Using pre-trained convolutional neural networks to predict maturity levels of strawberries

Jurriaan Buitenweg
Supervisor(s): Junhan Wen, Thomas Abeel
EEMCS, Delft University of Technology, The Netherlands
22-6-2022

A Dissertation Submitted to EEMCS faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering

Abstract

To reduce food waste, the strawberry harvesting process should be optimized. In the modern era, computer vision can provide huge amounts of help. This paper focuses on optimizing pre-trained convolutional neural networks (CNN) to determine the maturity level of strawberries on a 1-10 scale. Here, 1 means unripe and 10 means overripe. Maturity level 8 is marketable. Experiments are done with VGG19, Resnet50, InceptionV2, Alexnet, and EfficientNetB2 as classifiers on segments using ADAM and SGD as optimizers and cross-entropy as loss function. The same CNN's are applied as a backbone for FasterRCNN to see how they would behave within an object detection architecture. The biggest challenge during this research was the low amount of training data. The research showed that using convolutional neural networks as a maturity level predictor is possible, but a well made training set with an equal spread for each maturity level is necessary to possibly achieve high accuracy.

1 Introduction

Currently, almost 10% of strawberries go to waste during the harvest period [1]. Reducing food waste is an important tool in preventing further global warming, reducing world hunger and increasing economic growth [2]. The basis of this research is to see whether computer vision and AI can improve the harvest process and therefore result in a decrease in food waste.

The focus in this research is on neural networks, specifically convolutional neural networks. A convolutional neural network (CNN) is a neural network targeted towards image analysis. There is a lot of research done on using convolutional neural networks to classify the maturity of strawberries, so the focus will be on optimizing existing convolutional neural networks instead of building a new one.

Previous research was mainly done on classifying if a strawberry is (almost) ripe or not. In this research, the focus will be on optimizing pre-trained neural networks on predicting maturity levels (1-10) from RGB images, improving upon previous research.

CNNs used in this research are AlexNet, VGG19, Resnet50, InceptionV3, and EfficientNetB2. All of te models used are pre-trained on ImageNet [3]. These CNNs will be compared as classifiers on already segmented strawberries and as a backbone for Faster RCNN [4]. Each model will be compared when trained on stochastic gradient decent (SGD) [5] or adaptive moment estimation (ADAM) [6]. For classification, cross entropy [7] will be used as loss function. When used as a backbone, the loss function described in the original Faster RCNN paper will be used. Early stopping is applied to prevent overfitting.

These tasks result in the following research questions:

Can pre-trained convolutional neural networks be used for predicting strawberry maturity levels?.

1. Can existing pre-trained convolutional neural networks be used to classify the maturity level on a 1-10 scale given a segment? If so, how accurate?
2. Can existing pre-trained convolutional neural networks be used as a backbone within FasterRCNN to detect strawberries and classify their maturity level directly on a 1-10 scale? If so, how accurate?

To answer the first question, top1 and top3 accuracy together with the mean squared error (MSE) will be used as an evaluation metric. For the second question, MSE and averaged intersection over union will be used as an evaluation metric. The best performing models will be discussed in-depth with the help of loss function plots and a confusion matrix.

In 2, related work will be discussed. All the models used in this research will also be explained. Methods and materials used will be laid out in 3. The results of the experiments can be found in 4. A review on the responsibility within this research together with reproducibility is written down in 5. A discussion on the results of this paper and previous papers can be found in 6. Finally, the conclusion and future work section is to be found in 7.

2 Related work

There is already quite some work done in the area of using computer vision to accomplish fruit classification. Two important examples are: Measuring maturity and quality using machine vision and multi-spectral imaging to determine quality attributes and ripeness [8], and research regarding hyper spectral imaging, resulting in an accuracy of 98.6 percent [9].

Research conducted by a team from Indonesia [10] also showed the potential of using convolutional neural networks as a classifier. They first compared classification on strawberry segments with only two classes (ripe, unripe), and later with 4 stages of maturity. VGG achieved the highest accuracy with 96.59% and 89.12% respectively. This research improves upon the research from Indonesia by designing a system which is able to classify the maturity levels on a scale from 1 to 10.

What was missing from the paper from Indonesia is the combination of classification with object detection. This is important to look at since object detection parameters could have an impact on accuracy. In this research, an experiment will be done to see how the models behave in an object detection environment.

Research from Peru [11] applied a convolutional neural network to the maturity classification of apples, bananas, oranges and also strawberries. They obtained a precision of 96.34%. This accuracy was acquired by implementing a new CNN inspired on VGG-16 without pre-training. A big difference between this research and the research from Peru is that only existing pre-trained CNN's are used.

All CNN's used in this research are already extensively used in previous research and well implemented within multiple deep learning frameworks like Pytorch and Keras. For each CNN used, a small overview of its structure will be given below. Faster RCNN will also be explained because it will be used for object detection within this research.

2.1 Available CNN's and Faster RCNN

These CNN's were picked because they provide state-of-the-art accuracy according to the ImageNET classification challenge [3] and are also extensively pre-trained on ImageNET.

Available backbones

1. AlexNet [12] (figure 1)
   Using 5 convolutional layers with each increasing the amount of filters (but decreasing filter size), the Re-Lu activation function and max-pooling as pooling technique, AlexNet (figure 1) achieved state-of-the-art accuracy on the ImageNet recognition challenge in 2012 with an accuracy of 87.5 percent. Alexnet was the first CNN to achieve such accuracy in the ImageNet challenge.

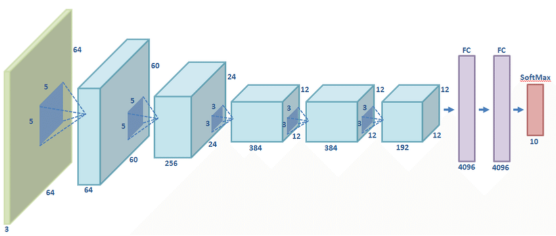

Figure 1: Structure of alexnet [13]

2. Visual Geometric Group (VGG-NET) [14] (figure 2)
   Found after AlexNet in 2014, VGG was built to find out the effects of depth on accuracy. It groups multiple convolution layers with smaller kernel size instead of having a convolutional layer with a big kernel size, this decreases the number of output features. VGG-NET has a parameter for the amount of layers which can be changed, where VGG-19 (figure 2) proves to have the best results and tradeoff [15].

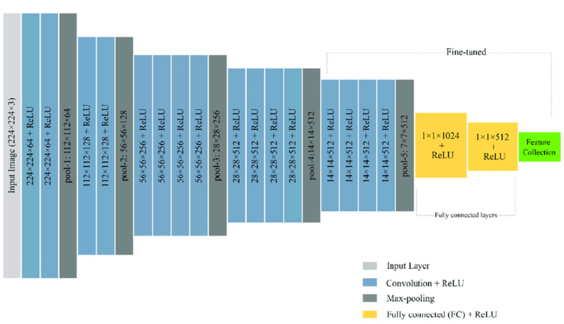

Figure 2: Structure of VGG 19 [16]

3. InceptionV3 [17] (figure 3)
   Made by Google, provides state-of-the-art accuracy of 93.3 percent while having less training parameters and floating point operations compared to its competitors. It does this by using 1x1 convolutions.

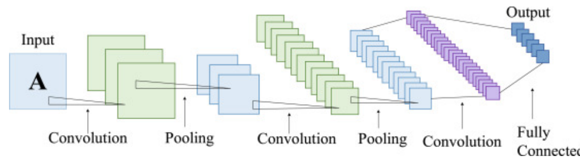

Figure 3: Structure of Inception V3 [17]

4. ResNet50 [18] (figure 4)
   Introduced in 2015. ResNet introduced residual learning[19] to the convolutional network field. This meant low loss and high accuracy while reducing the training time.

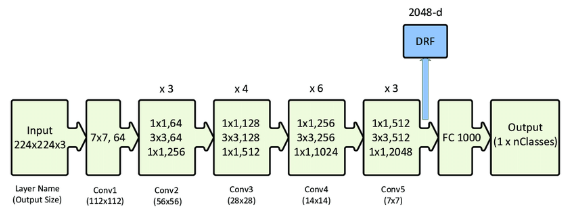

Figure 4: Architecture of Resnet50 [20]

5. EfficientNetB2 [21] (figure 5)
   In other methods, scaling eventually does not add anything to the accuracy which decreases efficiency. EfficientNet proposes a new scaling method, namely scaling depth, width and resolution uniformly by coefficients α, β, γ respectively. In this research, Efficient-B2 will be used since it proved to have state-of-the-art accuracy while being faster and smaller than other convolutional neural networks [21].

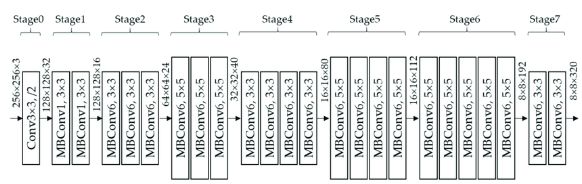

Figure 5: Architecture of EfficientNet-B2 [22]

Faster RCNN

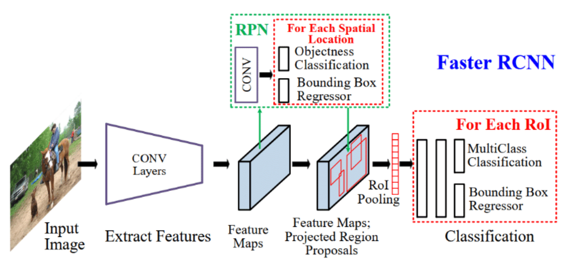

Figure 6: Faster RCNN Architecture [23]

Faster RCNN (figure 6) is a state-of-the-art object detection network introduced in 2016. It works as follows: An image is inputted into the system, where the CNN will extract the features into a feature map. Regions of interest (ROI's) are proposed for regions in the images where an object might be detected, in this case a strawberry with a maturity level. Finally, the classifier will assign a probability to each maturity level for a certain strawberry. In this research, the backbone doing the feature extraction will be tested by using different CNN's.

Faster RCNN was chosen as architecture because it was proven to offer a state-of-the-art accuracy / speed trade-off [24].

3 Methods and materials

In 3.1, the methods of the experiments are discussed. This includes the main tasks and evaluation metrics. In 3.2, the materials used in this research are explained.

3.1 Methods

In this research, two experiments will be performed to answer the research question stated in the Introduction. First, all CNN's will only be used as a classifier on segments. This method is applied to evaluate the performance of each CNN solely on classification without any FasterRCNN parameters (region of interest and anchor generators) that impact the results. SGD [5] and ADAM [6] will be used as optimizers, since these might have great impacts on the results. To prevent overfitting, early stopping is applied. To evaluate the models, top1 accuracy, mean squared error (MSE) and top3 accuracy will be used as a metric. For the best performing model, its confusion matrix and loss plot is discussed. In this experiment, cross-entropy is used as loss function.

The second experiment will use the CNN's as a backbone within FasterRCNN to see if classification can be combined with object detection. This experiment is done because in practice, the strawberries need to be detected from an image out of a strawberry greenhouse. Bounding boxes are predicted and only when the model is more than 60% sure that it is correct are passed through. To evaluate the second experiment, the focus will be on the intersection over union (IoU) of the bounding boxes to evaluate object detection. MSE is used to evaluate the classification after detection. For the best performing model, training and validation loss is discussed. For this experiment, the loss function from the original Faster RCNN paper is used [4].

3.2 Strawberry samples

Building a sample set of strawberries is an expensive process since each strawberry has to be examined by one or more experts on different aspects. Therefore, there is not much sample data available, which was one of the challenges of this research. Two sample sets will be used. Both of these sets will be split into training and test sets by using images of strawberries taken in May and June as a test set and images of strawberries taken in August as a training set to make sure there is no overlap.

Twenty percent of the training set will be used as validation. First, for training and testing the CNN's as a classifier, segments of strawberries will be used. This set of samples consists of 118 segments with three maturity scores (1-10) given by three different experts. The segments are not perfect since they can have distortion. For example, overlapping segments or not completely detected segments. This can be seen in figure 7. The data is not evenly distributed between maturity levels; a visualization of this can be seen in Figure 9.

For the second experiment, 2727 images of strawberry branches as in figure 8 were given, but each strawberry in these images needs to be matched to a segment to collect maturity data, leaving 57 samples for training and 39 samples for testing, the distribution can be seen in figure 10. All images are normalized.

(a) Perfect segment

(b) Distorted segment, you can see a green strawberry overlapping at the bottom.

Figure 7: Perfect segment and distorted segment

Figure 8: Example of an RGB image from the sample set for the FasterRCNN detection experiment

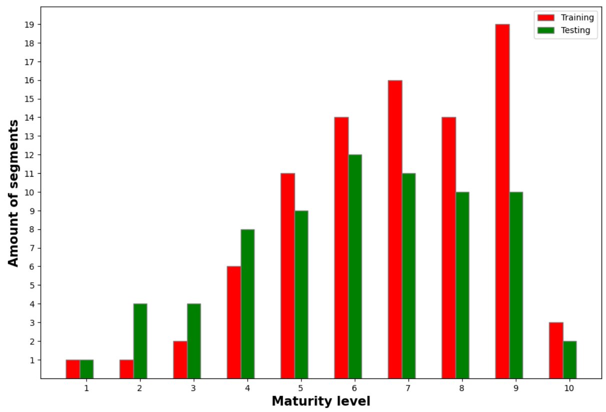

Figure 9: Distribution for the training and testing data for the segment classification experiment(experiment 1)

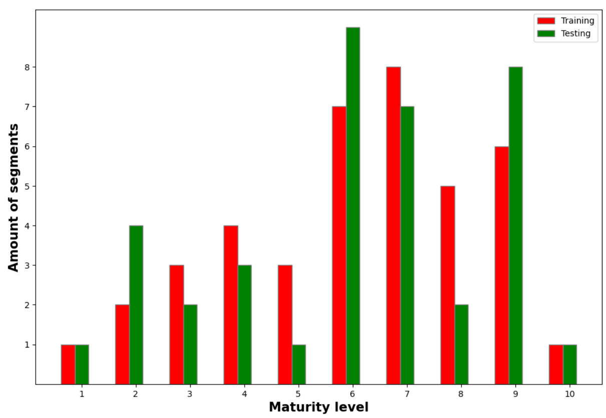

Figure 10: Distribution for the training and testing data for the direct maturity level prediction with object detection(experiment 2)

4 Experimental Setup and Results

Two experiments are done. Results from the first experiments are discussed in 4.1. For the second experiment, the results are further explained in 4.2.

4.1 CNN's as a classifier

The Pytorch library was used to implement the classifiers and the Faster RCNN architecture. First, for each of the CNN's, an extra linear layer was added to make it only output classification for 10 classes. The different CNN's were trained on the sample data for 1000 epochs on Delft Blue [25] with both stochastic gradient descent (SGD) [5] and adaptive moment estimation (ADAM) [6] as an optimizer with a learning rate of 0.001. This learning rate was chosen because its defaulted by PyTorch. As a loss function, cross entropy [7] was used. The score attached to each segment for training was determined by rounding the result of the following equation to the nearest integer:

$$label = \frac{score\_expert\_1 + score\_expert\_2 + score\_expert\_3}{3} \tag{1}$$

To prevent overfitting, early stopping is applied with a patience parameter value of 5. This means that training stops when the validation loss has not been decreasing for 5 epochs while training.

For the best performing model, the loss function is plotted over the epochs. This shows if the model is being trained well. After training, the model was ran on the full test set. A prediction is correct if it is classified as the same maturity level that had been determined by the formula above. The overall top1 accuracy of a model was determined by:

$$accuracy = \#correct/test\_set\_size \tag{2}$$

The mean squared error will also be used to evaluate performance since this takes into account the deviation of the error, so a off by one error is less penalized than a bigger deviation. The MSE is determined by:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(pred_n - actual_n)^2 \tag{3}$$

Finally, the top3 accuracy is taken into account, this percentage shows how often the correct label is in the top three predictions made by the model, this metric is chosen because its also an important metric in the ImageNET classifation challenge to evaluate the performance of CNN's.

Model	**Top1**	**MSE**	**Top3**	**Optimizer**
AlexNet	35.20%	1.56	69.00%	SGD
AlexNet	32.40%	2.05	63.38%	ADAM
EfficientNetB2	28.16%	2.00	73.23%	SGD
EfficientNetB2	22.53%	2.01	67.60%	ADAM
ResNet50	29.60%	3.22	62.00%	SGD
ResNet50	23.80%	6.53	47.88%	ADAM
VGG19	30.90%	3.43	59.15%	SGD
VGG19	21.10%	6.46	49.26%	ADAM
InceptionV2	25.35%	6.35	46.48%	SGD
InceptionV2	11.27%	7.46	45.07%	ADAM

Figure 11: Results of testing each model on the test set.

From the results in figure 11, models trained with SGD as an optimizer appear to perform better than models trained with ADAM. AlexNet with SGD as an optimizer appeared to have the best top1 accuracy and MSE. The results will be discussed more in depth with the help of its confusion matrix (figure 12) and the plot of the loss function (figure 13) during training in 6.

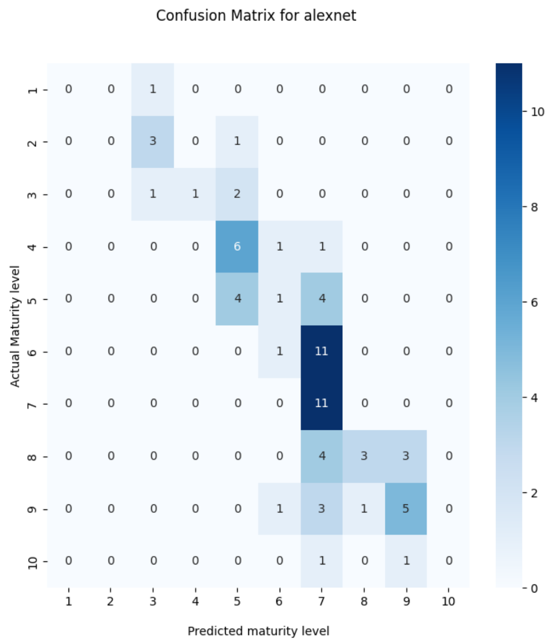

Figure 12: Confusion matrix for AlexNet with SGD as optimizer

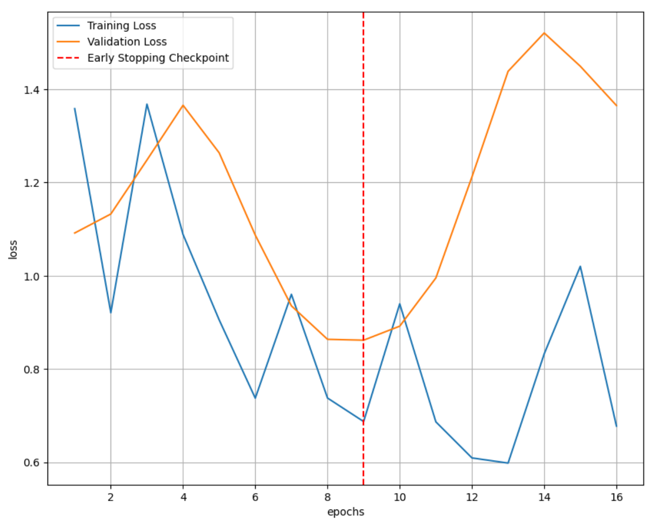

Figure 13: Validation and training loss of AlexNet with SGD as optimizer

Further discussion on the results can be found in 6.

4.2 CNN's as backbones

For the second experiment, the data is labeled in the same way as the first experiment, the model is trained on full images like in figure 8. Not only maturity classification is important in this experiment, but also detecting the strawberry. Because this task is more complex than the task of the previous experiment, the early stopping patience parameter is set to 100 over 2000 epochs.

Note that the model is trained on maturity levels directly and not first on if the object is a strawberry or not. The model outputs a probability (0-1) for each bounding box for a certain label, only bounding boxes with a probability of 0.6 or higher will be evaluated.

For this experiment, the average intersection over union (IoU) over the test set will be used as a metric for bounding box accuracy prediction. IoU is the percentage of overlap between the predicted bounding box and the ground truth bounding box (figure 14).

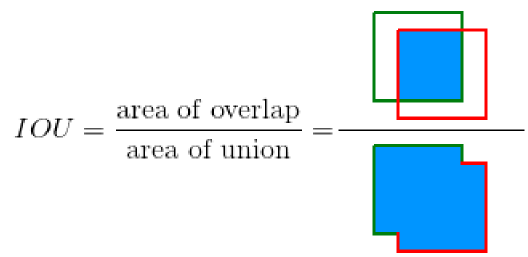

Figure 14: Illustration of intersection over union [26]

For evaluating the classification itself, Mean squared error will be used. The result can be seen in figure 15. Afterwards, the loss function for the best performing model will be discussed.

Model	MSE	Average IoU	Optimizer
AlexNet	0.00	0.00	SGD
AlexNet	0.00	0.63	ADAM
EfficientNetB2	0.00	0.00	SGD
EfficientNetB2	1.13	0.67	ADAM
ResNet50	0.33	0.04	SGD
ResNet50	2.49	0.73	ADAM
VGG19	0.00	0.00	SGD
VGG19	0.00	0.04	ADAM

Figure 15: Results of testing each model on the test set.

From the experiment, Resnet50 with ADAM as optimizer proofed to have the best performance. An average IoU above 0.5 is considered good in this experiment. For each model with a good IoU, the MSE is low (or even 0). This means that the classification part is done really well.

From the results above, it can be concluded that optimizer choice has a significant impact on the MSE and IoU. Models trained with ADAM as an optimizer have better overall results than models trained with SGD. The training loss of the best performing model (Resnet50) can be seen with both ADAM and SGD as optimizers in figure 17 and figure 16 respectively. These will be further discussed in 6.

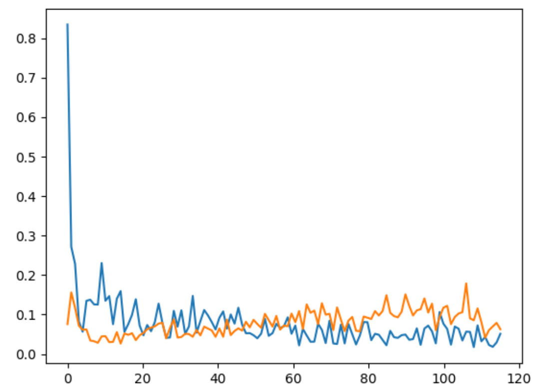

Figure 16: Validation and training loss of Resnet50 with ADAM as optimizer. The blue plot indicates training loss and the orange plot indicates validation loss.

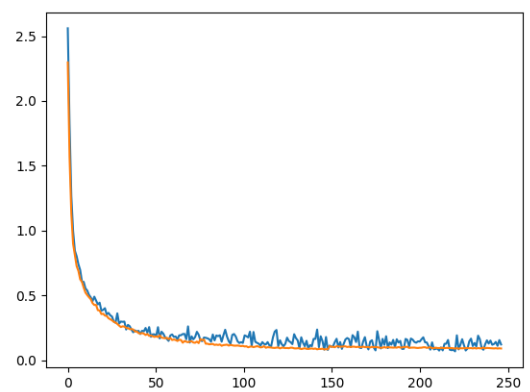

Figure 17: Validation and training loss of Resnet50 with SGD as optimizer. The blue plot indicates training loss and the orange plot indicates validation loss.

5 Responsible Research

Research and experiments involved only the person conducting the research. No surveys, interviews or external people were used to conduct the experiments and produce results. This eliminates all the ethical risks of using external people to conduct experiments. Additionally, all software used was licensed to be used in research using an Open-Source license or a license purchased from TU Delft.

Reproducing training results can be a challenge since the Delft Hyper Computing Unit was used. Training convolutional neural networks can be a costly process computation wise. Not having access to a supercomputer might make excessive training take a lot of time. All code used during this research together with the pre-trained model files can be found on github [1] [2] with instructions to reproduce testing results.

[1]https://github.com/Guthax/BEP-FasterRCNNBackbone

[2]https://github.com/Guthax/BEP

Lastly, convolutional neural networks are classified as AI. One of the ethical issues with using AI is a bias, meaning the system is trained to have a certain preference. Because the data is labeled according to the judgement of three human experts, there could be bias. This research does not have any impact on peoples lives, so a bias in the training data would not have a big impact.

6 Discussion

The confusion matrix in figure 12 shows that still a lot of ”off by one errors” are made, especially on higher maturity levels. More deviated errors occur in the lower maturity levels. An explanation for this could be the low amount of training data for those maturity levels. Levels 8 and 9 have by far the most training data available (figure 9), so it is expected that predictions in the area of high maturity levels are mostly (close to) correct. Some exceptions like maturity 7 predicted as 5 could be caused by distortion in the data like not completely detected segments or overlap like described in 3.2. The loss function in figure 13 shows that the training process of AlexNet can still be improved, since the model does not seem to converge. The early stopping algorithm does make sure the model is saved when the validation loss is the lowest.

For the second experiment, only models trained with ADAM provided good results (IoU $\geq$ 0.5 and low MSE). A possible explanation for this is when comparing the loss graphs during training (figure 16 and figure 17), the SGD model seems to be overfitting. A possible reason for the overfitting is that the validation loss has small spikes of the loss going up and down, this does not cause the early stopping to trigger which should prevent overfitting.

Previous research was done in predicting unripe, near-ripe and ripe maturity levels instead of a 1-10 scale. These methods provided high accuracy, far more than the methods in this research. There are two main reasons for this:

1. Previous research had far more training data available, which yields higher accuracy [27].
2. Past research done on different convolutional neural networks from Indonesia [10] showed that having multiple classes decreased accuracy, the number of classes in the research was 4. In this research, 10 classes were used so this could explain the low accuracy.

7 Conclusions and Future Work

In 7.1, the research and the results is concluded. In 7.2, work that can possibly continue this research is discussed.

7.1 Conclusion

The main research question was whether convolutional neural networks can be used to predict strawberry maturity levels on a scale of one to ten. To answer this question, two experiments have been performed. The first experiment was using the different CNN’s solely for classification on segments. The second experiment was to find out how these CNN’s would behave within Faster RCNN object detection to find out if this method can also be used in practice.

The results gathered determine that using CNN's to classify strawberry maturity levels on a 1-10 scale is possible, but a lot more carefully picked out and spread out training data is needed to possibly achieve high accuracy. The correct predictions from this experiment occurred mainly at high maturity levels, which had the most training data. The biggest reason for the low accuracy is that still many off-by-one errors are made. A way to boost top1 accuracy would thus to count a prediction correct when it fits one of the experts scores instead of the average.

From the models tested, AlexNet using SGD as an optimizer was found to have the best top1 accuracy of 35.2%. Although this is not a good top1 accuracy, a top3 accuracy of 69% and low MSE conclude that this method can be used to give an indication on strawberry maturity levels. For more precise predictions, more extensive training is needed. Specifically, a more balanced training set is needed, without distortions.

The results of the second experiment showed directly predicting strawberry maturity level from detection is possible. The strengths lie mostly in classification after detection. The detection part can still be improved. The best CNN to use as backbone for FasterRCNN is ResNet50 with ADAM as optimizer. This provided an MSE of 2.49 and an average intersection over union of 0.73. These results are sufficient to say that this method works. To possibly achieve state-of-the-art accuracy, more training is needed.

7.2 Future work

Using convolutional neural networks in agriculture has many possibilities. Using CNN's as a classifier is definitely possible but optimizing the training process is a difficult task. Primarily, sample data has the biggest impact on the accuracy of the classifier. A good data set is very important, this means a lot of well-made samples, which are spread over all the classes. Future research could have a more extensive data collection process, focusing on obtaining a good balanced dataset. Transformations like sheer or flips could also have an impact on overall accuracy since applying transformations gives the model extra training data. In this research, no transformations except normalization is applied. Future research could experiment with different transformations on the training set or using different color spaces than only RGB.

Future research could also focus more on using different stopping criteria to prevent overfitting instead of only early stopping (I.E weight regularization).

InceptionV2 was not applied as a backbone for FasterRCNN due to time constraints, but future research could continue on this. Also, many more convolutional neural networks exists that are not discussed in this paper. Some of these CNN's are focused on training time, which was not a requirement in this research since there was access to a supercomputer. In future research, this might not be the case so computation time could be taken into account when selecting models.

References

[1] N. Pullman, "Almost 10 per cent of strawberry crops go to waste," Oct 2017.

[2] "Food loss and waste must be reduced for greater food security and environmental sustainability," *United Nations Environment Programme*, Sep 2020.

[3] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.

[4] S. Ren, K. He, R. B. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *CoRR*, vol. abs/1506.01497, 2015.

[5] S. ichi Amari, "Backpropagation and stochastic gradient descent method," *Neurocomputing*, vol. 5, no. 4, pp. 185–196, 1993.

[6] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization,"

[7] P.-T. D. Boer, D. P. Kroese, and R. Y. Rubinstein, "A tutorial on the cross-entropy method," 2005.

[8] C. Liu, W. Liu, X. Lu, F. Ma, and W. Chen, "Application of multispectral imaging to determine quality attributes and ripeness stage in strawberry fruit," *PLoS ONE*, vol. 9, p. 87818, 2014.

[9] Z. Gao, Y. Shao, G. Xuan, Y. Wang, Y. Liu, and X. Han, "Real-time hyperspectral imaging for the in-field estimation of strawberry ripeness with deep learning," *Artificial Intelligence in Agriculture*, vol. 4, pp. 31–38, 2020.

[10] R. Sustika, A. Subekti, H. Pardede, E. Suryawati, O. Mahendra, and S. Yuwana, "Evaluation of deep convolutional neural network architectures for strawberry quality inspection," *International Journal of Engineering and Technology(UAE)*, vol. 7, pp. 75–80, 01 2018.

[11] M. Rodriguez, F. Pastor, and W. Ugarte, "Classification of fruit ripeness grades using a convolutional neural network and data augmentation,"

[12] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks,"

[13] J. Llamas, P. M. Lerones, R. Medina, E. Zalama, and J. Gómez-García-Bermejo, "Classification of architectural heritage images using deep learning techniques," *Applied Sciences*, vol. 7, no. 10, 2017.

[14] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2015.

[15] M. Shaha and M. Pawar, "Transfer learning for image classification," in *2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, pp. 656–660, 2018.

[16] R. Mostafiz, M. M. Rahman, A. Islam, and S. Belkasim, "machine learning knowledge extraction focal liver lesion detection in ultrasound image using deep feature fusions and super resolution," *Machine Learning and Knowledge Extraction*, vol. 2, 07 2020.

[17] C. Flores Vega, D. Achanccaray, and J. Andreu-Perez, "A convolutional neural network approach for a p300-based brain-computer interface for disabled and healthy subjects," 09 2018.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015.

[19] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," 2017.

[20] A. Mahmood, A. Giraldo, M. Bennamoun, S. An, F. Sohel, F. Boussaid, R. Hovey, R. Fisher, and G. Kendrick, "Automatic hierarchical classification of kelps using deep residual features," *Sensors*, vol. 20, p. 447, 01 2020.

[21] M. Tan and Q. V. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," *CoRR*, vol. abs/1905.11946, 2019.

[22] Y. Jie, X. Ji, A. Yue, J. Chen, Y. Deng, J. Chen, and Y. Zhang, "Combined multi-layer feature fusion and edge detection method for distributed photovoltaic power station identification," *Energies*, vol. 13, p. 6742, 12 2020.

[23] J. Duan and C. J. Kuo, "Bridging gap between image pixels and semantics via supervision: A survey," *CoRR*, vol. abs/2107.13757, 2021.

[24] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, and K. Murphy, "Speed/accuracy trade-offs for modern convolutional object detectors," 2017.

[25] Delft High Performance Computing Centre (DHPC), "DelftBlue Supercomputer (Phase 1)." https://www.tudelft.nl/dhpc/ark:/44463/DelftBluePhase1, 2022.

[26] R. Padilla, S. Netto, and E. da Silva, "A survey on performance metrics for object-detection algorithms," 07 2020.

[27] C. Chu, A.-L. Hsu, K.-H. Chou, P. Bandettini, and C. Lin, "Does feature selection improve classification accuracy? impact of sample size and feature selection on classification using anatomical magnetic resonance images," *NeuroImage*, vol. 60, no. 1, pp. 59–70, 2012.