NHL STENDEN
computer vision & data science

Detecting Bioplastic Objects Using Instance Segmentation and Hyperspectral Imaging

NHL Stenden Centre of Expertise in Computer Vision & Data Science

Esmeralda Willemsen
Supervisors: Klaas Dijkstra, Willem Dijkstra

Abstract—Plastic is being produced at an ever-increasing rate, mostly for short-lived products that end up polluting the environment. Bioplastics and recycling can play an important role in mitigating this problem. Improving the speed and/or accuracy of plastic sorting mechanisms is imperative. In this paper, the detection of plastic objects using instance segmentation and hyperspectral imaging is explored. State-of-the-art instance segmentation model Mask R-CNN is used, with 30 hyperspectral images of PE, PP and PET flakes as input. These types of models are typically used for regular RGB images, so dimensionality reduction is required. This dimensionality reduction of the images is done by an added convolutional layer that precedes the Mask R-CNN model. Excluding outliers, the average result over 20 runs of 100 epochs is an instance segmentation mask mAP (IoU 0.5:0.95) of 0.8261. Including outliers, mAP is 0.7057. Interestingly, adding a second convolutional layer for dimensionality reduction does not lead to improved results. A positive finding is that using hyper-hue dimensions, the output of pre-processing step HHSI, appears to work well as a way of separating the three plastics. Future research could shed light on the relation between hyper-hue dimensions and spectral bands. The hyperspectral bands that contribute the most to the dimensionality reduction process, appear to be the ones that show large absolute differences in average relative reflection between the three plastics. Overall, this paper shows that hyperspectral images in combination with instance segmentation framework Mask R-CNN and the use of a convolutional layer for dimensionality reduction work well for detecting different types of plastic flakes. Several aspects, such as the use of convolutional neural networks for dimensionality reduction, deserve further study.

Index Terms—Hyperspectral imaging, instance segmentation, dimensionality reduction, plastic sorting

1 Introduction

In this day and age, a world without plastic is unimaginable. Plastics have many unique properties: they are inexpensive, lightweight, strong and durable. Unsurprisingly, with an ever expanding population and our standard of living continuously improving, plastic production has increased from 0.5 to 260 million tons per year [1]. Unfortunately, two of the aforementioned advantages of plastics, being lightweight and being durable, also make plastics a significant environmental hazard. The majority of the plastic produced is used to make disposable packaging items or short-lived products that are permanently discarded within a year of manufacture. This is problematic because plastic that ends up in the environment is hazardous for all the organisms that reside there.

Bioplastics play a very important role in solving the plastic pollution problem. A bioplastic is defined as either made from renewable resources (bio-based), biodegradable, or both [2]. Examples of bio-based, non-biodegradable plastics are bio-PE, bio-PP and bio-PET. Examples of plastics that are both bio-based and biodegradable are PLA (polylactic acid), PHA (polyhydroxyalkanoate) and PBS (polybutylensuccinate).

With the amount of plastic being produced only increasing, it is important that the plastic recycling rate can keep up. However, plastic recycling is still limited compared to most other bulk materials. The plastic recycling rate worldwide was about 18% in 2018 [3]. Fortunately, the international market for recycled plastics is developing and recovery and recycling rates for plastics are increasing all around the world. However, further improvements are still necessary. To further increase the recycling rate, it is imperative that the sorting of the plastics can be executed both correctly and quickly. Plastic sorting mechanisms have to be capable of sorting all the different plastic types, including bioplastics. The main goal of this research is to determine to what extent object detection can be used to sort various bioplastics. Both plastic flakes and objects will be used in the research. Due to the fact that the amount of bioplastic objects that is available is limited, flakes will be used in the first stages of the research.

- *Esmeralda Willemsen is a Mathematical Engineering student at the NHL Stenden University of Applied Sciences, E-mail: esmeralda.willemsen@student.nhlstenden.nl.*
- *Klaas Dijkstra is a senior researcher at the NHL Stenden Centre of Expertise in Computer Vision & Data Science, E-mail: klaas.dijkstra@nhlstenden.nl.*
- *Willem Dijkstra is a researcher at the NHL Stenden Centre of Expertise in Computer Vision & Data Science, E-mail: willem.dijkstra@nhlstenden.nl.*

1.1 Research questions

To achieve the main goal of this research, three research questions must be answered. The research questions are as follows:

- How can we detect different types of regular plastic flakes?
- How can we detect different types of bioplastic flakes?
- How can we detect different types of bioplastic objects?

1.2 State of the art

Over the years, approaches to the problem of plastic sorting have become increasingly complex. Bonifazi, Capobianco, and Serranti (2018) describe using Partial Least-Squares Discriminant Analysis (PLS-DA), a machine learning tool that is a useful feature selector and classifier, to recognize different polymer flakes [4]. Deep learning is a subset of machine learning and is used in state-of-the-art plastic sorting research. Machine learning algorithms and tools such as PLS-DA are linear in nature. Deep learning algorithms such as Artificial Neural Networks (ANNs) are more complex in comparison. They are able to effectively address non-linear problems [5].

Previous research has shown that ANNs in combination with hyperspectral imaging can be used to successfully distinguish between different types and shapes of plastics and thereby aid the sorting process [6]. In this section the state of the art of artificial neural networks, hyperspectral imaging, plastic sorting and various detection methods and models will be discussed.

1.2.1 Artificial Neural Networks

The way artificial neural networks work is inspired by the way the human brain functions. An Artificial Neural Network (ANN) is an information processing model which consists of multiple processors that work simultaneously to model systems with an intricate relationship between input and output. The structure of a neural network consists of an input layer, multiple (at least one) hidden layers and an output layer. The neural network improves by training.

When training, the network makes initial predictions and calculates the loss function which indicates how good the predictions are. The neural network minimizes the loss function with an optimizer function and as a result, the predictions of the network will become more accurate. Problems concerning image classification and object detection can be solved with the help of a specific type of ANN, namely convolutional neural networks (CNNs) [7].

A CNN consists of one or more convolutional layers and pooling layers. A schematic visualization is shown in Figure 1. Subsequently, there may be one or more fully-connected layers. They take all the neurons in the previous layer and connect them to every single neuron of the current layer to generate global semantic information. It is not always necessary for a CNN to contain a fully connected layer, it can be replaced by a 1 x 1 convolutional layer [8]. The convolutional layers will determine the output of neurons which are connected to local regions of the input [9]. When training the CNN, the loss is calculated using the loss function. The loss gives an indication of how well the model is performing by comparing the output predictions with the ground truths. Gradient descent optimization is often used to minimize the loss. The gradient of the loss function tells us in which direction the loss function has the steepest rate of increase. Every weight is then updated in the negative direction of the gradient. The gradient is the multi-variable derivative of the loss function with respect to all the weights of the network [10].

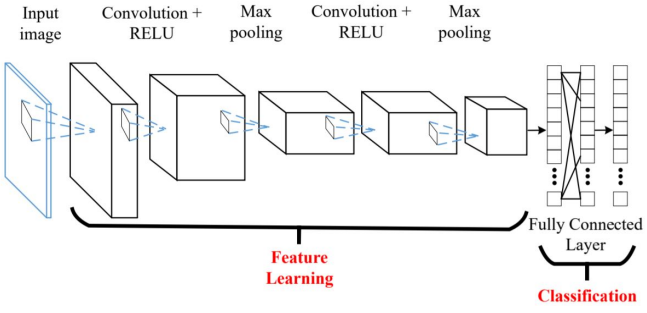

Fig. 1. Simplified schematic of a convolutional neural network [7].

1.2.2 Hyperspectral imaging

In contrast to human vision which covers three bands in the visible light spectrum (400 nm - 700 nm), hyperspectral imaging can cover several hundred bands of light [11]. Each individual pixel in a hyperspectral image contains the spectrum of that specific position. Hyperspectral images have the advantage of distinguishing subtle spectral differences and are therefore widely used [12]. Hyperspectral images are also called hyperspectral cubes. These cubes are three-dimensional: the x and y-axis correspond to the spatial dimensions and the third axis represents the wavelength dimension [13]. This is illustrated in Figure 2.

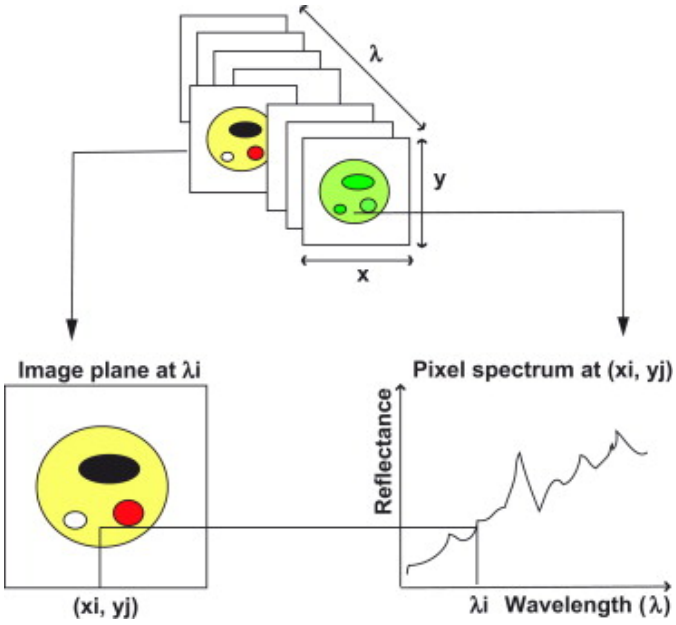

Fig. 2. Schematic representation of a hyperspectral cube showing the relationship between spectral and spatial dimensions [13].

Although hyperspectral images are very suitable for classifying materials of interest (in this case plastics) on the basis of their unique spectral characteristics, the sheer amount of information makes this a challenge (the so-called curse of dimensionality). Reducing the dimensionality of the images solves this problem. This does not lead to considerable information loss, as high-dimensional feature spaces are nearly empty, and it offers multiple benefits such as faster computations [14]. Another reason for dimensionality reduction is the fact that it is required if the images are to be used as input in an instance segmentation model such as Mask R-CNN (see 1.2.5. and 2.5.), which requires images to have three bands (dimensions). The dimensionality of data (spectral or spatial) can be reduced by feature extraction. The aim is to reduce dimensionality while maximizing separability between categories. With respect to shallow feature extraction techniques, Principal Component Analysis (PCA) is regarded as the most widely used technique. Using CNNs is one of the most popular choices for feature extraction when using deep learning models [14].

1.2.3 Plastic sorting

The sorting of plastic during the recycling process is currently predominantly done by the use of near-infrared (NIR) spectroscopy. Infrared cameras are used to scan the plastics that pass on a conveyor belt below the camera. NIR uses the wavelength signature of the different plastics to distinguish among them. Each type of plastic will have a unique response to the infrared light, thus the wavelength signature can be used in the process of identifying the plastic and consequently sorting the plastics correctly [15].

1.2.4 Object detection

In previous research different types of plastic have been successfully classified using semantic segmentation. Here, every pixel is assigned to a semantic class. Object detection classifies objects and precisely localizes them [16]. Object detection aims to model bounding boxes around every object in an image, even if they overlap. Object detection can be described as a two-step process. The first step is to find bounding boxes such that each box contains only one object, these bounding boxes are called region proposals. The second step is to then classify the image inside every bounding box in the image and

assign it a label. Previous research has shown that by using hyperspectral imaging and semantic segmentation, different types of plastic can be successfully classified. Object detection can be potentially beneficial to accurately sorting plastic since the location of the plastics is taken into account as well. If a variety of (bio)plastic objects pass on a conveyor belt under a hyperspectral camera, they first have to be classified correctly. The next step is to move the plastics to the correct belt, bin or tray. If the location of the (bio)plastic objects is known, the sorting mechanism can sort the objects correctly based on their location. For example, if an object in the bottom-left corner is identified as PLA, the mechanism can make sure it ends up in the PLA belt, bin or tray.

1.2.5 Instance segmentation

Instance segmentation goes one step further than object detection. Instance segmentation combines elements from object detection and semantic segmentation. The difference between object detection, semantic segmentation and instance segmentation is visualized in Figure 3. As explained above, the goal of object detection is to classify individual objects and determine their location using bounding boxes. The goal of semantic segmentation is to classify each pixel in the image without differentiating object instances. Using instance segmentation both of these goals can be achieved with the same method [17]. When developing a method that sorts (bio)plastics accurately, it is advantageous to use instance segmentation compared to object detection. If the sorting mechanism only knows the bounding box of an object, the orientation of the object within the bounding box is still unknown. This is important information in the sorting process.

(a) Object Detection (b) Semantic Segmentation (c) Instance Segmentation

Fig. 3. Object detection (a) localises the different people, but at a coarse, bounding-box level. Semantic segmentation (b) labels every pixel, but has no notion of instances. Instance segmentation (c) labels each pixel of each person uniquely [18].

1.2.6 Detection frameworks

Convolutional Neural Networks (CNNs) are the state of the art for object detection in images. A number of object detection models have been developed based on CNNs. They fall into two categories: two-stage approach and one-stage approach. In two-stage approach models region or object proposals are generated first, and classification and detection happens next, using bounding boxes. One-stage models do not have an intermediate region proposal stage. These models are faster, but less accurate. The first two-stage model was R-CNN [19]. It was followed by improved models such as Fast R-CNN [20], Faster R-CNN [21], FPN [22] and Mask R-CNN [17]. The latter is an instance segmentation model. When comparing these models, in terms of accuracy, Mask R-CNN outperforms the rest followed by Faster R-CNN [12]. Examples of one-stage object detection models are YOLOv3 [23], SSD [24], YOLOv2 [25], RetinaNet [26] and RefineDet [27]. Research has shown that compared to one-stage models two-stage models are more accurate but slower [28]. A new family of detectors, called EfficientDet, has recently been developed that achieves state-of-the-art accuracy with fewer parameters than previous object detection and semantic segmentation models [29].

2 Materials and Methods

In this section the materials and methods are described that are used in this research.

2.1 Hardware

The research was conducted on a virtual machine (VM). The specifications of the VM are shown in Table 1.

CPU:	12 cores
RAM:	14.75 GiB
GPU:	1x NVIDIA RTX 2070 (8GB)
OS:	Debian 10.1 "Buster"

Table 1. Specifications of the VM.

2.2 Camera

In this research, hyperspectral images are used. The images are made with the Specim FX17, a near-infrared (900-1700nm) hyperspectral camera suitable for applications in many fields such as recycling and chemical imaging. The Specim FX17 is a line-scan camera. This means that the camera captures a (hyperspectral) image of one line at a time. By moving the object being photographed underneath the camera, the image of the entire object is obtained. The setup used in this research is shown in Figure 4. The camera has a frame rate of 670 frames per second (full range, 224 spectral bands) to 15,000 frames per second (4 spectral bands selected) [13].

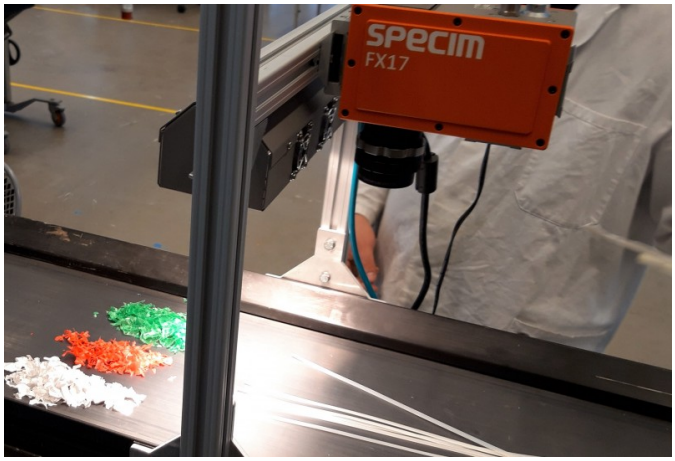

Fig. 4. Setup Specim FX17. The plastic samples are moved underneath the camera via a conveyor belt.

2.3 Datasets

Due to circumstances, new datasets could not be created during this research[1]. The dataset used for this research is a subset of an existing dataset that was created for an earlier project concerning object detection of plastic flakes [30]. There are three polymer classes shown in the images: PE, PP and PET. The dataset consists of 9 images of regular plastic polymers. Only one type of plastic is depicted on each of the images. The original images made by the Specim FX17 have a size of 640 x 640 pixels. Two examples of original images are shown in Figure 5. In these images there are 25 heaps of polymer flakes per image. The 25 heaps are considered 25 objects in this research. They will be divided into 25 images that are used separately. This results in 225 (9×25) images of 128×128 pixels total.

[1] Due to the COVID-19 pandemic taking place during this research, there was no access to the Specim FX17 and as a result, images of bioplastic flakes/objects could not be obtained.

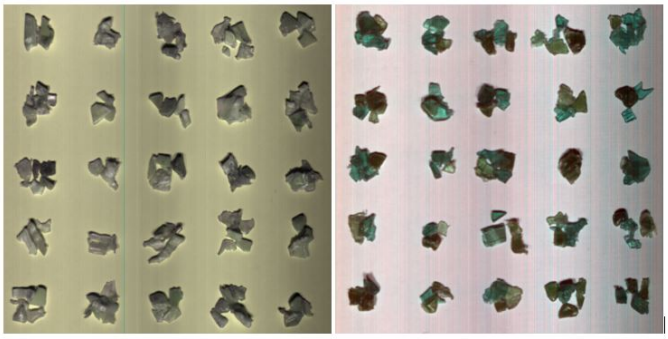

Fig. 5. Examples of images taken by the FX17. Each image shows 25 heaps of polymer flakes. Every heap is considered an object.

Ground truth masks are created for this existing dataset. An example of an original image and its ground truth mask is shown in Figure 6. The ground truth masks play an important role in the learning process of neural networks. Because the quality of the ground truth masks of this dataset varies, only a subset of this dataset is used containing the images for which accurate masks were created. This subset is divided into one dataset for training containing 24 images (heaps of flakes) and one dataset for testing containing 6 images. The way the 30 images are split over the two datasets is set to make sure that there are no images in the test set that the model trained on. Furthermore the division is set so that there are equal amounts of the 3 plastic types in the training set. The training set contains 8 images per plastic type and the testing set contains 2 images per plastic type.

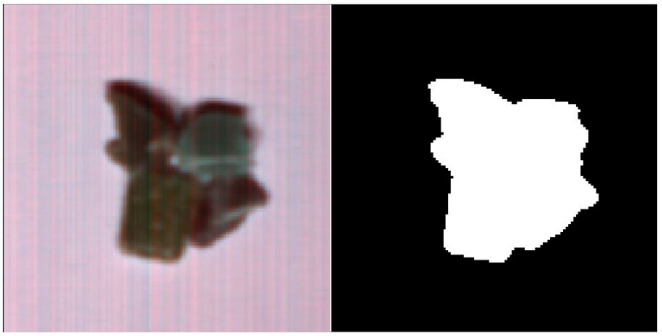

Fig. 6. Example of an image and its ground truth mask.

2.4 Preprocessing

In this section preprocessing methods that will be used in this research will be discussed. These methods are used in order to optimize the results of the model. In general, the goal of preprocessing is to obtain uncontaminated data for further processing [31].

2.4.1 Flat-field correction

The output of the Specim FX17 consists of RAW files and HDR files. These files are converted to arrays in order to apply a flat-field correction. This correction mitigates imperfections in the images. These imperfections can be caused by, for example, uneven illumination or dust on the lens and the sensitivity of the camera sensor for each spectral band. This affects the intensity values of the pixels. The correction, which results in the relative reflection image, takes the form of Equation 1. It should be applied to all wavelengths of the imaging system.

$$Rs(\lambda) = \frac{Is(\lambda) - Id(\lambda)}{Ir(\lambda) - Id(\lambda)} \times Rr(\lambda) \quad (1)$$

In this equation, Is is the intensity image of the sample itself, Ir is a reference image obtained from a white panel, and Id is the dark-reference image, collected by turning the light off and covering the lens. $Rr(\lambda)$ is the reflectance factor of the white panel [32]. In this research, the reflectance factor of the white panel is assumed to be equal to 1. The effect of the flat-field correction can be seen in Figure 7 which shows one specific band of the original hyperspectral cube before and after applying the flat-field correction. Looking at the images, the before image shown on the left has visible vertical lines in the background. The after image on the right has an even background colour.

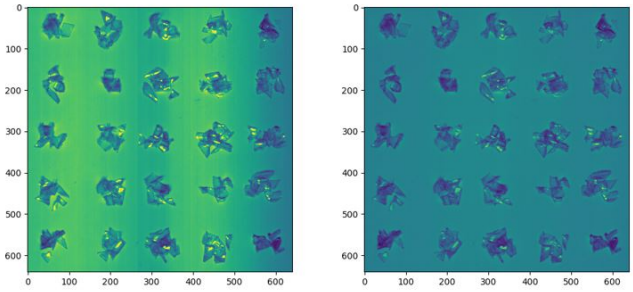

Fig. 7. Before (left) and after (right) images of applying the flat-field correction.

2.4.2 Hyper-Hue-Saturation-Intensity

A second image processing technique that can be applied to hyperspectral images uses the hyper-hue-saturation-intensity (HHSI) color space. It is a method that can be used when dealing with directional lighting that causes intensity variations due to specular reflectance. The problem is that this can misguide the classification algorithm. The HHSI method separates colour (hyper-hue) from saturation and intensity (from which it is independent). Visual inspection of the dataset used in this research indicates that the images show intensity variations due to specular reflectance. Hence, application of the HHSI method was deemed useful. When applying the HHSI method, a color space called HHSI is derived from an n-dimensional image. This color space is composed of hyper-hue, saturation and intensity. When only hyper-hue is used for classification, and both saturation and intensity are not used, the results of this classification improve. The hyper-hue is represented by vectors in (n-1) dimensions. A big advantage is that there is no loss of information when images are transformed to a HHSI high-dimensional colour space [33]. Figure 8 shows the HHSI method applied to one of the images used in this research.

2.5 Mask R-CNN

In this section the framework used in the research is described. As explained in the introduction, instance segmentation models provide more information than either semantic segmentation or object detection models do. The output of instance segmentation models covers both the classification and mask of an object. As this output is deemed very useful for sorting plastic objects, it was decided to use an instance segmentation model. Mask R-CNN is a state-of-the-art framework for instance segmentation and is therefore used in this research. The framework has a two-stage pipeline. The basic structure is illustrated in Figure 9. The first stage is a Region Proposal Network (RPN), which is identical to that of Faster R-CNN and produces candidate object bounding boxes. There is a choice of different backbones that can be used. Research has shown that using a ResNet-FPN backbone gives very good results [17]. Therefore it was decided to use a ResNet-FPN backbone, ResNet-50-FPN. In the second stage, classification and bounding-box regression takes place and, in parallel, Mask R-CNN predicts a binary mask for each RoI (Region of Interest). The state-of-the-art results of Mask R-CNN are

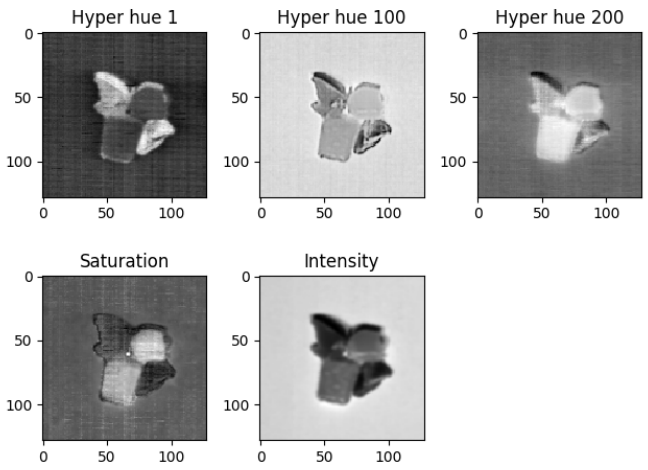

Fig. 8. HHSI method applied to image from dataset.

partly caused by the use of a special layer, RoIAlign, that ensures pixel-for-pixel alignment between inputs and outputs. This layer avoids the double quantization for feature extraction that takes place in RoIPool, a standard operation used in Mask R-CNNs predecessor, Faster R-CNN. Quantization causes misalignment between inputs and outputs. Faster-R CNN was never designed for pixel-to-pixel alignment, but for Mask R-CNN, which predicts object masks, this is a different matter. In order to precisely align extracted features with input data, the model uses bilinear interpolation to compute feature map values [17].

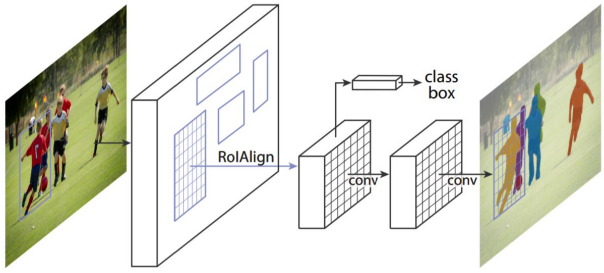

Fig. 9. The Mask R-CNN framework for instance segmentation [17].

2.6 Statistical tests

In one of the experiments, two sets of data are compared. When comparing the results of two groups (or sets of data), it is necessary to perform a statistical test. The first step in this process is to determine which statistical test is suitable for the problem. A well-known test for comparing two independent groups on the same dependent variable is the independent t-test. Before performing the t-test it needs to be verified that certain assumptions are met. One of these assumptions is that the data is normally distributed [34].

There are several statistical tests for assessing normality. The Shapiro-Wilk test is a good choice, as research has shown that this is a particularly powerful test [35]. The test is especially suitable for small samples (< 50) [34].

When the data is not normally distributed and the samples are small, the non-parametric Mann-Whitney U-test can be used to compare two groups instead of the independent t-test [36].The Mann-Whitney U test is also robust when the data contains outliers [37]. The null and alternative hypotheses for the Mann-Whitney U test are as follows:

H_0: The distributions of the two groups are the same.
H_a: The distributions are different.

The Mann-Whitney U test is based on a simple principle. First, a combined ranking is made of all the datapoints. If the first part of this ranking is mostly made up of datapoints from one group and the last part consists mostly of datapoints from the second group, the odds are they come from populations with different averages. The way to compare how the groups are spread across the ranking is by adding up all the ranks per group and then comparing the sums. If the rank sums are close to being equal, the datapoints from the groups are mixed together quite well. If the rank sums differ a lot, the data is not mixed well and the populations differ. Momeni, Pincus and Libien mention two assumptions for performing the Mann-Whitney U test. The first is that the dependent variable has a scale. This is essential because when ranking the data, it needs to be clear which is greater and which is smaller. The second assumption is that the observations are independent [38]. The general formula that computes the Mann-Whitney U statistic for the first group is as stated in Equation 2.

$$U_1 = n_1 n_2 + \frac{n_1(n_1+1))}{2} - T_1 \tag{2}$$

With the following variables:
T_1 = the sum of the ranks of group 1
n_1 = the number of datapoints from group 1
n_2 = the number of datapoints from group 2

The U-score is the difference between maximum possible rank sum and the actual ranks summed. The U-score for the second group can be computed with the same formula but with the indices 1 and 2 switched. A bigger U score means that the members (datapoints) of that group are more in the top part of the ranking, whereas a smaller U score means that the datapoints are more in the lower ranking. Performing a Mann-Whitney U test always generates two U scores, one for each group. The lower of the two U scores is subsequently compared to the critical U value. If the U statistic is lower than the critical U value, the null hypothesis is rejected [39].

2.7 Evaluation metrics

The performance of the framework used in this research, Mask R-CNN is evaluated by the standard metrics of the popular dataset COCO [40]. The most important of these metrics are described in this section. The two main tasks performed by Mask R-CNN are object detection and instance segmentation. The performance metrics related to these tasks indicate how well the predictions of the model match the ground truth.

With regard to object detection, Mask R-CNN outputs bounding box, class, and confidence score. A prediction is called a True Positive (TP) if the predicted class corresponds with the ground truth, and the IoU (Intersection over Union) between the predicted bounding box and the ground truth box exceeds a certain threshold. When a prediction does not fulfil either or both of these conditions, it is called a False Positive (FP) [12]. In this case, the prediction does not match a ground truth. A False Negative (FN) is a ground truth not predicted by the model [41]. The IoU is defined as the area of the intersection divided by the area of the union of a predicted bounding box and a ground truth bounding box. Equation 3 shows how the IoU between two areas A and B is computed. An IoU threshold of, for example, 0.50 means that the predicted box is considered a TP if the IoU is greater than 0.50.

$$IoU(A,B) = \frac{|A \cap B|}{|A \cup B|} \tag{3}$$

The performance of the model is often expressed by mean Average Precision (mAP). Before explaining this metric, it is necessary to define precision and recall. Precision is the ratio between the number of correct detections and total detections. This is shown in Equation 4.

$$precision = \frac{TP}{TP+FP} \tag{4}$$

Recall is the ratio between the number of correct detections and the number of ground truth objects to be detected [12]. This is shown in Equation 5.

$$recall = \frac{TP}{TP+FN} \tag{5}$$

Often it is not desirable to regard less confident predictions as true positives. For this reason it is common practice to set a confidence threshold below which predictions are not considered True Positives even if they fulfil the conditions described earlier. Precision and recall are often visualized in the so-called precision-recall curve, which shows the precision and recall at each threshold of confidence [41]. The COCO evaluation metrics precision (and recall) are computed for 101 confidence thresholds between 0 and 1 with step size 0.01 and for all the classes. All these values for precision are stored in a multidimensional array. When these values are averaged over the 101 confidence thresholds, the area under the precision-recall curve is calculated and corresponds to the metric Average Precision (AP). If all of the values are averaged over the 101 confidence thresholds and over all of the classes, the metric mean Average Precision (mAP) is calculated.

The COCO metrics offer various options with regard to the IoU thresholds. There are three different options for calculating mAP: using IoU threshold 0.5, using IoU threshold 0.75 and using multiple IoU thresholds. When using multiple IoU thresholds, the mAP is calculated using 10 IoU thresholds between 0.50 and 0.95 with a step size of 0.05, which are subsequently averaged.

The above described process for calculating evaluation metrics for object detection can also be executed for instance segmentation. The only difference lies in the computation of the IoU, which is calculated for masks instead of bounding boxes.

3 Experiments & Results

This section describes the experiments that are conducted in this research. The dataset to be used in all experiments is the one described in the previous section, consisting of hyperspectral images of PE, PP or PET plastic flakes.

3.1 Experiment 1: Assessing the performance of Mask R-CNN

The first experiment entails using Mask R-CNN for detecting flakes of PE, PP and PET on hyperspectral images. The experiment relates to the first research question ('How can we detect different types of regular plastic flakes?'). Use is made of the preprocessing techniques flat-field correction and the HHSI method as described in 2.4. The dimensionality of the images is reduced to 3 to make the images suitable as input for the model. This is done by adding one convolutional layer, with kernel size 5×5, preceding the model. When running the model, this layer is included in the training loop.

For this first experiment, the model is run 20 times. Each run consists of 100 epochs. After experimenting with the number of epochs, it became apparent that the metric mean Average Precision (for bounding boxes as well as instance segmentation) shows no further improvement after 30 to 70 epochs. Thus, it was concluded that 100 epochs would suffice. As optimizer Stochastic Gradient Descent (SGD) is used. Since this optimizer is contained in the source code of Mask R-CNN, it is assumed to be the best choice for this experiment. The default parameters are kept unchanged, since the selection of appropriate learning parameters requires considerable experience, according to Liu et al [12].

During each run, the model outputs the updated values of various parameters (such as mAP and mAR) after each epoch. After training the model, the predicted masks for the original images can be generated and visualized. An example of such a mask and the corresponding original hyperspectral image is depicted in Figure 10.

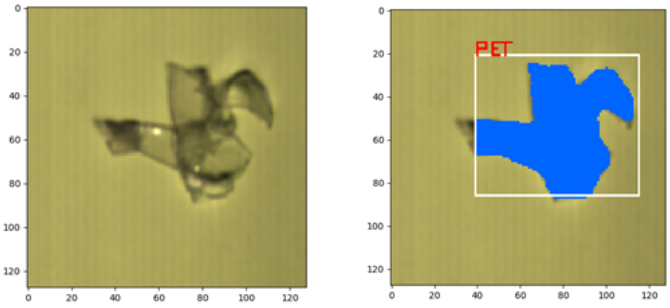

Fig. 10. Example of original image vs predicted mask image.

Table 2 shows the results of this experiment. The results are averaged over all 20 runs. As there were a couple of outliers, the results are shown both with and without them. The outliers are computed using the interquartile range (IQR).

	mAP instance segmentation		mAP bounding box	
	average	**standard deviation**	**average**	**standard deviation**
IoU: 0.5:0.95	0.7057	0.2562	0.6718	0.2208
IoU: 0.5	0.8863	0.1986	0.9222	0.1411
IoU: 0.75	0.8154	0.3306	0.7676	0.3619
Without outliers:				
IoU: 0.5:0.95	0.8261	0.0597	0.7724	0.0836
IoU: 0.5	0.9741	0.0495	0.9741	0.0495
IoU: 0.75	0.9741	0.0495	0.9351	0.1275

Table 2. Performance metrics Mask R-CNN.

3.2 Experiment 2: Comparing 1 and 2 convolutional layers for dimensionality reduction

In order to answer the second and third research question ('How can we detect different types of bioplastic flakes/objects?'), hyperspectral images of bioplastics are needed. Because these images were not available[1], two more experiments related to the first research question have been designed.

Experiment 2 involves a repetition of the first experiment, 20 runs of Mask R-CNN, again using the dataset described in 2.3., with two convolutional layers for dimensionality reduction instead of one. With two convolutional layers, dimensionality reduction takes place in two steps: from 223 to 112 and from 112 to 3. It is interesting to find out whether this leads to a significantly different result.

Statistically, this equates to the question whether the results from the 20 runs with one convolutional layer and those from the runs with two convolutional layers have the same distribution. In this experiment, we focus on comparing instance segmentation mAP, the mean average precision of the mask prediction (IoU: 0.5:0.95). The results of the performance metrics (mAP instance segmentation and mAP bounding box) are summarized in Table 3. The mAP instance

segmentation results (not averaged over 20 runs) for both one and two convolutional layers, are attached in Appendix A. The null hypothesis and alternative hypothesis are as follows:

H_0: The distributions of the two groups are the same.
H_a: The distributions are different.

	mAP instance segmentation		mAP bounding box	
	average	**standard deviation**	**average**	**standard deviation**
IoU: 0.5:0.95	0.7333	0.2652	0.6829	0.2446
IoU: 0.5	0.8970	0.2206	0.9258	0.1546
IoU: 0.75	0.8404	0.3544	0.7893	0.3532
Without outliers:				
IoU: 0.5:0.95	0.8435	0.0417	0.7792	0.0915
IoU: 0.5	0.9886	0.0342	0.9886	0.0342
IoU: 0.75	0.9886	0.0342	0.9886	0.1321

Table 3. Performance metrics Mask R-CNN. Dimensionality reduction by 2 convolutional layers.

Performing the Shapiro-Wilk test for normality of the data, executed in SPSS, shows that the data is not normally distributed, thus an independent t-test on the results cannot be trusted to give valid results. The output of the Shapiro-Wilk test is shown in Table 4. The values in the Sig. column show the p-value which has to be above 0.05 (for a significance level of 95%) for the data to be considered normally distributed. Since the p- value for both groups is far below 0.05, it can be concluded that the data does not follow a normal distribution.

Shapiro-Wilk			
Nr of convolutional layers	**Statistic**	**df**	**Sig.**
1	0.680	20	0.000
2	0.556	20	0.000

Table 4. SPSS output Shapiro-Wilk test for normality

When performing the Shapiro-Wilk test in SPSS, the program also generates graphs which indicate whether data is normally distributed or not. These graphs are called Q-Q plots. A Q-Q plot is used to compare a sample distribution to a theoretical distribution [42]. The theoretical distribution in this case is the normal distribution and the sample distribution is related to this experiment. Visual inspection of the Q-Q plots, shown in Figure 14 and Figure 15 in Appendix B, shows that the data for both groups deviates from the normal distribution, which is displayed as the diagonal line. This confirms the earlier described result of the Shapiro-Wilk test.

In view of the result of the Shapiro-Wilk test, it can be concluded that the independent t-test may not be performed on the data. Instead, the Mann-Whitney U test is conducted. Performing the Mann-Whitney U test in SPSS generates the summary displayed in Table 5.

In Table 5 it can be seen that the mean rank of group 1 is slightly lower than the mean rank of group 2, however, they are very close to each other. SPSS also generates the p-value, which is the deciding factor for accepting the null hypothesis or the alternative hypothesis. The p-value for this test is equal to 0.350. This value has to be lower than 0.05 (for a significance level of 95%) to accept the alternative hypothesis. Since the value is greater than 0.05, the null hypothesis is accepted. This means that the distributions of the two groups are the same. In other words, there is no statistically significant difference

Ranks			
Nr of convolutional layers	**N**	**Mean Rank**	**Sum of Ranks**
1	20	18.75	375
2	20	22.25	445
Total	40		

Table 5. SPSS output Mann-Whitney U test.

between using one or two convolutional layers for dimensionality reduction.

3.3 Experiment 3: Determining the relevance of dimensions

The third experiment entails analysing the dimensionality reduction carried out by the convolutional layer especially added to Mask R-CNN for this purpose. This layer extracts the most relevant information from the input images, creating 3 bands that contain as much information as possible for classification purposes. In order to gain a deeper understanding of the dimensionality reduction that is taking place, an attempt is made to find out which of the 224 original bands contribute most to the resulting 3 dimensions.

The experiment is conducted by carrying out 10 runs of the model with one convolutional layer for dimensionality reduction. The weights that reflect the importance of the 223 hyper-hue dimensions of the input images are stored in arrays every epoch. The 223 hyper-hue dimensions are the output of the HHSI preprocessing step. The weights are stored in a three dimensional array: [3 x 223 x 25] (3 reflecting the resulting bands, 223 reflecting the hyper-hue dimensions and 25 reflecting the kernel components). The 10 resulting weight arrays (after 10 runs of 100 epochs) are used to generate graphs that visualize the importance of the 223 hyper-hue dimensions. For these graphs, the weights per hyper-hue dimension are first averaged over the 25 kernel components. This results in 223 vectors with 3 items (representing the 3 bands). In order to obtain a metric by which the 223 vectors can be compared, the length of each vector is calculated using Equation 6.

$$\| v \| = \sqrt{v \cdot v} = \sqrt{v_1^2 + v_2^2 + \cdots + v_n^2} \tag{6}$$

In this equation, v represents the vector and n is the number of elements in the vector. In this case, n equals 3. The length of the vector represents the importance of that particular hyper-hue dimension for separating the classes. To obtain a more generalized understanding, the values in the ten arrays, corresponding the the 10 runs, are averaged. The resulting graph is depicted in Figure 11. The peaks in this graph point to hyper-hue dimensions that have relatively large weights in the convolutional layer and therefore can be assumed to play a relatively large role in separating the plastics from each other. The graph shows a clear structure, suggesting that hyper-hue information is a useful way of separating and classifying PE, PP and PET. Unfortunately, it is difficult to directly link hyper-hue dimensions to wavelengths. Establishing this link would make it possible to compare the graph with existing spectral data of the plastics. The spectral data is depicted in Figure 13, which shows the relative reflectance values of the three plastics included in this research.

In order to make some comparison possible, the model is run once without the HHSI preprocessing step. The instance segmentation mAP that was achieved for this run was 0.789. The results are shown in Figure 16. Because this graph is hard to interpret due to outliers, it is placed in Appendix C. The same graph, excluding outliers, is shown in Figure 12. The values in the graph are computed in the same manner as described above.

Figure 12 shows that some wavelengths are more important than others for distinguishing the three plastics. Linking the data in the graph to Figure 13, it appears that the wavelengths that show large absolute differences in average relative reflectance (see Figure 13) are the ones that play a relatively important role in distinguishing between the three plastics. Examples are the wavelengths between 900 and 1100 and the wavelengths between 1250 and 1350; PE, PP and PET show large absolute differences in average relative reflectance around these wavelengths and their relative importance during the dimensionality reduction process is also large (see Figure 12).

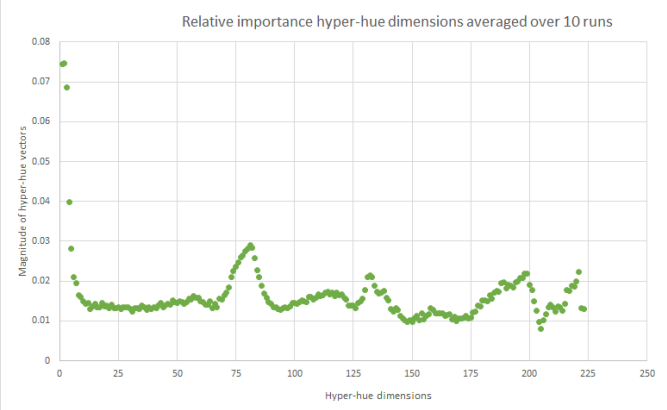

Fig. 11. Relative importance of hyper-hue dimensions for separating PE, PP and PET.

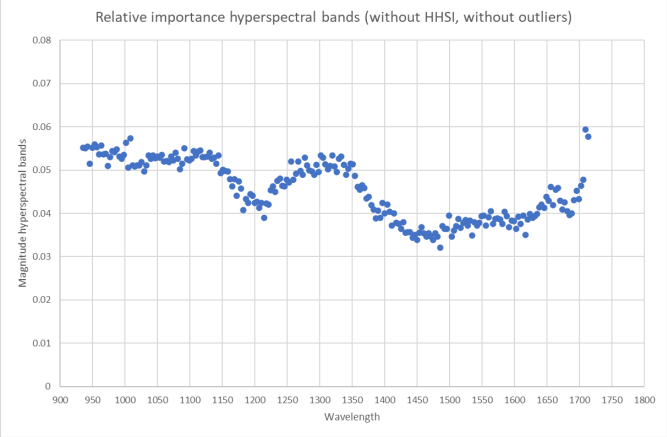

Fig. 12. Relative importance of hyperspectral wavelengths for separating PE, PP and PET (without outliers).

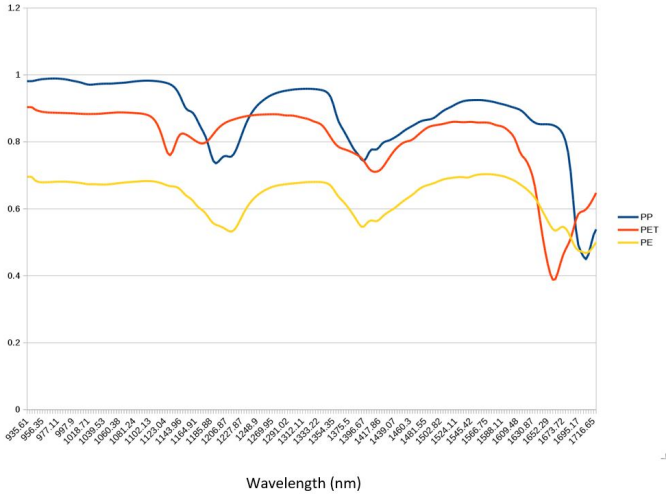

Fig. 13. Average relative reflectance of PE, PP and PET.

4 Conclusion & Discussion

Improving the plastic sorting process is necessary for increasing the global plastic recycling rate. This study explores the detection of plastic flakes on hyperspectral images by an instance segmentation convolutional neural network. A baseline experiment was conducted, using state-of-the-art network Mask R-CNN with a preceding convolutional layer for dimensionality reduction and 30 images of plastic flakes (PE, PP or PET) as input. Preprocessing consisted of applying the flat-field correction and the HHSI method. The results of the experiment are promising: excluding outliers, the average performance of the model after 20 runs of 100 epochs is a instance segmentation mask mAP (IoU of 0.5:0.95) of 0.8261 and a bounding box mAP (IoU of 0.5:0.95) of 0.7724. Including outliers, mask mAP is 0.7057 and bounding box mAP 0.6718. These scores are much better than the performance of Mask R-CNN on COCO test images – the first three scores are more than twice as high [17]. Granted that COCO test images are much more of a challenge than the ones used in this research, it is safe to say that the results of this study are encouraging.

An especially interesting part of this study is the convolutional layer that was added to the model to reduce the dimensionality of the hyperspectral images. Judging by the baseline results described above, we can surmise that this layer performed well. An experiment was conducted to find out whether two convolutional layers for dimensionality reduction further improve the performance of the model. For this experiment, 20 runs of 100 epochs were carried out with an extra convolutional layer, and the results were compared with the findings of the baseline experiment described above. Statistical tests on the results showed that adding an extra layer does not have a significant effect on instance segmentation mask mAP. This is an interesting result that deserves further investigation. Another experiment focused on which dimensions of the hyperspectral images contribute most to the three bands created by the convolutional layer. For this experiment, 10 runs of the model were conducted, plus one more without the HHSI preprocessing step for comparison. The weights attached to the various dimensions at the end of the runs are seen as a reflection of their contribution (to the separability of the classes). The results suggest that some hyper-hue dimensions are clearly more relevant than others for separating PE, PP and PET. Linking these hyper-hue dimensions to specific wavelengths is an interesting subject for future research. When the plastics are separated on the basis of hyperspectral bands, instead of hyper-hue dimensions, the results appear to suggest that the most relevant wavelengths for separating the plastics are the ones where the relative

reflectance of the three plastic types differs the most.

A limitation of this research is the fact that dimensionality of the images had to be reduced to three in order to use the pre-trained Mask R-CNN model. It is recommended that the option of rewriting the Mask R-CNN code to accommodate images with more than three dimensions is explored. Another limitation of this study is the small number of images used. It is recommended to repeat the experiments in future research with an augmented dataset. It is further recommended that this dataset also includes images of bioplastics. Another suggestion for future research is to create more challenging images, for example by increasing the number and types of flakes/objects per image or by occlusion of the objects on the images. As mentioned before, the number of convolutional layers used for dimensionality reduction also deserves to be researched more. It is, for example, interesting to look into increasing the number of convolutional layers more drastically or changing the kernel size of the convolutional layers. Future research could also focus on the subject of the third experiment about the importance of hyper-hue dimensions or individual hyperspectral bands (depending on whether the HHSI preprocessing method is applied) in the dimensionality reduction stage. A final suggestion for future research is to test the performance of instance segmentation model Mask Scoring R-CNN. This model is claimed to perform better than Mask R-CNN. This improved performance stems from the fact that the model prioritizes more accurate mask predictions.

The promising results of this study show the potential of state-of-the art instance segmentation technique Mask R-CNN for localizing and detecting PE, PP and PET flakes. The combination of Mask R-CNN with the other techniques used in this research - hyperspectral images, preprocessing by flat-field correction and the HHSI method, and dimensionality reduction by CNN – worked well. These techniques and their combined application are worth further study, aimed at the ultimate goal of increasing the effectiveness of plastic sorting mechanisms and reducing the amount of plastic waste around the world.

REFERENCES

[1] Colette Wabnitz and Wallace J Nichols. Plastic pollution: An ocean emergency. *Marine Turtle Newsletter*, (129):1, 2010.

[2] Li Shen and Ernst Worrell. Plastic recycling. In *Handbook of Recycling*, pages 179–190. Elsevier, 2014.

[3] OECD. Improving plastics management: Trends, policy responses, and the role of international co-operation and trade. Technical report, 2018.

[4] Giuseppe Bonifazi, Giuseppe Capobianco, and Silvia Serranti. A hierarchical classification approach for recognition of low-density (ldpe) and high-density polyethylene (hdpe) in mixed plastic waste based on short-wave infrared (swir) hyperspectral imaging. *Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy*, 198:115–122, 2018.

[5] Shutao Li, Weiwei Song, Leyuan Fang, Yushi Chen, Pedram Ghamisi, and Jón Atli Benediktsson. Deep learning for hyperspectral image classification: An overview. *IEEE Transactions on Geoscience and Remote Sensing*, 57(9):6690–6709, 2019.

[6] Richardson Santiago Teles de Menezes, Rafael Marrocos Magalhaes, and Helton Maia. Object recognition using convolutional neural networks. In *Artificial Neural Networks*. IntechOpen, 2019.

[7] Patrik Kamencay, Miroslav Benčo, Tomáš Miždoš, and Roman Radil. A new method for face recognition using convolutional neural network. 2017.

[8] Jiuxiang Gu, Zhenhua Wang, Jason Kuen, Lianyang Ma, Amir Shahroudy, Bing Shuai, Ting Liu, Xingxing Wang, Gang Wang, Jianfei Cai, et al. Recent advances in convolutional neural networks. *Pattern Recognition*, 77:354–377, 2018.

[9] Keiron O'Shea and Ryan Nash. An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*, 2015.

[10] Rikiya Yamashita, Mizuho Nishio, Richard Kinh Gian Do, and Kaori Togashi. Convolutional neural networks: an overview and application in radiology. *Insights into imaging*, 9(4):611–629, 2018.

[11] Ce Yang, Won Suk Lee, and Paul Gader. Hyperspectral band selection for detecting different blueberry fruit maturity stages. *Computers and Electronics in Agriculture*, 109:23–31, 2014.

[12] Li Liu, Wanli Ouyang, Xiaogang Wang, Paul Fieguth, Jie Chen, Xinwang Liu, and Matti Pietikäinen. Deep learning for generic object detection: A survey. *International Journal of Computer Vision*, pages 1–58, 1809.

[13] AA Gowen, CPo O'Donnell, PJ Cullen, G Downey, and JM Frias. Hyperspectral imaging–an emerging process analytical tool for food quality and safety control. *Trends in food science & technology*, 18(12):590–598, 2007.

[14] Behnood Rasti, Danfeng Hong, Renlong Hang, Pedram Ghamisi, Xudong Kang, Jocelyn Chanussot, and Jon Atli Benediktsson. Feature extraction for hyperspectral imagery: The evolution from shallow to deep. *arXiv preprint arXiv:2003.02822*, 2020.

[15] Hamed Masoumi, Seyed Mohsen Safavi, and Zahra Khani. Identification and classification of plastic resins using near infrared reflectance. *Int. J. Mech. Ind. Eng.*, 6:213–220, 2012.

[16] Christian Szegedy, Alexander Toshev, and Dumitru Erhan. Deep neural networks for object detection. In *Advances in neural information processing systems*, pages 2553–2561, 2013.

[17] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[18] Anurag Arnab and Philip HS Torr. Pixelwise instance segmentation with a dynamically instantiated network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 441–450, 2017.

[19] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[20] Ross Girshick. Fast r-cnn. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[21] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[22] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[23] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[24] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[25] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271, 2017.

[26] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

[27] Shifeng Zhang, Longyin Wen, Xiao Bian, Zhen Lei, and Stan Z Li. Single-shot refinement neural network for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4203–4212, 2018.

[28] F Sultana, A Sufian, and P Dutta. A review of object detection models based on convolutional neural network. *arXiv preprint arXiv:1905.01614*, 2019.

[29] Mingxing Tan, Ruoming Pang, and Quoc V Le. Efficientdet: Scalable and efficient object detection. *arXiv preprint arXiv:1911.09070*, 2019.

[30] A Stellingwerf and J Hu. Description of polymer classification by applying hyperspectral imaging and deep learning techniques. NHL Stenden University of Applied Sciences, 2018.

[31] Maider Vidal and José Manuel Amigo. Pre-processing of hyperspectral images. essential steps before image analysis. *Chemometrics and Intelligent Laboratory Systems*, 117:138–148, 2012.

[32] Jianwei Qin. Hyperspectral imaging instruments. In *Hyperspectral imaging for food quality analysis and control*, pages 129–172. Elsevier, 2010.

[33] Huajian Liu, Sang-Heon Lee, and Javaan Singh Chahl. Transformation of a high-dimensional color space for material classification. *JOSA A*, 34(4):523–532, 2017.

[34] Banda Gerald. A brief review of independent, dependent and one sample t-test. *International Journal of Applied Mathematics and Theoretical Physics*, 4(2):50–54, 2018.

[35] Nornadiah Mohd Razali, Yap Bee Wah, et al. Power comparisons of shapiro-wilk, kolmogorov-smirnov, lilliefors and anderson-darling tests. *Journal of statistical modeling and analytics*, 2(1):21–33, 2011.

[36] Nadim Nachar et al. The mann-whitney u: A test for assessing whether two independent samples come from the same distribution. *Tutorials in quantitative Methods for Psychology*, 4(1):13–20, 2008.

[37] Donald W Zimmerman. A note on the influence of outliers on parametric and nonparametric tests. *The journal of general psychology*, 121(4):391–401, 1994.

[38] Amir Momeni, Matthew Pincus, and Jenny Libien. *Introduction to statistical methods in pathology*. Springer, 2018.

[39] Frank Wilcoxon, SK Katti, and Roberta A Wilcox. Critical values and probability levels for the wilcoxon rank sum test and the wilcoxon signed rank test. *Selected tables in mathematical statistics*, 1:171–259, 1970.

[40] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[41] B Planche and E Abres. *Hands-On Computer Vision with TensorFlow 2*. Packt Publishing, 2019.

[42] Kenneth Stehlik-Barry and Anthony J Babinec. *Data analysis with IBM SPSS statistics*. Packt Publishing Ltd, 2017.

A EXPERIMENT 2: RESULTS 40 RUNS

	mAP instance segmentation	
run	**1 convolutional layer**	**2 convolutional layer**
1	0.692	0.867
2	0.884	0.85
3	0.842	0.875
4	0.294	0.842
5	0.883	0.85
6	0.85	0.884
7	0.705	0.783
8	0.85	0.867
9	0.892	0.112
10	0.85	0.754
11	0.859	0.825
12	0.85	0.121
13	0.833	0.85
14	0.809	0.883
15	0.073	0.751
16	0.084	0.867
17	0.446	0.892
18	0.825	0.093
19	0.734	0.833
20	0.859	0.867

Table 6. Results of 40 runs of Mask R-CNN: comparison between 1 and 2 convolutional layers for dimensionality reduction.

B EXPERIMENT 2: Q-Q PLOTS

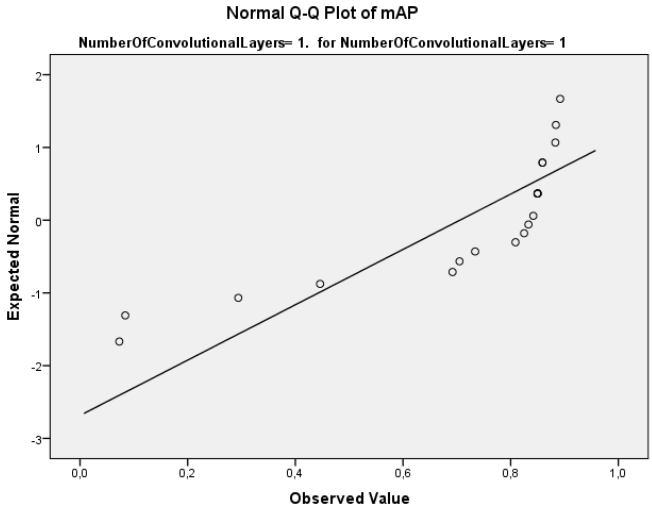

Fig. 14. SPSS output Q-Q plot group 1.

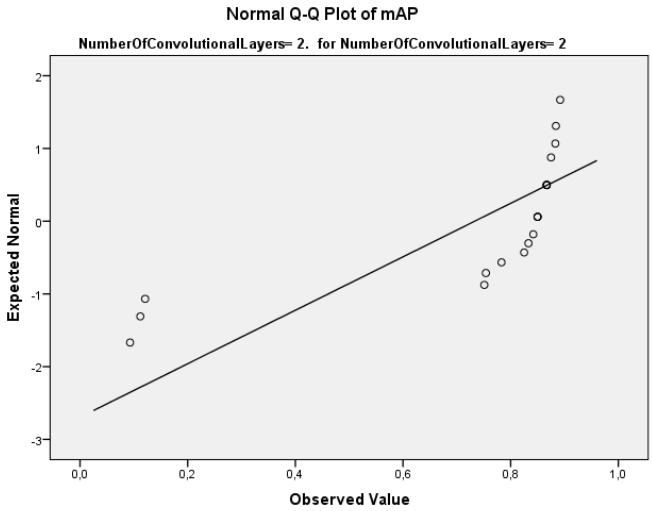

Fig. 15. SPSS output Q-Q plot group 2.

C Experiment 3: Relative importance of hyperspectral wavelengths

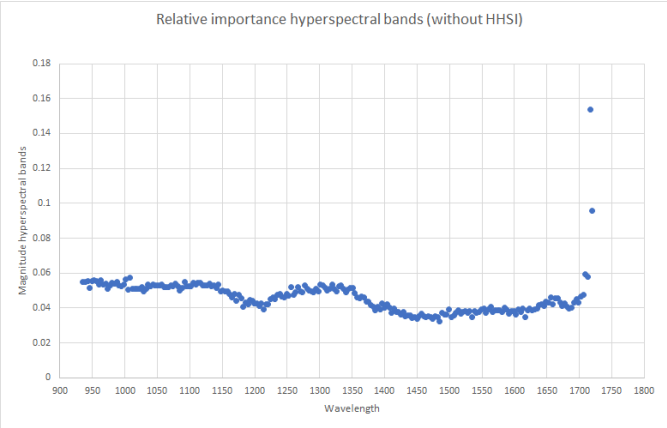

Fig. 16. Relative importance of hyperspectral wavelengths for separating PE, PP and PET (including outliers).