further testing will include the use of a robot. However, our robot will accomplish faster image processing by accelerating the AI model using the Sundance VCS-1<sup>10</sup> and VCS-junior systems fitted with a powerful Xilinx ZU4EV MPSoC<sup>11</sup>.

#### ACKNOWLEDGMENT

The authors would like to thank to Sundance Multiprocessor Technology Ltd. and the Sundance University Program for the financial support, access to equipment, and technical support provided.

<sup>10</sup>Available online, <https://www.sundance.com/vcs-1/>, last accessed 30/07/2020

<sup>11</sup>Available online, <https://www.xilinx.com/support/documentation/selection-guides/zynq-ultrascale-plus-product-selection-guide.pdf>, last accessed 30/07/2020

#### REFERENCES

- [1] A. Angus, P. J. Burgess, J. Morris, and J. Lingard. Agriculture and land use: Demand for and supply of agricultural commodities, characteristics of the farming and food industries, and implications for land use in the UK. *Land Use Policy*, 26(SUPPL. 1):230–242, 2009.
- [2] Robert Bogue. Fruit picking robots: has their time come? *Industrial Robot*, 47(2):141–145, 2020.
- [3] Damian Maye, Hannah Chiswell, Mauro Vigani, and James Kirwan. 'Present realities' and the need for a 'lived experience' perspective in Brexit agri-food governance. *Space and Polity*, 22(2):270–286, 2018.
- [4] Uk-ras White. Agricultural Robotics: The Future of Robotic Agriculture. 2018.
- [5] Ya Xiong, Cheng Peng, Lars Grimstad, Pål Johan From, and Volkan Isler. Development and field evaluation of a strawberry harvesting robot with a cable-driven gripper. *Computers and Electronics in Agriculture*, 157(December 2018):392–402, 2019.
- [6] Rahul Dagar, Subhranil Som, and Sunil Kumar Khatri. Smart Farming - IoT in Agriculture. *Proceedings of the International Conference on Inventive Research in Computing Applications, ICIRCA 2018*, (Icirca):1052–1056, 2018.
- [7] Ya Xiong, Yuanyue Ge, Lars Grimstad, and Pål J. From. An autonomous strawberry-harvesting robot: Design, development, integration, and field evaluation. *Journal of Field Robotics*, 37(2):202–224, 2020.
- [8] Ahmad Ostovar, Ola Ringdahl, and Thomas Hellström. Adaptive image thresholding of yellow peppers for a harvesting robot. *Robotics*, 7(1):1–16, 2018.
- [9] Shigehiko Hayashi, Kenta Shigematsu, Satoshi Yamamoto, Ken Kobayashi, Yasushi Kohno, Junzo Kamata, and Mitsutaka Kurita. Evaluation of a strawberry-harvesting robot in a field test. *Biosystems Engineering*, 105(2):160–171, 2010.
- [10] Guanjun Bao, Shibo Cai, Liyong Qi, Yi Xun, Libin Zhang, and Qinghua Yang. Multi-template matching algorithm for cucumber recognition in natural environment. *Computers and Electronics in Agriculture*, 127:754–762, 2016.
- [11] Xindong Ni, Xin Wang, Shumao Wang, Shubo Wang, Ze Yao, and Yibo Ma. Structure Design and Image Recognition Research of A Picking Device on the Apple Picking Robot. *IFAC-PapersOnLine*, 51(17):489–494, 2018.
- [12] Qingchun Feng, Wei Zou, Pengfei Fan, Chunfeng Zhang, and Xiu Wang. Design and test of robotic harvesting system for cherry tomato. *International Journal of Agricultural and Biological Engineering*, 11(1):96–100, 2018.
- [13] Ya Xiong, Pal Johan From, and Volkan Isler. Design and Evaluation of a Novel Cable-Driven Gripper with Perception Capabilities for Strawberry Picking Robots. *Proceedings - IEEE International Conference on Robotics and Automation*, (2):7384–7391, 2018.
- [14] Yuanyue Ge, Ya Xiong, Gabriel Lins Tenorio, and Pal Johan From. Fruit Localization and Environment Perception for Strawberry Harvesting Robots. *IEEE Access*, 7:147642–147652, 2019.
- [15] Niall O'Mahony, Sean Campbell, Anderson Carvalho, Suman Harapanahalli, Gustavo Velasco Hernandez, Lenka Krpalkova, Daniel Riordan, and Joseph Walsh. Deep Learning vs. Traditional Computer Vision. *Advances in Intelligent Systems and Computing*, 943(Cv):128–144, 2020.
- [16] Joseph Redmon and Ali Farhadi. YOLOv3: An Incremental Improvement. 2018.