Diseño de un Sistema en Línea de Clasificación y Selección de Frutos de Café Usando LEDs de Banda Estrecha y MCUs

Leonardo Manrique Naranjo

Universidad Nacional de Colombia
Facultad de Ingeniería y Arquitectura, Departamento de Ingeniería Eléctrica, Electrónica y Computación
Manizales, Colombia
2020

Design of an On-line Multispectral Coffee Fruit Classification and Sorting System Using Narrowband LEDs and MCU

Leonardo Manrique Naranjo

Thesis submitted in partial fulfillment for the requirements for degree of
MSc in Engineering - Industrial Automation

Advisor:
Ph.D. Gustavo Adolfo Osorio

Research line:
Signal and Image Analysis and Recognition, Electronic Design
Research group:
Percepción y control inteligente

Universidad Nacional de Colombia
Facultad de Ingeniería y Arquitectura, Departamento de Ingeniería Eléctrica, Electrónica y Computación
Manizales, Colombia
2020

Acknowledgements

All the work conducted during the last two years has been possible thanks to the financial and logistic support of Titoma Design Ltd., who strongly believes in the talent of Colombian engineers and acknowledges the tremendous social, economic and cultural impact of agricultural practices.

Special thanks to Eng. Hsuming Hsieh for the mechanical modeling and prototyping, Eng. Juan Vargas for the flawless hardware implementations and creative and patient testing, and PhD. Gustavo Osorio for the high-level guidance and advising through invaluable practical and philosophical discussions, demonstrating how an equally enjoyable and fulfilling experience can take place.

Abstract

The growing trend of specialty coffees brings the possibility of increased income for Colombian coffee producers, who have been recurrently hit by market conditions, climate change, supplies cost increase and aftermath of violence. A key requirement for specialty coffee, regardless of the post-production method to use, is the exclusive use of ripe coffee fruits, which could be red or yellow; all green fruits must be removed.

The design proposed in this thesis is the result of a financial and technical market research, and implements a low-cost discrete color sensor and commercially available LEDs system controlled by a high-performance Cortex-M4 core MCU, running algorithms derived from supervised learning techniques optimized for a MCU, achieving an electrical assembly of under US$100 when manufactured in high volumes. This work focuses on the increased discriminant capacity of the sensor, achieved by examining different narrowband light source configurations through intensive computing methods.

While multiple classifiers are studied, the resulting model achieves an accuracy over 99% using an ensemble LDA, at a rate of more than 10 fruits per second or 72 Kg per hour in a portable form-factor, by using only 3 discrete LEDs and a light to frequency converter.

Keywords: Coffee Fruits, On-line, Omnidirectional, Inspection, Sorting, LDA, Feature selection, Embedded processing, SVC, Random Forests, GNB, Lasso, Classifiers, Feature extraction, Color sorting

Resumen

La tendencia creciente de cafés especiales abre la posibilidad de generación de ingresos adicionales para los productores de café Colombianos, quienes han sido golpeados recientemente por condiciones de mercado, cambio climático, incremento de costo de suministros y resultados de la violencia entre otras razones. Un requerimiento clave para cafés especiales, sin importar el método de post-producción utilizado, es el uso exclusivo de frutos de café completamente maduros, los cuales pueden ser rojos o amarillos; todos los frutos verdes deben ser removidos.

El diseño propuesto en esta tesis es el resultado de una investigación de mercado tanto financiera como técnica, e implementa un sensor de color discreto de bajo costo, así como LEDs disponibles comercialmente controlados por un procesador de alto desempeño de arquitectura ARM Cortex-M4, ejecutando algoritmos de clasificacion derivados de técnicas entrenamiento supervisado optimizados para micro controladores, obteniendo un ensamble electrónico de costo inferior a US$100 en volúmenes altos. El trabajo se enfoca en la capacidad discriminante aumentada del sensor, la cual se obtiene gracias al análisis de diferentes configuraciones de fuentes de luz de banda angosta a través de métodos de computación intensiva.

Si bien múltiples clasificadores son utilizados, el model resultante obtiene una certeza superior al 99% utilizando un clasificador LDA tipo ensamble, a una tasa de clasificación mayor a 10 frutos por segundo o 72 Kg por hora en un factor de forma portátil, utilizando solo 3 tipos de LEDs discretos y un conversor de luz a frecuencia.

Palabras clave: Frutos de café, omnidireccional, inspección, clasificación, LDA, Selección de características, Procesamiento embebido, SVC, Bosques Aleatorios, GNB, Lasso, classificadores, Extracción de características, Selección por color.

Contents

Figure list

Table list

List of Symbols

Symbol	Definition
ADC	Analog to digical converter
ARM	Advanced RISC machines
BJT	Bipolar junction transistor
CCD	Charge coupled device
CIFE	Conditional infomax feature extraction
CMIM	Conditional mutual information maximization
CMOS	Complementary metal-oxide-semiconductor
COM	Communication serial port
COP$	Colombian peso
CPU	Central processing unit
CV	Cross validation
DMA	Direct memory access
FET	Field effect transistor
FMI	Fowlkes-Mallows index - Geometric mean of precision and recall
$FMI_{accuracy}$	Geometric mean in terms of accuracy
FMI_{proba}	Geometric mean in terms of probability
$FMI_{weighted}$	Gemoetric mean in terms of accuracy times probability
FN	False negative rate
FP	False positive rate
FPGA	Field programmable gate array
FPU	Floating point unit
FTDI	Future technology devices international
GNB	Naïve Bayes classifier
GPIO	General purpose input and output
I2C	Inter integrated circuit
IR	Infra red
LDA	Linear discriminant analysis
LED	Light emitting diode
MCU	Micro controller
MIM	Mutual information Index

Symbol	Definition
ms	millisecond
nm	nanometer
OvO	One versus One
OvR	One versus Rest
PC	Personal computer
PCB	Printed circuit board
PCBA	Printed circuit board assembly
PTC CREO	3D CAD software
PWM	Pulse width modulation
RF	Random Forests
SMD	Surface mount device
SVC	Support vector classifier
SVM	Support vector machine
TDM	Time division multiplexing
TP	True positive Rate
TTL	Transistor to transistor Logic
UART	Universal asynchronous receiver transmitter
USB	Universal serial bus
USD$	United states Dollar

1 Introduction

1.1 Background and motivation

In Colombia, more than 600,000 families who derive their income from coffee production are facing increasing economic pressure due to volatile market conditions [1], climate change [37, 17] and supplies cost increase. Internationally, Colombian coffee is seen as a low-cost option mostly used for blends. Other coffee exporting countries have managed to position their coffees as a higher end product, thanks to the adoption of innovative growing and post-collection techniques [36].

Specialty coffees offers a novel opportunity for growers to improve their economic output dramatically. The first step in specialty coffee production is pure-mature coffee selection, which currently does not take place in most Colombian farms, both due to lack of knowledge and lack of resources. Coffee growers who supported the design of the device agree that the design and manufacturing of an automated, low-cost coffee fruit sorting machine would enable and speed up such processes.

In chapter 2, we outline the results of a state of the art review which shows several solutions for dry coffee sorting, however no commercial solutions are currently available for coffee in berry state. Patents such as US9909978, US4203522A, US4513868A and US5538142 detail systems which can sort granular material through color measurement, splitting it in acceptable and rejects by means of pneumatic ejection. These devices are meant to operate on dry coffee fruits, and not on freshly picked coffee berries.

Previous works have tackled the problem of coffee fruits sorting reaching accuracies within a 79% to 99% range in a low speed set-up of less than 10 fruits per second. Among the signal processing and machine learning approaches used we found Bayesian Classifiers and Neural networks (Sandoval [35]), clustering algorithms (Betancur [4]) and color space transforms SCT (Montes [25]) and HSV (Ramos [32]) with subsequent region segmentation techniques to determine maturity. Most of the existing work does not account for the need to conduct a complete scan of all the coffee fruit faces, now total device cost, which are mandatory requirement to ensure no false positives are allowed.

1.2 Objectives

Our main objective is to design a low cost coffee sorting system that can be purchased by the average Colombian farmer, with accuracy above 99% and a sorting speed of 10 fruits per second. Chapters 2 and 3 go into deeper detail on the system requirements:

- Low Cost as defined by a business case analysis of Chapter 2
- Over 99% sorting accuracy
- Processing speed over 10 fruits per second
- Two face fruit scan

Chapter 3 details the design decisions made to achieve the low cost objective, such as using a discrete color sensor instead of CMOS or CCD sensor arrays or cameras, which are not only expensive by themselves, but also require a more expensive computational platform. In order to improve the discriminant capacity of the system we propose a multispectral light system which in turn requires an optical integration system. The product is designed with high volume manufacturing and patentability in mind.
This thesis describes the design process of a coffee fruit classifier which exceeds the existing systems performance and meets the product objectives by implementing a supervised machine learning algorithm and a narrowband multispectral system to maximize the discriminant capacity of a very low-cost discrete sensor and commercially available LEDs.

1.3 Materials and methods

The design process starts with a review of existing devices in order to set a starting point for a possible complete product architecture. This exhibits common points in their construction. We then execute a literature review which points towards color as being the best and possibly sole characteristic that can be consistently used to determine maturation stage in coffee fruits, especially minding that some coffee varieties turn yellow instead of red when fully ripe.

We split the device in three main sections: Dispensing, Capture and Sorting, with the core of the work focusing on Capture and Sorting. First, a detailed analysis of each of the system components takes place. We emphasize the process of selecting the LEDs based on optimizing the system performance with the given color sensor. Several supervised learning algorithms are used to execute and validate such selection.

To design the capture system, we built upon existing work and practical results of Montes [26] and Tamayo [42], and develop a radially arranged set of high power and wide view angle SMD LEDs, including a power control stage. A set of reflective optical elements are required

to equalize the radiated power and compensate for the off-center distance of the emitters. The result is a coherent multispectral light beam built from narrowband LEDs. The data is acquired using a low-cost light to frequency converter. The decision of which LEDs to use is the center of the research and discussion of chapter 4.

The development of the classification algorithm takes place in parallel with LED selection. Multiple supervised machine learning classifiers are trained and evaluated under different light conditions, selecting those with highest discriminant performance, measured in terms of precision, recall and classifier certainty. Chapter 5 explores the optimization of the best performing algorithm with aims at implementing it on a MCU platform. On the following sections of this chapter we explain the key elements of the sorting system.

As a last step, we conduct an optimization process for the best performing algorithm, in order to minimize the number of features or characteristics required, thus speeding up the decision-making process and lowering the computing power required, possibly opening a door for the use of lower cost micro-controllers.

The resulting prototype is a field-ready device which can sort coffee fruits with a classifying accuracy over 99% by using only 3 discrete commercially available LEDs and a simple light to frequency converter controlled by a MCU running algorithms trained by Machine Learning methods. The system exploits the maximum discriminant capacity of the modest sensor by using narrowband spectral imaging techniques and machine learning algorithms, making it cost efficient, portable and reliable.

The Firmware is implemented in C through various state machines and service managers on top of STM32 Cube MX Hardware abstraction layer, making extensive use of the MCU DMAs, Interrupt and serial peripherals.

1.4 Thesis structure

This document is divided in 5 Chapters.

- Chapters 1 and 2 introduce the problem and physical constraints.
- Chapter 3 describes the proposed system.
- Chapter 4 goes into deeper detail on the LED and Sensor selection.
- Chapter 5 discusses the optimization of the selection algorithm for MCU implementation.

2 Coffee fruit sorting system

2.1 Coffee in Colombia

In Colombia, more than 600,000 families derive their income from coffee growing. The average farm size is 2 hectares [12] with average productivity of 3650 Kg per year [3], which as of February 2020 is sold at COP 7,720 per Kg [14] representing income of US$8,000 per year. For the last 10 years, Colombian coffee growing families have produced between 800.000 and 1.600.000 bags of 60 Kg or 48 to 96 tons of coffee per month, with more than 80% of it being exported [13]. Despite the production numbers, farmers consistently report the income derived from coffee growing is insufficient and unreliable; this has pushed many to look into alternate crops, as an example, coffee growing is only the 6th in volume in Caldas [5].

2.1.1 Specialty coffees

Specialty coffees are those which preserve physical (size, shape, humidity, appearance, defects), sensorial (smell, visual, taste), cultural-practical (collection, washing, drying) or final processes characteristics (toasting, grinding, preparation). Farfan [8] mentions these characteristics distinguish Specialty from common coffees and explain why customers are willing to pay a premium price.

Within specialty coffees, those with rare varieties and special post-collection processes usually obtain the highest cupping scores, which translates to higher sell cost. Panama consistently produces high quality specialty coffees, and in 2019 the most expensive coffee at the SCAP Best of Panama bid reached USD$1029/lb [36] with several other coffees exceeding USD$100/lb. These coffees undergo an aerobic natural fermentation processes prior to pulping. A key requirement for the fermentation process is that green fruits must be fully removed.

Studies have determined that the highest quality of a coffee cup can be obtained from mature fruits, while green fruits deteriorate the taste and aroma due to diverse defects such as ferment and acre. Green coffee fruits have a great negative impact when their concentration exceeds 2.5% [15]. The lowest number of coffee cup defects was found when using only ripe fruits [23].

2.2 Current sorting process

A limited number of coffee growers are currently producing specialty coffees. They are being forced to implement a manual selection process (Figure **2-1**).

Figure 2-1: Colombian coffee growers performing manual selection.

Ripe fruits selection is typically done manually in raised wooden beds (Figure **2-2**). The process is inefficient, error prone and time consuming.

Figure 2-2: Manual selection beds.

2.3 Business case for a color sorting system

A typical Colombian coffee farm could produce an additional yearly income of US$1,500 by implementing specialty coffee production methods, the cost of implementing a coffee sorting device should not exceed this mark, as coffee growers typically struggle to get a credit approved.

Coffee growers cannot afford expensive sorting systems such as those used for tomatoes, olives and other fruits. Previous works reached desktop prototypes for color sorting, however none of them have been built with the objective of being easily and affordably manufactured in high volumes. A low manufacturing cost could open a door to novel business models such as equipment rental, a modality known and appreciated by the farmers.

2.4 Coffee fruit characteristics

2.4.1 Color

Marin [23] points that the green and red pigmentation of the coffee fruit skin is dictated by the concentration of chlorophyll and anthocyanin. The decrease in chlorophyll concentration is a valuable indicator of the ripening of fruits, as found by Minguez [27] in Olives. Carvajal [18] confirms color is the most suitable variable to determine the maturation stage of fruits.

Color on coffee fruits progresses from green to yellow and red as the level of chlorophyll content on the skin decreases. This alters the reflectance characteristics in a non-uniform manner generating patches. It is necessary to scan the entire surface of the fruit in order to correctly classify it, as can be seen on Figure **2-3**. The maturation process is not uniform across the fruits of a given tree (Figure **2-4**). This results in pickers collecting fruits from all maturation stages at the same time.

2.4.2 Size

Fruit size varies greatly between varieties. Robusta coffees such as Pacamara and Maragogipe generally produce larger fruits which can exceed 20mm consistently, while typica and Caturra have higher density beans. Size also changes with maturation stage. Alvarez [7] reported that in average fruits grow as they become ripe; however, it is not a reliable prediction of maturation stage by itself.This is made evident in Figure **2-5**. A coffee sorting device must be engineered to process fruits of varying sizes, regardless of their maturation stage.

Figure 2-3: Typical mix of coffee fruits with different maturation stages present.

Figure 2-4: Left: Colombia variety coffee tree branch at peak of the season. Right: coffee collection bucket.

Figure 2-5: Fruits of the same variety and lot, picked the same day exhibit different sizes regardless of maturation stage.

2.5 Color classification machine vision systems

Color spectroscopy, both reflective and transmissive, used together with linear regression analysis have been widely used in the field of inferential metrology as a successful non-invasive method. Fen [10] used halogen lamps with laboratory spectrometers in the 350nm to 1000nm range to detect the presence of pesticides in vegetables by reflection, building a classifier using a neural network derived from a Principal Component Analysis. Zhang. [24] used a similar set-up and process to determine foliage damage. Several teams have studied fruits. Wang [45] managed to estimate the vitamin C content in chili peppers and Li [21] determined the content of soluble solids in oranges using a spectrometer. Multispectral imaging sensors have been used together with halogen lamps to predict fruit characteristics such as sugar content in bananas [2], maturation stage and soluble solids in mangoes [33], depth of impacts in guavas [28] and senescence in apples [41] and prunes [43]. Regression methods of Partial Least Square Regression and Linear Discriminant Analysis were used in all of the aforementioned cases.

Other studies have attempted to classify fruits by color using machine vision. Wang [47] designed a sorting system for melon and fruit product, Putra [31] designed a fruit sorting tool by means of color sensors using tomatoes, Sidehabi [40] developed a machine vision system for sorting passion fruit based on ripeness by training SVMs, and Sihombing [30] developed a laboratory proof of concept for fruit sorting using a discrete color sensor.

Color selection has been deemed as the most efficient method to define the maturation state of coffee fruits during their development and ripening [18, 23]. Coffee fruits generally exhibit homogeneous and heterogeneous color distribution patterns, as reported by Marin [23] and Carvajal [18]. Sandoval informs in [35] that during the maturation stage of coffee it is possible to have fruits where the color is homogeneous over the complete epidermis, while in other stages there is a soft or steep variation in color. An important percentage of the processing load of classifiers comes from imperfections in the image acquisition process, such as background removal, reflection and shadow control [26].

Diverse studies have taken place with the aim of classifying and selecting fruits by maturation stage using multispectral analysis. The algorithms used have reached selection accuracy of 85% to 97% in controlled environments and with low speeds. Sandoval [35] Obtained classification errors between 5.43% and 7.46% using Bayesian classifiers and neural networks, with processing rates of under 10 fruits per second, Betancur [4] reached 79% true acceptance rates, while Montes [25] obtained a 91% effectiveness using a Bayesian classifier and Ramos [32] reached between 94.8 and 99.6% variable efficacy. All previous work was conducted in laboratory set-ups, none of these units were designed nor optimized for on-line real-time sorting, and most required expensive CMOS sensor set-ups with computer or FPGA pro-

cessing. All of the studies referenced to date only inspect one face of the coffee fruits, which lowers the reliability.

2.6 Existing commercial sorting systems

Several coffee sorting machines can be found in the market such as the FMS2000 (Satake, Stafford ,USA), Sortex SG1 (SG Solution Co. Ltd., Hefei, China) or HELIUS (TOMRA Sorting, Denver, USA); however, these can only sort coffee beans (the fruit after its pulped and dried) which makes them unsuitable for specialty coffees because selection must take place before the fruits are dried. Other generic sorters such as Color Sorter (Zhongrui Weishi Optoelectronic Co, Shenzhen, China) can be adapted for fruits at a very high cost. The operation principle of all devices is similar: fruits or beans fall through slanted surfaces, are then illuminated by high power white LEDs and scanned by various types of sensors, typically cameras. The captured frames are analyzed and pneumatic ejection takes place. Due to the lack of feasible automated options, coffee sorting in Colombia remains a manual, expensive, tedious, inefficient and error prone process.

2.7 Coffee sorting system requirements

Low cost

Prior research has used illumination set-ups, spectrometers or sensors and data processing systems which are prohibitively expensive; the existing solutions in the market also greatly exceed the yearly budget of a typical farmer. A target cost of US$500 is reasonable and comparable with standard coffee processing tools. The ejection system requires existing off-the-shelf compressors and electrovalves, which could sum up to US$300 to US$400 when purchased in high volumes. As a result, the electronic assembly should be produced in volumes of thousands of units per run for less than COP$350,000 or USD$100.

High sorting accuracy

Academic reports mention that a balanced cup tolerates a maximum rate of false positives (green fruits classified as mature) of 2.5% [15]. Prior studies have been unable to reach this target. Furthermore, the specialty coffee purchasers have a 0 tolerance for false positive, while susceptibility to false negatives (mature fruits classified as green) is more relaxed. The system is only viable if it is reliable and accurate, thus this is the main design driver. Within the framework of this document, sorting accuracy is defined in terms of precision (rate of

True Positive rate) and recall (rate of False Positive rate).

High processing speed

Most of the research to date has emphasized in the feasibility of color sorting, but few prototypes make it to on-line test models, hence processing speed is discarded or not officially recorded. A coffee sorting system is intended primarily for specialty coffees which are produced in small lots which must be processed within hours of having been collected. The classifier is expected to also be used for regular (non-specialty) coffee; it is undesirable to have a large device which can process thousands of kilograms per hour, as that would needlessly increase the cost of a device meant to be used only a few hours per day.

On-line portability

Due to the typical distribution of a coffee farm, it is necessary to relocate the sorting equipment after use. It can also be the case that different coffee sorting locations are established throughout the harvest period. It is a design imperative that the final device can be manufactured in scale, and not only used for laboratory analysis.

2.8 Coffee sorting system components

The literature review shows that most attempts at classifying coffee by maturation stages use color and no other variables. An evaluation of existing devices shows that typical coffee sorting systems generally follow the same construction with three main stages in the process (Figure **2-6**):

- Dispensing: Fruits are oriented and aligned with the sensor for scanning.
- Capture: Fruits are detected, illuminated and sensor data is read.
- Sorting: A classification decision is made and separation mechanisms are activated.

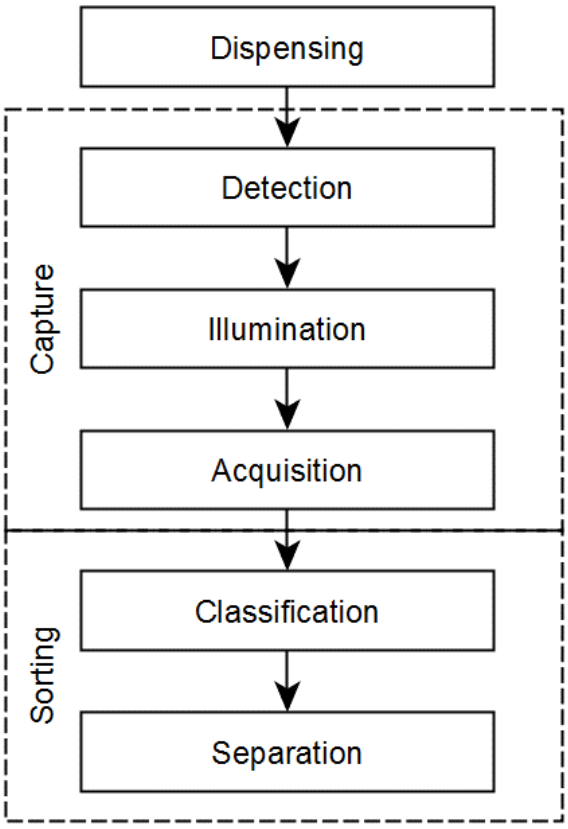

Figure 2-6: High level diagram of the key components of a color sorting system

3 Design of a color inspection and sorting system

In Chapter 2 we split the typical construction of a sorting system in 3 main parts: Dispensing, Capture and Sorting. We focus our research on the Capture and Sorting, and simplify the Dispensing stage using a free-falling construction with funnels as found in several other devices.

3.1 Dispensing

The typical construction of sorting systems for manufacturing applications use centrifugal alignment drums, conveyor belts or free-falling dispensing. For discrete color sensor applications, it is necessary to ensure one by one fruit scanning, where fruits are well aligned with the light sources and the sensor scanning area.

Coffee growers are accustomed to using pulping, sorting and threshing machines which use funnels to guide free falling fruits or beans towards the processing mechanisms. The coffee sorting machine will also use this principle during its first iterations.

3.1.1 Fruit speed

To determine the speed requirements, the terminal speed of terminal speed of coffee fruits, assuming an initial speed of 0, as we can expect from fruits being poured over the intake:

$$V = \sqrt{(\frac{(2 * W)}{(C * \rho * A)}}$$

Where
W= weight of falling object = 2 gr
ρ= density (of air) = 1.225 kg/m3
A= projected area, of a 7mm radius object
C= drag coefficient = 0.47

$$V = \sqrt{\frac{(2 * 0.002)}{(0.47 * 1.225 * \pi * 0.007^2)}} = 6.7m/s$$

The terminal speed of an average sized coffee fruit would reach 6.7m/s. Due to typical construction of coffee processing equipment, fruits would see a drop of maximum 1 mt. We find the speed after a 1 mt fall:

$V = \sqrt{2 * g * h}$

$V = \sqrt{2 * 9.8 * 1} = 4.4m/s$

The fruits are not expected to reach terminal velocity and the maximum expected speed is 4.4m/s. Coffee fruits of Arabica varieties range between 5mm to 25mm in diameter; a 10mm separation between fruits can be assumed. For the smallest fruit this means a total separation of $5mm + 10mm = 15mm$; at a speed of 4.4 m/s a 5 mm fruit would be in front of the sensor for 1.1 ms. This is a worst-case scenario. The system must be able to obtain sufficient reads to reach a decision in less than 1ms.

3.2 Capture

3.2.1 Detection System

A fruit detection system is implemented through a contactless IR barrier sensor GP1A57HRJ00F (SHARP, Tokyo, Japan), which is inexpensive and easy to source and implement. The barrier sensor dimensions and shape can be manually altered, allowing to detect larger fruits. Attention has been paid to the location and orientation of the IR sensor because it emits light in the sensitive region of the sensor.

3.2.2 Color Sensor

Color is a representation of the reflectance characteristic of objects at different wavelengths. Strictly speaking, color is a human construct, as they are simply names assigned to wavelengths at which the human eye is sensitive. Color sensors are typically designed to mimic the response of the human eye to different wavelengths.

CCD or CMOS sensor arrays

Sensor arrays (cameras) capture a large amount of redundant data because the color of the surface of the fruits typically only contains various tones of red, green or yellow, distributed in patches. Furthermore, the problem of background and shadow removal lowers the classifiers performance [42, 26] and increases processing time and required computational power, as they require expensive and complex CPU or FPGA platforms for data acquisition and processing.

Discrete color sensors

Commercial discrete color sensors are built as arrays of photodiodes which represent different spectral regions, referred to as channels, typically red (R), green (G) and blue (B). These channels are created using a single type of photodiode and multiple band-pass filters. The filters present non-linearities due to the nature of the manufacturing process and materials used, typically dyed glass or acrylates, which create uneven and overlapping spectral sensitivity curves.

The use of narrowband illumination techniques can maximize the spectral responsiveness of a sensor channel if light is emitted only or mostly in the highest sensitivity areas with lowest inter-channel overlap. These sensitivity areas are sensor-dependent, and usually peak around blue (450 to 470nm), green (525 to 550nm) and red (650 to 700nm), which are also the peaks of the human eye sensitivity for different colors.

The programmable color light-to-frequency converter TCS3200 (ams AG, Styria, Austria) features discrete channels (R, G, B) and allows for fast data acquisition below 1ms given proper lighting conditions, while offering a varying frequency output interface that can be read by a high-speed MCU. A custom PCBA is built with four TCS3200 ICs (Figure **3-1**), which allows for independent channel readouts in parallel, eliminating the need to capture using TDM.

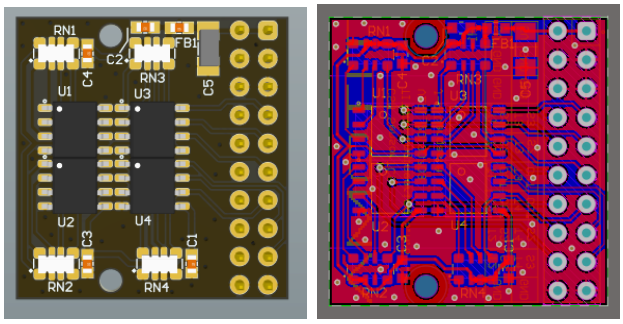

Figure 3-1: Custom made PCBA for TCS3200 implementation. Altium Designer 19

3.2.3 LED Light source arrays

The color sensors implemented feature uneven, non-linear and overlapping spectral responsivity curves. We implement a multi-spectral LED array in order to operate on the spectral regions of the sensor of higher sensitivity and lower overlap, as is further explained in Chapter 4. Multiple LEDs of the same wavelength must be used in order to reach the required radiated power. Several factors are relevant for the electro-mechanical design of the light sources.

Mechanical considerations

The largest mature fruits measure 20mm in average, to support fruits of this size a 22mm conductor is used. The fruits will have displacement in two axis with respect to the center of the sensor array, as a result a light beam greater than 25mm is created so it can fully illuminate the fruit regardless of position.

Optical considerations

The amount of light reaching the fruit must be maximized in order to ensure highest possible resolution of the sensor output. It is desirable to use LEDs with highest radiated power within electrical and financial constraints. Hotspots would result on fruits being incorrectly assessed as having a greater component of a given color. Thus, uniformity of light must be guaranteed. This can is achieved by:

- Using multiple LEDs of the same type in symmetric arrays.
- Using of wide emission angle LEDs.

Commercial considerations

Candidate LED wavelength selection is limited by availability of commercial solutions. LED wavelength is defined by the spectral emission properties of the semiconductors used. Gallium nitride (InGaN) is used in the 395nm to 530nm, Aluminum indium gallium phosphide (AlInGaP) is normally used in the 565nm to 645nm, and gallium arsenide (AlGaAs) on the 660nm to 680nm region [6].

Electrical considerations

Commercial LEDs operate in the voltage range of 1.5V to 4.5V. Current consumption ranges from 0.1mA to over 1A. When the operating current rises, the rating of all components must be increased, resulting in higher component costs. Higher power LEDs allow for greater resolution of the sensor readings. If intensity saturates the sensors the LED luminous output can be modulated using PWM to reduce their intensity, hence higher power LEDs are preferred. High power SMD LEDs of up to 2.5W are selected for evaluation.

Light source PCBAs

Light source wavelength modifications produce different classification results, this is due to the quantum efficiency characteristics of the sensors, and the spectral reflectance of the fruits. A custom PCBA featuring 13 different wavelength LEDs arranged in a radially symmetric pattern is designed using Altium Designer 19.3. and prototyped (Figure **3-2**).

The final design features 13 different wavelengths plus a white, for a total of 14 different emitters. The PCB receives 12V external power from a commercial power supply. It features a power driver stage (BJT + FET) which is controlled directly by the MCU connected through the breakout board.

The LEDs are laid out in a radially symmetrical pattern in order to maximize beam uniformity. Even though small footprint LEDs are used, not all emitters can be concentrated in a small enough area, which means LEDs closer to the center will radiate with higher power to the target than those on the edges. LED selection is tightly coupled with sensor performance; thus, the best LED configuration is found using wrapping methods of classifier evaluation through supervised machine learning. Chapter 3 explores the LED selection process in detail.

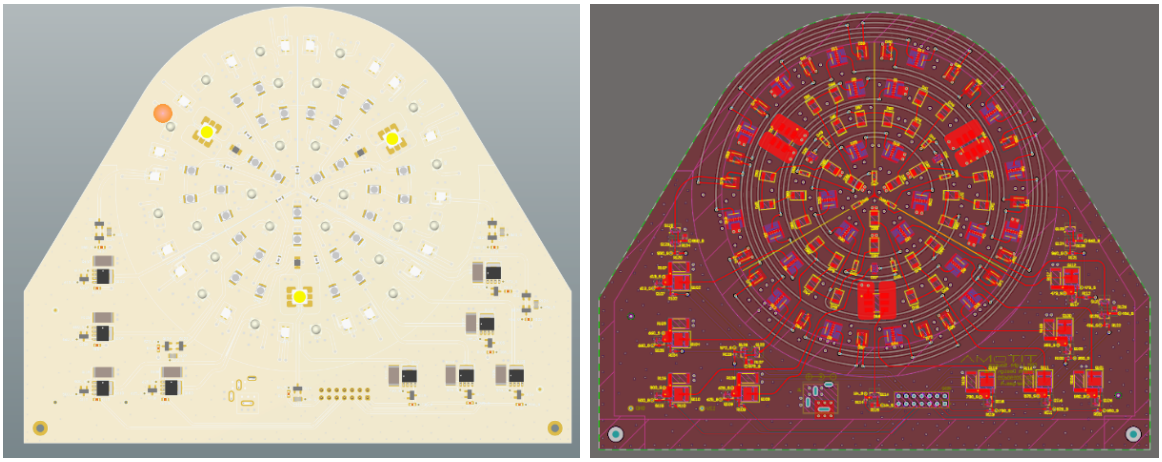

Figure 3-2: Custom PCBA for LED light source implementation. Altium Designer 19 View

Light uniformity stabilization system

Due to the size and optical characteristics of the LEDs it is not possible to guarantee an equal irradiated power from each emitter which generates a hotspot problem as seen in figure **3-4**. In order to resolve it, several constructions are studied and reflecting hemisphere and mirror array offers the best ratio of uniformity to power loss. This configuration can be seen in Figure **3-4**. The reflective arrays were designed in PTC CREO 5.0 and 3D printed. The internal reflective adhesive material used ensures that all beams of light hitting the target have previously bounced in at least two surfaces. The multiple bounces lowers the irradiated power arriving at the fruits by extending the total travel of the light beam, this goes against the design objectives but it increases the uniformity across different light sources.

Two mirrored structures of light sources and sensors were built in order to ensure scan of the complete fruit surface (Figures **3-5**, **3-6**, **3-7**). Thanks to the low cost of the sensors and

Figure 3-3: Hotspots produced by light source PCBA with reflective cone

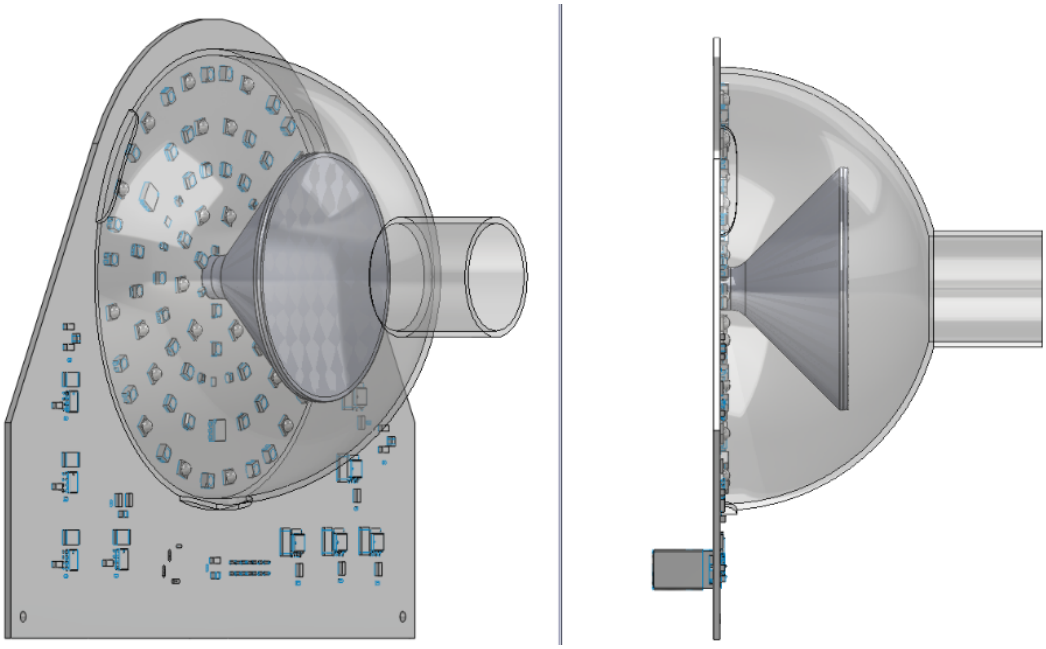

Figure 3-4: Light source PCBA with reflective baffle and hemisphere

light emitters it is possible to replicate the array without compromising the cost efficiency requirements.

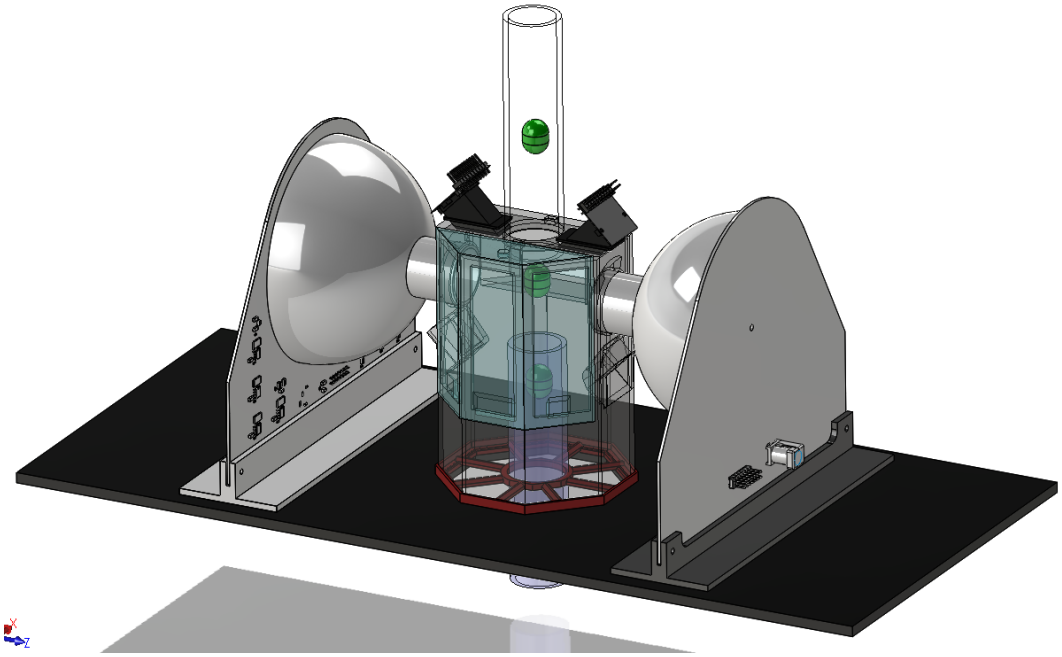

Figure 3-5: Light source and sensor acquisition optical system

3.2.4 MCU

The real-time and low-cost requirements can be met without compromise by 32 bit MCUs in the market from diverse manufacturers. The high-performance ARM Cortex-M4 STM32F429 (STMicroelectronics, Geneva, Switzerland) features the required processing speed and peripherals to execute sensor captures at intervals shorter than 1ms. The MCU can exceed 100MHz and offers analog and digital input and output interfaces for diverse LED and sensor control while executing real time data capture and transfer.

Peripherals used (Figure **3-9**):

- UART and I2C Serial Interfaces: sensor communication
- USB: PC communication
- ADC: Analog sensor data acquisition
- GPIOs with Interrupts and Capture Compare: Frequency sensor data acquisition
- GPIOs with PWM: LED intensity control

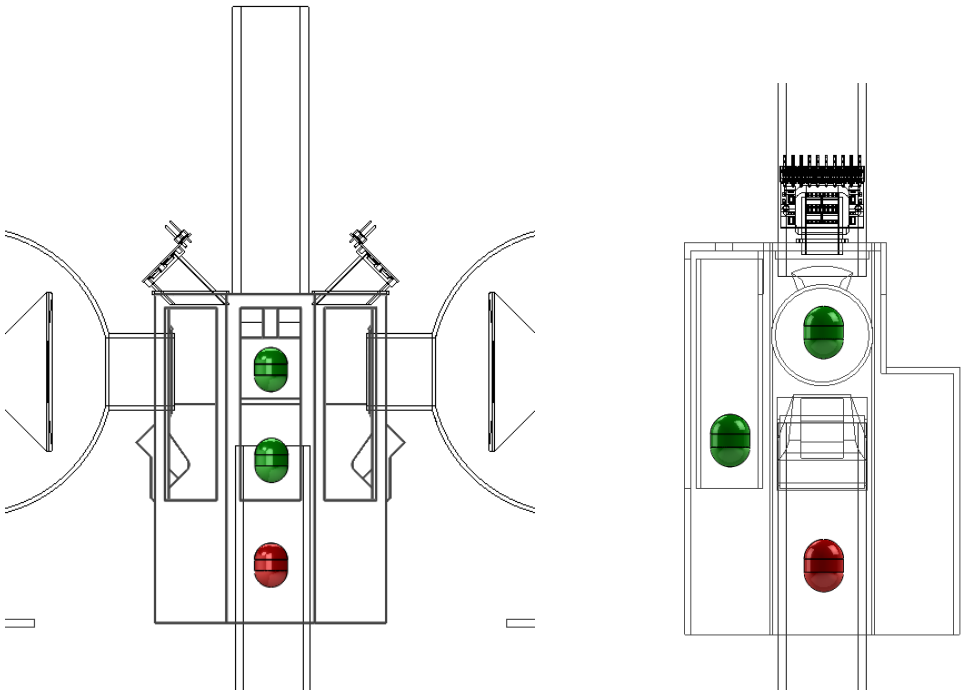

Figure 3-6: Light source and sensor acquisition optical system. Cross view

- Internal Timers: Capture compare and PWM control
- DMA: Accelerated data capture with lower CPU burden
- Floating point operation unit: Classification mathematical operations.

The illumination LEDs are controlled via internal hardware PWM peripherals so as to minimize the STM32F429I load. The LED control lines are enabled by setting controlling timers. A separate timer controls each LED so they can all operate at different clock speeds if needed. The PWM operating frequency is set at 10,000Hz, which is the limit for the given MCU operating frequency for some of the PWM clocks. The PWM duty cycle is adjusted from 0 to 100%, which in turn adjusts the brightness of the LEDs.

One MCU UART serial port is connected to an FTDI UART to USB converter cable TTL-232R-3V3; this interface sends and receives serial data to a PC serial (COM) port.

A PC graphical user interface for hardware control and data acquisition in C++ is developed using Embarcadero Builder C++ 10.3. It serves as a control panel which allows configuration of system parameters and trigger transfer and reception of serial commands to MCU, allows selection of LEDs to be used, receives LED and sensor data and plots the results, and parses and saves capture data in readable text format for further analysis.

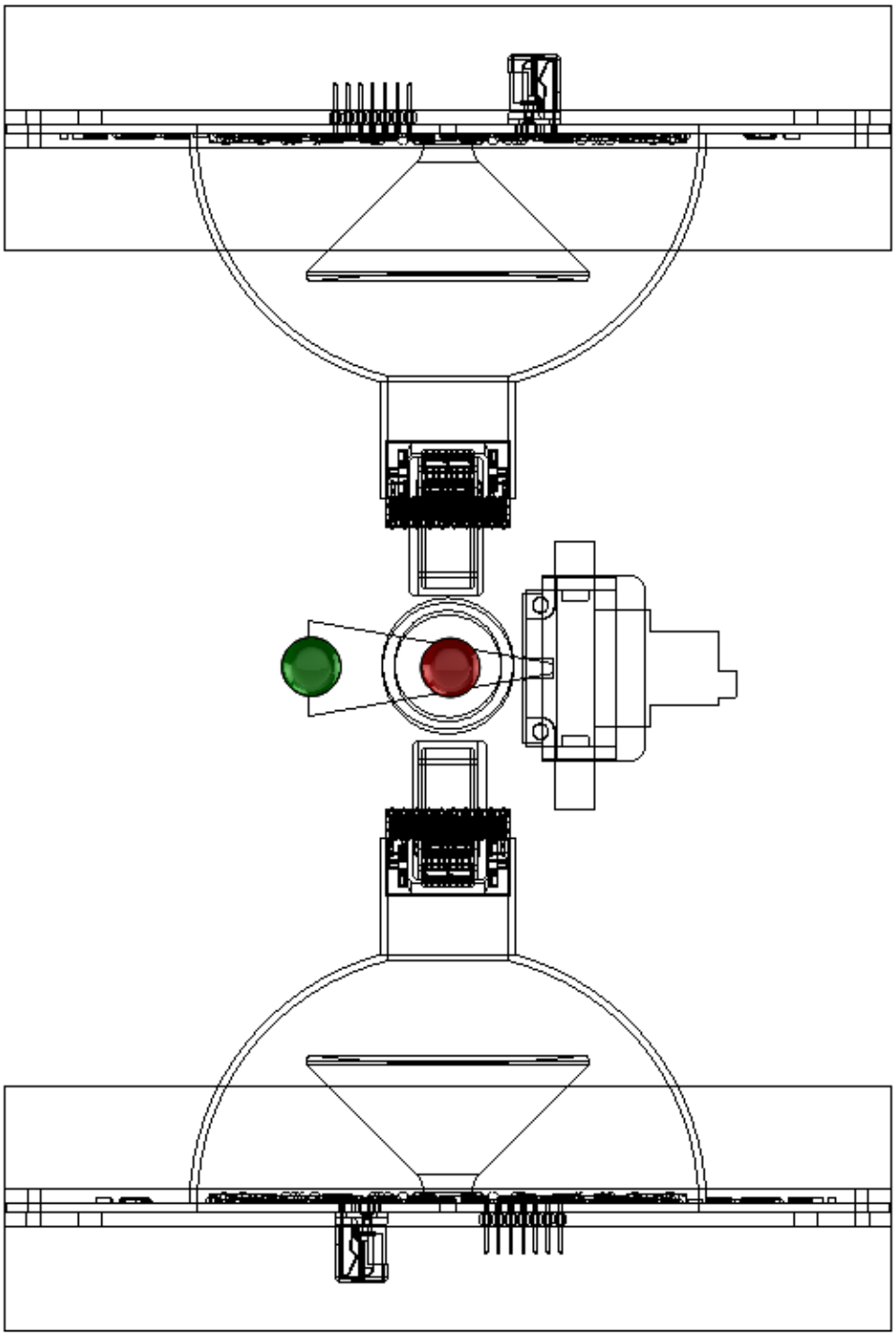

Figure 3-7: Light source and sensor system top view

PERIPHERAL	TYPE	REQUIREMENT	STM32F429	FUNCTION
ADC	Required	10+ bits >= 1 KSPS >= 6 channels	12 bits Up to 2400 KSPS 16 channels	APS5130 Sensors
Input Capture	Required	>=8	Up to 18 channels	TCS3200 Sensors
Hardware PWM	Required	>= 13	Up to 18 channels	LED driving
UART	Required	> 2 (PC + WiFi)	Up to 4 UARTs Up to 6 USARs	AS7262 / PC / WiFi
I2C	Required	>=1	Up to 3 I2C	VELM6040 Sensor
GPIOs	Required	>= 10	> 20 (after other functions satisfied)	Motor, detectors, Buttons, spare
LCD	Preferred		1 (STM32F429DISC1)	Improve debugging
Clock Freq	Preferred	>= 100 MHz	Up to 180 MHz	
Math	Preferred	Floating point or DSP	FPU core	Classifier decision function

Figure 3-8: MCU processing requirements vs. STM32F429 features

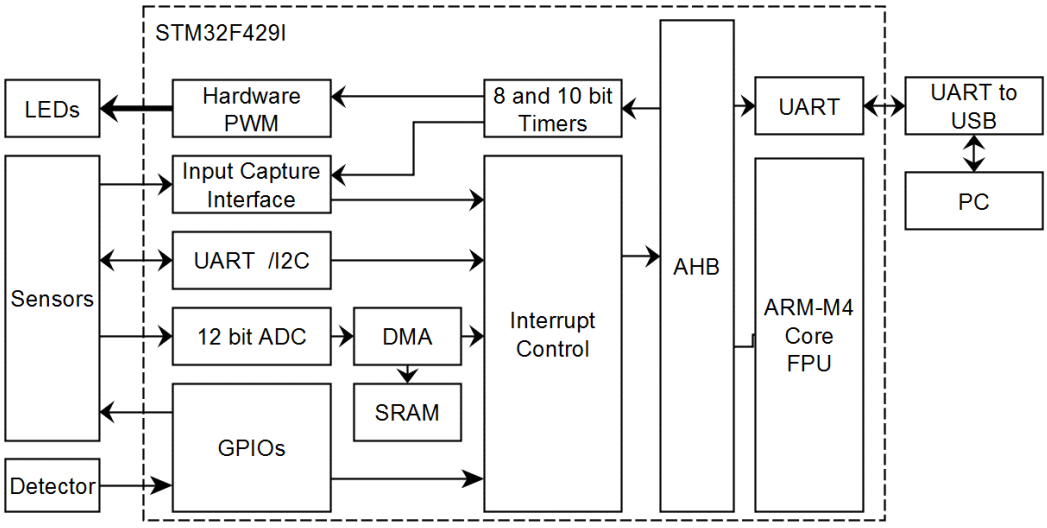

Figure 3-9: Block diagram of STM32F429I peripherals used

The imported data is then analyzed in Python 3.7 using scikit-Learn v0.20.3 [29], which allows for capture plotting, data mining and classifier training. The objective is to train the classifiers to recognize specific colors, hence the fruits are selected to have uniform coloration. Each fruit sample is labeled as green, yellow and red. Some coffee varieties such as Yellow Caturra remain yellow as they become ripe (Figure **3-10**) making it confusing to implement the commonly used terms of under mature, mature and over-mature.

Figure 3-10: Yellow Caturra variety coffee fruits, which remain yellow after being fully ripe.

3.3 Sorting

3.3.1 Classifiers training

The acquired data is imported, smoothened and filtered for outliers, as classifiers are highly sensitive to noise. Synthetic data points are generated to accelerate the algorithm convergence through the batch learning process. Feature extraction of non-linear combinations are produced so linear regression classifiers can classify nonlinearly separable datasets.

The data is separated in a train and test subset using a $0.75 : 0.25$ split, so each classifier is cloned and trained on 75% of the data and then evaluated on the remaining 25%. Throughout the process, dataset sizes fed to each classifier are optimized through learning curve analysis. Different configurations of hyper parameters are tested through 5-fold cross validation using Scikit GridSearchCV which yields the best performing configuration. The complete pipeline is plotted on Figure **3-11**
Different types of classifiers are trained, including LDA [38], Naïve Bayes, Random Forests, Linear Regression with L1 penalty (Lasso) [44] and Support Vector Machines. Multiple

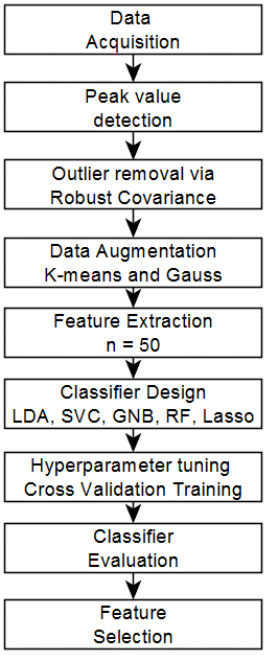

Figure 3-11: Data acquisition and Classifier training pipeline

inter-class solution strategies are analyzed: multiclass, one vs. rest and one vs. one. LDA is selected as the main classifier thanks to its training speed and implementation complexity. Chapter 4 describes the classifier training process in detail.

The features used are further analyzed in order to lower the complexity of implementation, by removing the characteristics which add the least amount of information to the decision-making process. Two features suffice to obtain accurate classification; however, the best overall LDA performance is achieved with 14 features. Chapter 5 describes the feature selection process in detail.

A LDA classifier with one vs. one multiclass solution strategy consistently ranks high in separation accuracy. Figure **3-12** shows perfect accuracy (dashed line) throughout 30 prediction cycles of 40 fruits per class. The colored lines show classifier certainty on the prediction of each class.

3.3.2 Classification

After optimization, the classifiers decision making function is implemented on the STM32F429I. When a new fruit is detected, the sensor data is obtained and filtered, the characteristics of interest are calculated and three different classifiers (one per class) are evaluated. The analysis and decision module compares the result and confidence of each of the classifiers

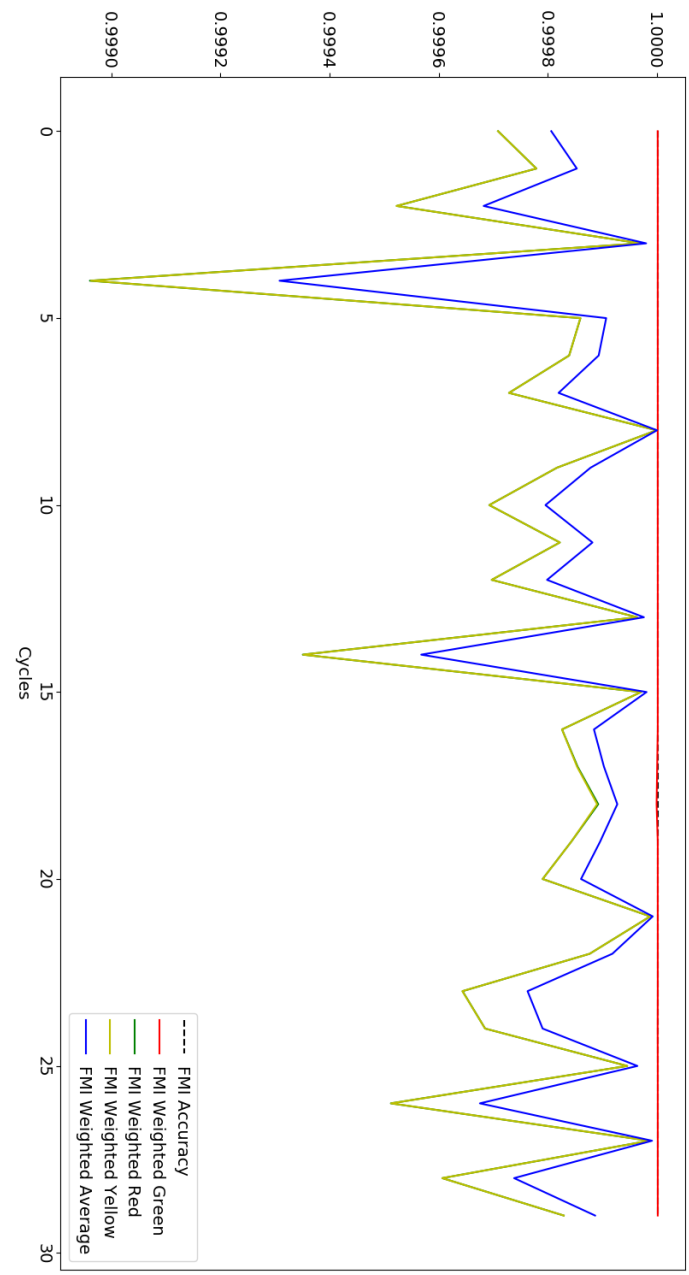

Figure 3-12: LDA OvO classifier performance per class, test repeated 30 times

and determines the result (Figure **3-13**).

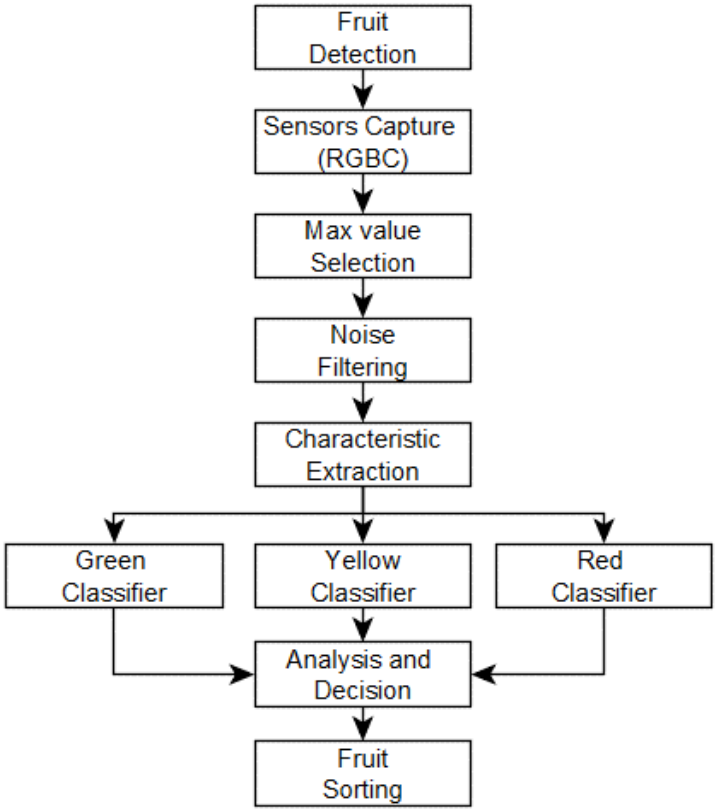

Figure 3-13: Visual inspection capture, analysis and decision flow.

3.3.3 Ejection driver system

The pneumatic electro-valve ejection system is outside the scope of this document. A a simple BJT and FET power control stage is used to trigger a high speed valve which permits a jet of air to re-direct sorted fruits.

4 LED selection for optimal sensor discriminant capacity

A standard color analysis set-up is comprised of a light source, which can be a multispectral one where multiple different narrowband LEDs are activated at different time intervals, or a wide-band one where a single white light source is kept activated, and a color sensor or spectrometer. In the aforementioned cases, the efficiency of the system is derived from its capacity to extract information at key wavelengths where the chemical composition of the object being scanned presents different reflectance characteristics.

Discrete color sensors are manufactured based on a photodiode that converts light to current. Photodiodes are designed to be sensitive in the range of wavelengths visible to the human eye. Optical filters can be used to obtain different color channels, these filters are built using glass or plastic elements installed in front of the photodiode, or dyes deposited on the photodiode surface. The frequency response of the filters varies both in bandwidth and magnitude (Figure **4-1**). Furthermore, the manufacturing process has tolerances which can produce value deviations from those referenced in the sensors literature.

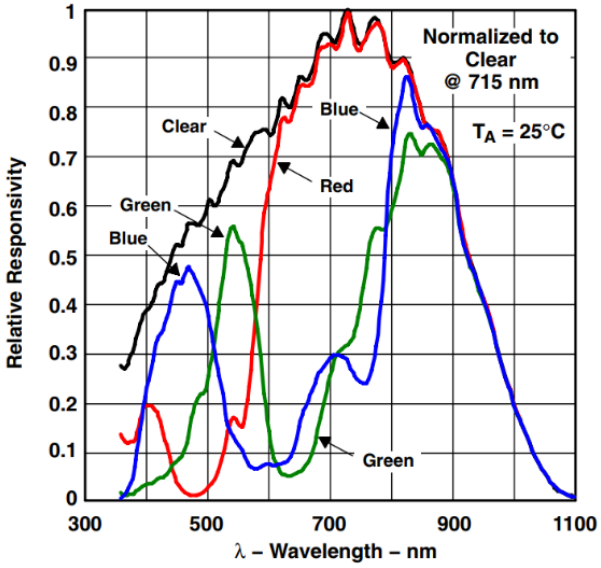

Figure 4-1: TCS3200 photodiode spectral responsivity.

A typical white LED presents asymmetric radiation peaks in several wavelengths (Figure **4-2**), which can coincide with overlapping regions of the sensor channels.

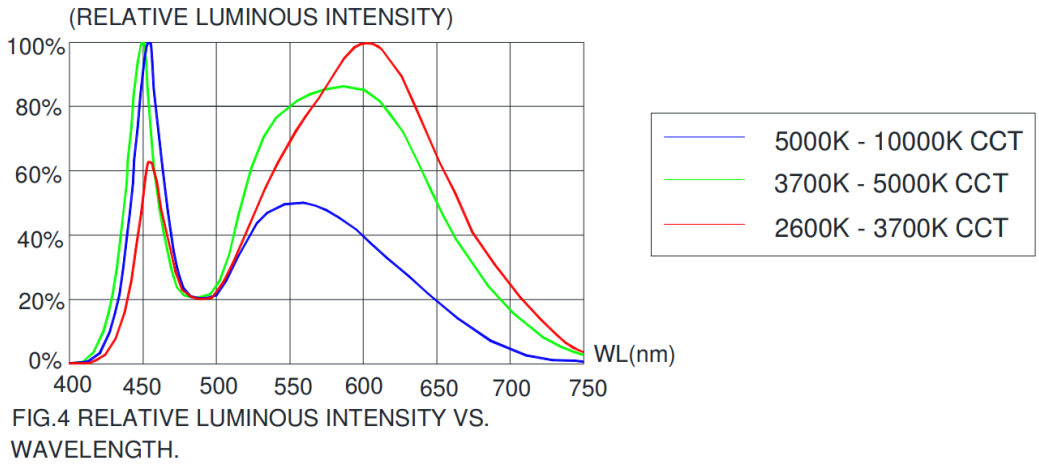

Figure 4-2: CLP6B-WKW White LED luminous intensity vs Wavelength profile.

The use of narrowband LEDs allows fine tuning of specific wavelengths and bandwidths to ensure that application-specific captures are executed while exciting only one sensor channel at a time.

4.1 Candidate sensors

Without loss of generality and in order to simplify the data analysis and processing requirements complexity and time, discrete color sensors are selected instead of CMOS or CCD arrays. Table 4.1 shows the candidate sensors.
AS72622 (ams AG, Styria, Austria) and VEML6040 (Vishay Intertechnology, Malvern, USA) are color sensor with analog to digital conversion stages and APS5310 (Kingbright, Taipei, Taiwan) is a color photodiode sensor with analog output, but it is TCS3200 (ams AG, Styria, Austria) a color to frequency converter, the selected device due to capture speed. Existing PCBAs are used for evaluation of AS7262 and VELM6040 (Figure **4-3**). Custom PCBAs are designed and prototyped for TCS3200 and APS5310 (Figure **4-4**).

4.2 Candidate LEDs

Although definitions vary through different authors, a LED with spectral bandwidth of less than 30nm is usually classified as narrowband. LED availability and die development are

Sensor	**Interface**	**Max Datarate (bps)**	**Acquisition Time (ms) (1)**	**Spectral Sensitivity (cnt/mw.cm2)**	**Channel Separation**	**Cost (US$)**	**Dimensions (mm x mm)**
AS7262	UART	115200	>75	45	6 channels, no overlap	$3.12	4.7 x 4.5
VEML6040	I2C	400000	>60	56-96	3 channels, overlap	$1.05	2 x 1.25
TCS3200	Frequency	N/A	<1 (2,4)	330-400	3 channels, overlap	$1.85 x 4 (3)	4 x 5 (3)
APS5130	Analog	N/A	<1 (3)	6	3 channels, overlap	$0.89	4 x 3

Table 4-1: List of tentative sensors considered for the research

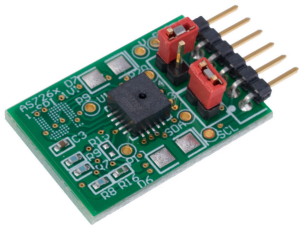

Figure 4-3: AS7262 development kit.

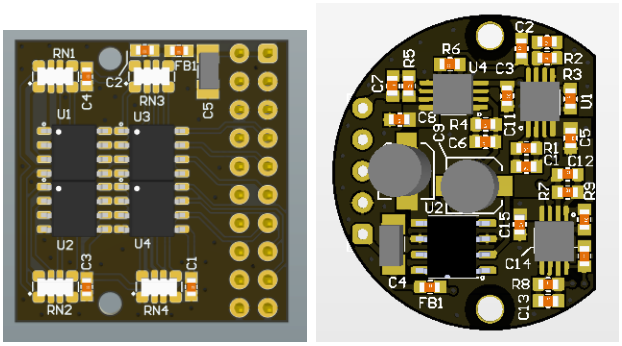

Figure 4-4: TCS3200 and APS5130 custom PCBAs. Altium Designer 19 View.

driven by commercial needs, which is in turn limited by the properties of the chemical compositions used to achieve emissions on specific wavelengths. A subset of available narrowband LEDs spanning from 390 to 950 nm are initially selected for the study (Table 4.2).

A LED driver PCB is designed using discrete FET transistors where the LEDs are arranged in a radial pattern to ensure maximum light beam collimation, reducing the effect of generated hotspots, and a reflective structure with an integrating hemisphere are built in order to equalize the light irradiation density of the different LEDs, producing a more even light source beam. After initial light power and photodiode sensitivity tests, 7 LEDs are selected for further tests (Table 4.2).

4.3 LED selection strategy

A wrapper optimization method is one where the best system configuration is assessed by evaluating multiple set-ups and comparing the resulting classification scores. The objective is to find the wavelength or combination of wavelengths which produce the highest accuracy score, so it can be implemented on a MCU platform. This process is sensor-dependent. If the color sensing system or the light sources are modified, then the process loses validity and must be run again.

WL	Part Number	Manufacturer	Optical Power (lm)	View Angle (deg)	Spectral Bandwidth	Vf (V)	I (mA)	Electrical Power (mW)
390	QBHP684E-UV395BS	QT-Brightek (San Francisco, USA)	85	120	10nm	3.4	700	2380
450	150224BS73100	Wurth Elektronik (Niedernhall, Germany)	4.07	120	20nm	3.5	30	105
470	XPEBBL-L1-0000-00302	Cree Inc. (Durham, USA)	49	135	∼15nm	3.1	350	1085
500	L1C1-CYN1000000000	Lumileds (San Jose, USA)	69	170	30nm	2.6	350	910
525	XPEBGR-L1-0000-00F02	Cree Inc. (Durham, USA)	126	135	20nm	3.2	350	1120
574	SML-Z14M4TT86	Rohm Semiconductor (Tokyo, Japan)	0.5	100	30nm	2.1	50	105
590	XPEBAM-L1-0000-00801	Cree Inc. (Durham, USA)	77	130	10nm	2.2	350	770
613	XPEBRO-L1-R250-00B02	Cree Inc. (Durham, USA)	97	130	5nm	2.2	350	770
625	XPEBRD-L1-0000-00801	Cree Inc. (Durham, USA)	77	130	10nm	2.2	350	770
660	XPEEPR-L1-0000-00901	Cree Inc. (Durham, USA)	36.3	130	20nm	2.1	350	735
730	XPEBFR-L1-0000-00801	Cree Inc. (Durham, USA)	33.8	140	20nm	1.85	350	647.5
860	SFH-4250S	Osram Opto Semiconductors GmbH (Regensbug, Germany)	3	120	30nm	3.1	100	310
950	SFH-4240-Z	Osram Opto Semiconductors GmbH (Regensbug, Germany)	1.8	120	42nm	1.5	100	150

Table 4-2: Candidate LEDs.

WL	Part Number	Manufacturer	Optical Power (lm)	View Angle (deg)	Spectral Bandwidth	Vf (V)	I (mA)	Electr. Power (mW)
470	XPEBBL-L1-0000-00302	Cree Inc. (Durham, USA)	49	135	∼15nm	3.1	350	1085
500	L1C1-CYN1000000000	Lumileds (San Jose, USA)	69	170	30nm	2.6	350	910
525	XPEBGR-L1-0000-00F02	Cree Inc. (Durham, USA)	126	135	20nm	3.2	350	1120
590	XPEBAM-L1-0000-00801	Cree Inc. (Durham, USA)	77	130	10nm	2.2	350	770
613	XPEBRO-L1-R250-00B02	Cree Inc. (Durham, USA)	97	130	5nm	2.2	350	770
660	XPEEPR-L1-0000-00901	Cree Inc. (Durham, USA)	36.3	130	20nm	2.1	350	735

Table 4-3: Candidate LEDs after minimum radiated power testing.

Classification algorithms will obtain different results depending on the physical conditions of the samples, which are in turn modified by the type of light source used. A performance scoring system must be defined to quantify the classifiers performance across different light source set-ups, using the geometric means of precision and recall as well as the classifier level of prediction confidence.

Captures of 120 fruits, representing 3 maturation stages are made under 127 combinations of light sources available at maximum power. Data is imported, smoothened and filtered for outliers. The highest value of each capture is used. Synthetic data points are generated. This accelerates the batch learning algorithms convergence. Feature extraction combinations are used. This enables linear regression classifiers (such as lasso) to classify non-linearly separable datasets.Results are scored and analyzed in order to determine the light sources, classifiers and features with greatest classifying capacity.

4.3.1 Classifiers training

A multitude of classifiers is available through state of the art software packages, and predicting which of them would have the best performance through diverse datasets is not trivial. Different classifiers are selected (Table 4.3.1) and trained (Figure **4-5**) in order to maximize the possibility of a given light configuration to provide a high accuracy separable dataset .

Naïve Bayes Classifier (GNB)

This simple yet robust classifier is used as a benchmark. Due to the Gaussian nature of the data it is expected to obtain good separation results. A GNB is a simple to implement probabilistic estimation, and a partial implementation using Mahalanobis distance computed

Classifier	**Training complexity**	**Classifying performance**	**Feature importance rank**	**MCU implementation complexity**
GNB	Low	Good with Gaussian distributed variables	No	Mid
Lasso	Low	Linearly separable only	Yes (with shrinkage)	Low
LDA	Low	Flexible	Yes	Low
RF	High	Flexible	Yes	High
SVC	High	Good with not linearly separable	No	High

Table 4-4: Classifiers used for LED selection

from covariance matrices is trivial, which results appealing when considering the need to implement the final algorithm on an MCU

Linear Discriminant Analysis Classifier (LDA)

The linear discriminant analysis implements space transformations that maximizes the within class convergence and between class separation. It uses the Fisher score to assess accuracy and can be used to compare performances for different configurations. LDA assigns weight or importance to the coefficients (features) used during training, which permits evaluation and reduction to simplify the final classification algorithm through feature selection.

Lasso (Linear regression with L1 penalty)

Lasso is a linear regression classifier with shrinkage, which penalizes non-contributing coefficients, making them zero, resulting in a feature selection process at the same time. Its implementation complexity and feature importance are similar to those of LDA, with the added benefit of automatic feature selection through embedded reduction.

Random forests (RF)

One of the most robust state of the art classifiers are the random search forests. It is expected to produce close to ideal results for any given dataset. RFs are more complex to implement and substantially more power intensive. RFs also implement coefficient importance.

SVC

Support vector machine classifiers use various processing Kernels to achieve separation of non-linearly-separable features in the original feature space. This makes them a flexible option, and while they are the most computer intensive to train, and their learning complexity increases exponentially with the dataset size, they remain a high capacity classifier.

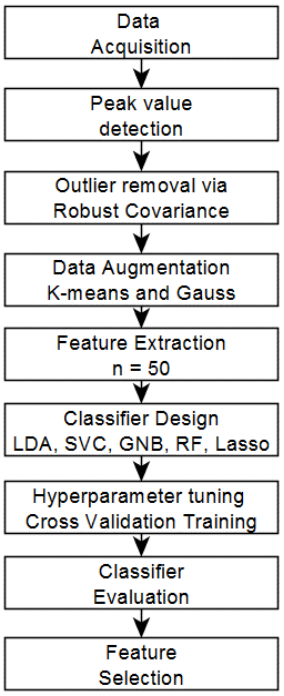

Figure 4-5: Data acquisition, processing and classifier training pipeline

When classifiers are trained and analyzed, the individual class performance is assessed, thus a single classifier is interpreted as 3 different classifiers, one per class.

4.3.2 Data acquisition

For the experiment, 120 coffee fruits representing and equally distributed along 3 different maturation stage classes, are selected (Figure **4-6**).

Some coffee such as Yellow Caturra and Yellow Bourbon varieties remain yellow as they become ripe, hence the use of maturation terms (under mature, mature and over-mature) is discouraged and instead each fruit sample is labeled as green, yellow and red.

A given combination of LEDs is turned on, and the fruits are dropped in front of the light source and sensor arrays. As the fruits pass in front of the sensor a bell-shaped graph is generated (Figure **4-7**). The maximum values are reached when the fruit has the highest

Figure 4-6: Sample subset of fruits used for tests.

alignment with the sensor. With proper synchronization, the timeframe of 5ms of highest alignment can be obtained (Figure **4-8**). The captures contain information on Clear, Red, Green and Blue channels of the TCS3200 sensor. As expected, the data sets produced by different light sources vary greatly (Figure **4-9**).

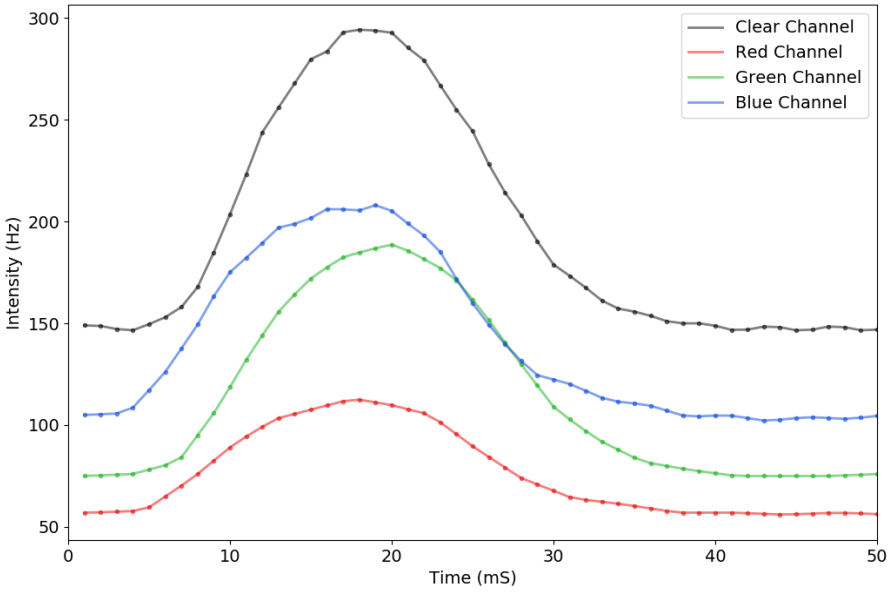

Figure 4-7: Sensor capture of a coffee fruit through a 60ms interval.

4.3.3 Data Noise

Several different factors contribute to data noise.

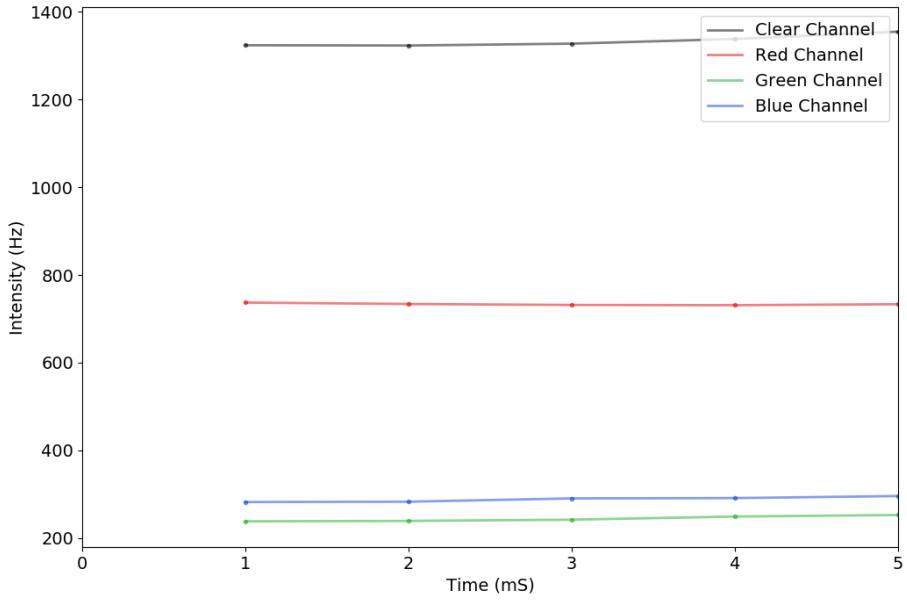

Figure 4-8: Sensor capture of a coffee fruit depicting the 5ms of highest alignment.

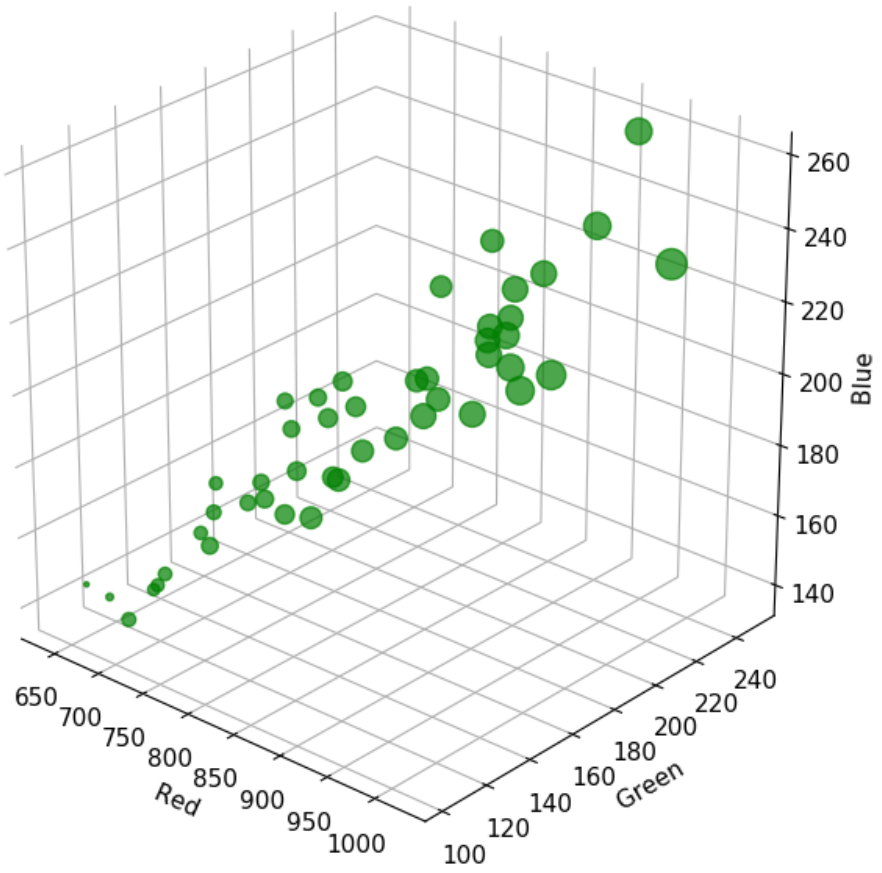

Figure 4-9: Green fruits, various sizes with '473 500 525 590 613 625 660' light source

Sensor size

For the TCS3200 implementation 4 sensor intergrated circuits are used per PCB in order to capture the 4 channels in parallel. This maximizes data acquisition speed, but yields a total array area of 15mm x 15mm. When captures are made, the fruit will not be precisely centered with all sensors at the same time, this results in some channels capturing higher value readings than others.

Fruit alignment

The fruits can measure up to 22mm, this requires a large conductor. Smaller fruits (5mm or maybe less) have a wide range of X and Y movement with respect to the sensor. While the light source system is designed to maximize uniformity, there will still be variations in captured values if the fruit is in the centered to the sensor, compared to the fruit being towards the edge. This is exactly the same effect as that described on the previous subsection.

To compensate for these effects, the maximum independent value per channel is used per capture, regardless of the moment in time when the capture was made, as it is expected to represent the moment of highest alignment.

Fruit speed

Various factors such as fruit size, quantity, density and humidity can impact the speed of the fruit when it passes in front of the sensor. Faster fruits (usually smaller) produce lower intensity readings, as the sensor has less time to reach stability.

Fruit size

Fruit size not only impacts speed but also the absolute reading values, as larger fruits reflect more light toward the sensor than smaller fruits. Fruit size is the greatest contributor to variance in within-class readings. Different size fruits of the same class stretch over an axis where the R, G, B, C ratios remain consistent (Figure **4-10**). Noise cannot be predicted nor fully suppressed because of its nature, but data can and must be filtered, as outliers can severely impact classifier performance.

4.3.4 Data Filtering

Classifiers are highly sensitive to outliers either due to training corruption when outliers distort population means and covariance, or evaluation errors when outliers are used to validate the classifier results yielding false negatives.

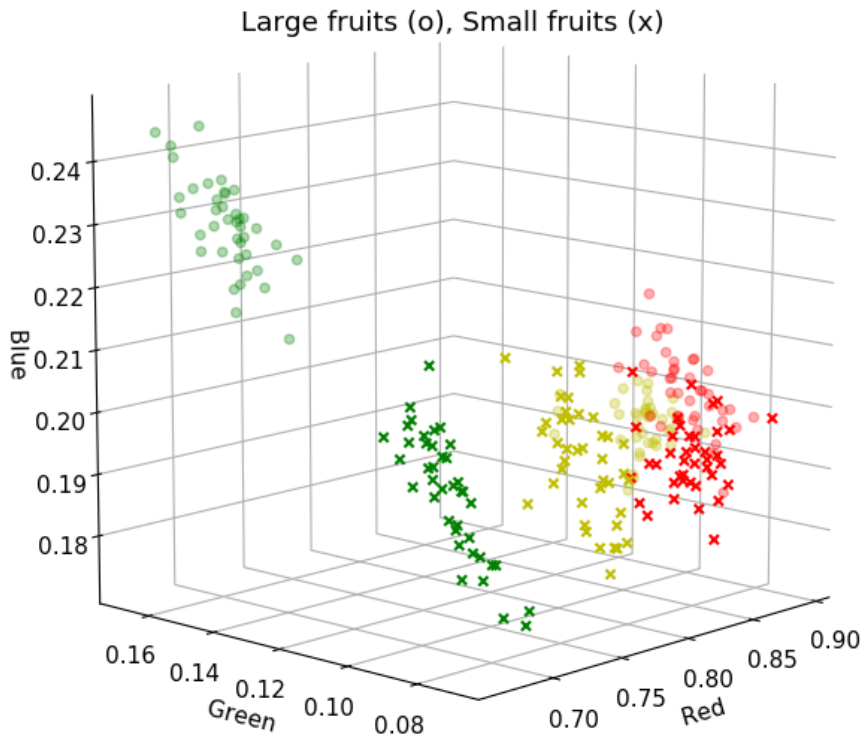

Figure 4-10: Small vs Large fruits with 473 525 660 light source.

The Gaussian nature of the data implies that outlier removal filtering can be efficiently achieved by training a robust covariance (elliptic envelope) detector before any further processing takes place. Covariance matrices are obtained per class, then, using a contamination factor of 5%, the samples with largest Mahalanobis distance to class centroids are removed.

4.3.5 Data Augmentation

Regression classifiers benefit from the use of synthetic data as this can speed up convergence by compensating for the low number of class samples. Two data augmentation methods are explored: generation of data using k-neighbors means, where a new data point is created from the means of every 2 existing points, and Generation of data using Gaussian model, built based on covariance matrices and random samples are generated from the distribution obtained.

4.3.6 Feature Extraction

Additional features are generated from the initial 4 sensor channel values, obtaining a total of 50 different features as non-lineal combinations of the base characteristics and color space transformations (HSV, SCT, CIExy). The complete list can be seen on Table 4.3.6.

ID	Name	Formula	ID	Name	Formula
0	X_max_clear	C	25	X_n_red_over_green	R/G
1	X_max_red	R	26	X_n_red_over_blue	R/B
2	X_max_green	G	27	X_n_green_over_red	G/R
3	X_max_blue	B	28	X_n_green_over_blue	G/B
4	X_max_red_over_green	R/(G+0.00001)	29	X_n_blue_over_red	B/R
5	X_max_red_over_blue	R/(B+0.00001)	30	X_n_blue_over_green	B/G
6	X_max_green_over_red	G/(R+0.00001)	31	X_n_red_times_green	R*G/C^2
7	X_max_green_over_blue	G/(B+0.00001)	32	X_n_red_times_blue	R*B/C^2
8	X_max_blue_over_red	B/(R+0.00001)	33	X_n_green_times_red	G*R/C^2
9	X_max_blue_over_green	B/(G+0.00001)	34	X_n_green_times_blue	G*B/C^2
10	X_max_red_times_green	R*G	35	X_n_blue_times_red	B*R/C^2
11	X_max_red_times_blue	R*B	36	X_n_blue_times_green	B*G/C^2
12	X_max_green_times_red	G*R	37	X_n_red_squared	R^2/C^2
13	X_max_green_times_blue	G*B	38	X_n_green_squared	G^2/C^2
14	X_max_blue_times_red	B*R	39	X_n_blue_squared	B^2/C^2
15	X_max_blue_times_green	B*G	40	X_n_red_cubed	R^3/C^3
16	X_max_red_squared	R^2	41	X_n_green_cubed	G^3/C^3
17	X_max_green_squared	G^2	42	X_n_blue_cubed	B^3/C^3
18	X_max_blue_squared	B^2	43	H	HSV - H
19	X_max_red_cubed	R^3	44	S	HSV - S
20	X_max_green_cubed	G^3	45	V	HSV - V
21	X_max_blue_cubed	B^3	46	Alpha	SCT - Alpha
22	X_n_red	R/C	47	Beta	SCT - Beta
23	X_n_green	G/C	48	CIE_x	CIE x
24	X_n_blue	B/C	49	CIE_y	CIE y

Table 4-5: Features generated through extraction process.

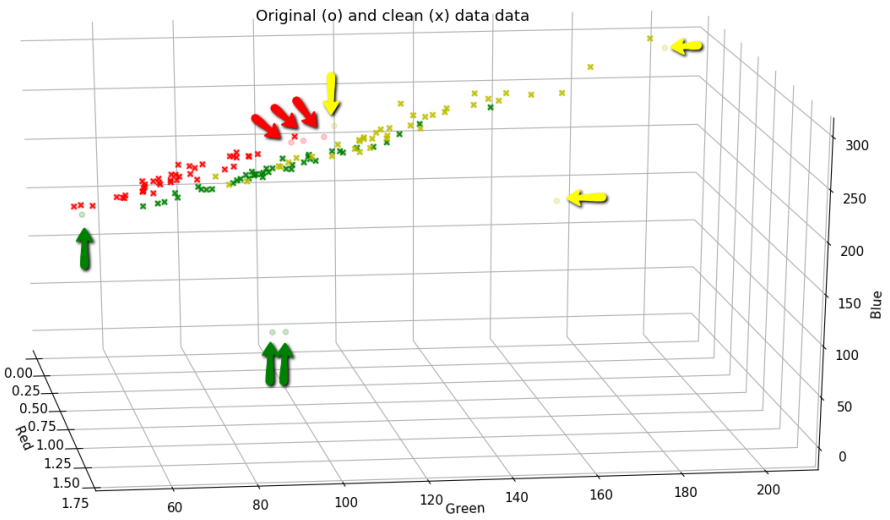

Figure 4-11: Original (o) and Clean (x) data before and after contamination removal process, with removed points marked by arrows.

4.3.7 Classifier training strategy

A multiclass solution strategy refers to how the classifier should attempt to define the likelihood of a sample belonging to any pair of classes. Different multiclass solution strategies increase the classifier performance, particularly in non-linearly separable cases. The solution strategies are implemented as custom classifier class wrappers on top of the standard Scikit methods.

- Multiclass (all vs all)
- One vs Rest (OvR)
- One v One (OvO)

In total 15 classifiers of 5 types with 3 solution strategies are trained using the K-fold technique with 0.75 to 0.25 train to test ratio. Each classifier is cloned and trained on 75% of the data and evaluated on the remaining 25%. The hyper-parameters are independently tuned for each classifier and each different configuration is cross validated 5 times using Scikit GridSearchCV, which selects the best performing classifier.
The training process described is repeated 10 times and the worst-case results are used, which prevents over-fitting.

4.3.8 Sample size analysis (learning curve)

Training 15 different classifiers with 127 configurations of light sources over multiple data folds and with 5 times cross validation is a resource intensive task. Furthermore, using

synthetic data points can lead to over-fitting. To address these concerns a learning curve analysis can be executed in parallel with classifiers training. The learning curves are analyzed both per class and averaged. The results help in finding the optimal sample size. This analysis shows that GNB and SVC do not benefit from data augmentation, while LDA, Lasso and RF do.

4.4 Performance evaluation metrics

The area under the curve of a receiver operating characteristic plot score is useful in assessing optimality a non-separable datasets and has thus gained wide approval in recent years (Figure **4-12**); however, in fully separable datasets, it is common to find perfect scores.

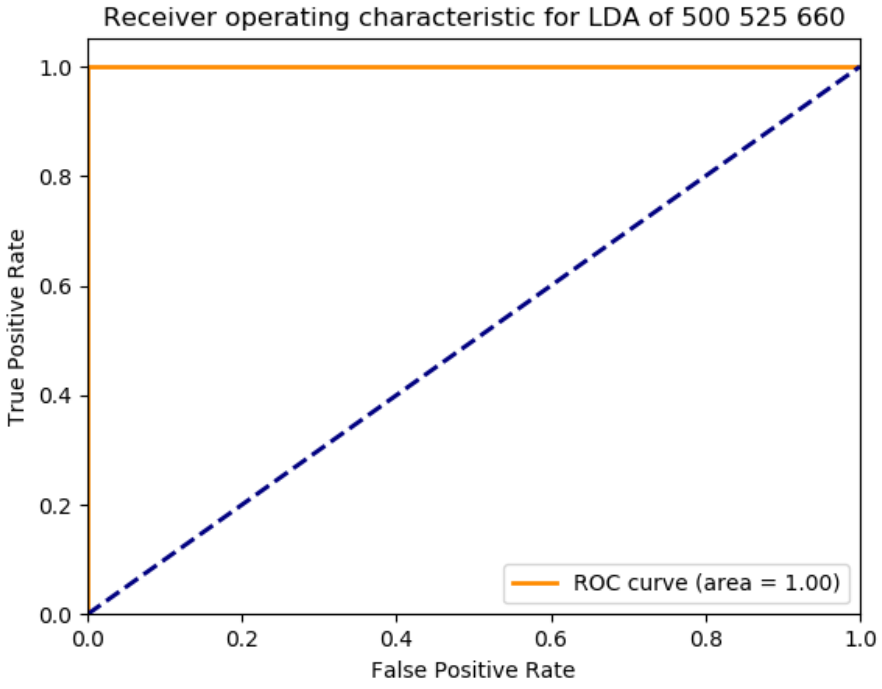

Figure 4-12: RoC plot and AUC index for a fully separable dataset with perfect accuracy

A different performance scoring metric is proposed. Each classifier training run yields a confusion matrix. From such confusion matrix, the geometric mean of precision and recall (Fowlkes Mallows index, FMI) can be obtained per class. This is the $FMI_{accuracy}$.

$$FMI_{accuracy} = \frac{TP}{\sqrt{(TP+FP)*(TP+FN)}}$$

Where

$FMI_{accuracy}$= Geometric mean of precision and recall
TP= true positive count
FP= false positive count
FN= false negative count

To analyze the multiclass problem, a probability confusion matrix can be built. For each classifier, the probability of each decision made is computed. The columns of the confusion matrix represent the real values and the rows represent the probability of the predictions Based on the probability confusion matrix, a $FMI_{probability}$ can be obtained. This probability represents the confidence of the predictions per class.

$$FMI_{probability} = \frac{TP_{probability}}{\sqrt{(TP_{probability} + FP_{probability}) * (TP_{probability} + FN_{probability})}}$$

Where
$FMI_{probability}$ = geometric mean of precision and recall of the probability confusion matrix
$TP_{probability}$ = true positive sum of probability confusion matrix
$FP_{probability}$ = false positive sum of probability confusion matrix
$FN_{probability}$ = false negative sum of probability confusion matrix

The $FMI_{weighted}$ per class is obtained as the multiplication of the accuracy FMI score and the FMI probability per class

$$FMI_{weighted} = FMI_{accuracy} * FMI_{probability}$$

This score captures both the accuracy (as the mean of precision and recall) and the confidence of the classifier. In other words, it not only scores whether a classifier can label test data accurately, but also how high the confidence is (how much margin of error exists).

4.5 Classifier performance comparison

The weighted score of the classifiers is plotted per class for every wavelength (Figure **4-13**, Table **4-6**). The light configurations are sorted in descending order of performance. The chart below shows the minimum (worst-case) weighted score across 10 different runs with different splits over the same wavelength. A heat map matrix is used where darker tone represents higher weighted score (accuracy). The classifiers are grouped together and results are split per class.

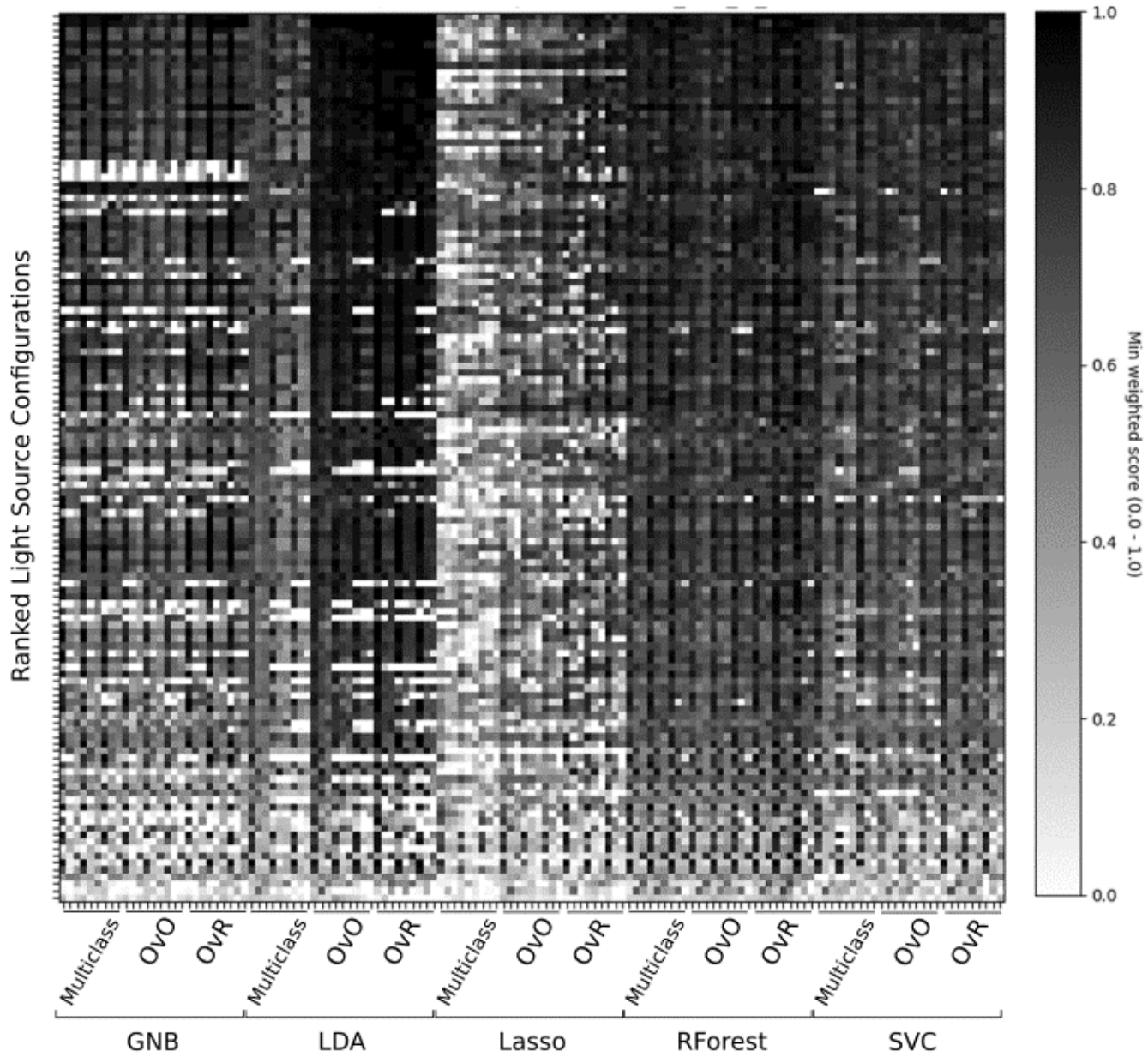

Figure 4-13: Minimum weighted score (FMI weighted) of 3 classes for 15 classifiers through 127 different light source configurations after 10 cycles. Columns from left to right: GNB, LDA, Lasso, RF, SVC, all with Multiclass, OvO and OvR strategies for Green, Red and Yellow.

Rank	Configuration	FMI Accuracy	Rank	Configuration	FMI Accuracy	Rank	Configuration	FMI Accuracy
1	500 525 660	1	44	473 500 525 590 613 625 660	0.948	87	500 590 613 660	0.873
2	473 500 525 625	1	45	473 500 525 590 613	0.948	88	473 613 660	0.866
3	473 525 590 625	1	46	525 590	0.947	89	473 525 590 625 660	0.865
4	473 500 590 613	1	47	473 525 590 613 625	0.947	90	473 500 613 625 660	0.858
5	473 525 660	1	48	473 500 590 613 625	0.947	91	473 590 613 625 660	0.851
6	473 500 590 625	1	49	500 525 625 660	0.946	92	500 525 590 660	0.845
7	473 525 590 660	1	50	500 525 590 613 625 660	0.946	93	473 500 613 660	0.844
8	500 590 660	1	51	473 500 590 625 660	0.946	94	500 613 625 660	0.843
9	473 500 613	1	52	525 590 625 660	0.945	95	525 613 625 660	0.84
10	473 525 625 660	1	53	473 500 525 613 660	0.944	96	500 590 613 625	0.838
11	473 525 590 613	1	54	500 525 613 625	0.944	97	500 525 613 660	0.836
12	473 525 613	1	55	473 500 525 625 660	0.94	98	590 625	0.83
13	473 525 613 660	0.999	56	590 660	0.936	99	500 625	0.83
14	525 660	0.999	57	500 613 660	0.935	100	500 590 613 625 660	0.826
15	500 590 625	0.999	58	525 613	0.929	101	590 613 625 660	0.822
16	473 500 525 660	0.999	59	473 590 625 660	0.925	102	500 590	0.822
17	473 500 525 590 613 625	0.999	60	473 525 590	0.925	103	473 613 625	0.819
18	473 590 613 660	0.996	61	473 590 625	0.923	104	500 525 613 625 660	0.809
19	525 590 613 660	0.995	62	473 590 613	0.921	105	473 500 590 613 660	0.798
20	473 500 625	0.995	63	473 500 613 625	0.92	106	473 525	0.793
21	473 525 590 613 660	0.993	64	500 613 625	0.912	107	473 660	0.788
22	500 525 625	0.989	65	590 625 660	0.911	108	473 613 625 660	0.787
23	500 525 590 613 625	0.987	66	473 525 625	0.911	109	473 500 525	0.766
24	525 590 625	0.977	67	473 590 613 625	0.907	110	590 613 625	0.764
25	500 525 590 613	0.974	68	525 590 613	0.907	111	473 625 660	0.708
26	500 525 590 625	0.971	69	500 525 590	0.906	112	473 500	0.69
27	500 590 613	0.959	70	500 660	0.906	113	590 613	0.684
28	500 525 613	0.957	71	473 525 613 625 660	0.905	114	613 660	0.674
29	525 625	0.957	72	525 613 625	0.905	115	613	0.673
30	525 625 660	0.956	73	473 500 525 613 625	0.905	116	473 613	0.673
31	525 590 660	0.955	74	473 500 590 613 625 660	0.904	117	525	0.646
32	500 625 660	0.954	75	473 500 590 660	0.903	118	625 660	0.645
33	525 613 660	0.954	76	473 500 525 590 625 660	0.899	119	473 625	0.635
34	473 525 613 625	0.952	77	500 525 590 625 660	0.897	120	625	0.628
35	473 500 660	0.952	78	473 525 590 613 625 660	0.895	121	473 590	0.615
36	590 613 660	0.952	79	473 500 525 590 613 660	0.895	122	660	0.605
37	473 500 525 590	0.951	80	473 500 525 613 625 660	0.895	123	613 625	0.601
38	500 525 590 613 660	0.951	81	473 500 590	0.894	124	500	0.573
39	473 500 525 590 660	0.95	82	473 590 660	0.894	125	590	0.547
40	500 590 625 660	0.95	83	473 500 625 660	0.894	126	500 525	0.512
41	473 500 525 590 625	0.949	84	525 590 613 625 660	0.887	127	473	0.351
42	473 500 525 613	0.949	85	500 613	0.879			
43	525 590 613 625	0.949	86	613 625 660	0.874			

Table 4-6: Light source configurations rank.

Light sources	FMI accuracy	FMI weighted
500 525 660	1.00000000	0.88301608
473 500 525 625	1.00000000	0.87022115
473 525 590 625	0.99999998	0.80164741
473 500 590 613	0.99999982	0.76443844
473 525 660	0.99999783	0.82279834
473 500 590 625	0.99999725	0.78090578
473 525 590 660	0.99996925	0.85017478
500 590 660	0.99994971	0.85564721
473 500 613	0.99993132	0.71431167
473 525 625 660	0.99987944	0.81068977

Table 4-7: Best light source configurations for all classifiers

Certain light configurations such as 525 590 625nm exhibit poor performance with the baseline GNB classifier, but it still performs well with other classifiers; this supports the need for multiple classifier training in order to assess overall performance.

All classifiers exhibit much better separation of Green compared to Yellow and Red. The best wavelengths are those that maximize Red and Yellow separation. Overall LDA classifiers perform above average.

Some independent white spots suggest that at least one of the training runs had very poor performance, which can be explained by noise or data corruption. Larger patches can expose real low performance problems.

4.5.1 Best overall light source configurations

Light source configuration scores per class are averaged and sorted. An average score of 1,000 shows the classifier is able to perfectly separate all 3 classes. The second sorting parameter is the weighted score which allows to sort the light configuration by the level of certainty of their predictions (Table **4-7**).

Only two light source configurations obtained perfect separation results with at least one of their classifiers for all three classes, however several configurations are very close to being perfect, in these circumstances the FMI weighted index can offer more information about their confidence. Derivations of the best light sources do not always produce better results, which suggests that including additional light sources is destructive, as expected due to the overlapping nature of the sensor channels.

Light sources	FMI accuracy
473 500 525 625	0.929
500 525 660	0.913
500 525 590 613	0.906

Table 4-8: Best light sources for GNB.

Light sources	**FMI accuracy**
500 525 660	0.929
473 525 590 660	0.919
473 500 525 625	0.918

Table 4-9: Best light sources for LDA.

4.5.2 Best wavelength per classifier

The light source configurations produce diverse data distributions which favors different classifiers. The best overall light configuration is always in the top 3 of all classifiers, confirming the notion that feature selection trough filtering should not take place before classifier training. Tables **4-8**, **4-9**, **4-10**, **4-11**, **4-12** show the best wavelengths per classifier.

4.5.3 Best wavelength per class

A similar analysis can be conducted per class. Here we see again that the best overall classifiers are those that perform better at separating Red and Yellow, as Green separation is easily achieved in multiple datasets. Tables **4-13**, **4-14** and **4-15** show the best wavelengths per class.

4.5.4 Number of wavelengths used and performance

The TCS3200 sensor is unable to produce separable datasets using only one wavelength, however two wavelengths suffice, as can be seen with 525 660, but the best results are obtained with several 3 to 4 wavelength conditions and multiple configurations can achieve this.

Light sources	**FMI accuracy**
500 590 660	0.84
525 590 660	0.788
525 660	0.782

Table 4-10: Best light sources for Lasso.

Light sources	FMI accuracy
525 660	0.936
500 525 660	0.93
473 500 525 625	0.925

Table 4-11: Best light sources for RF.

Light sources	FMI accuracy
473 500 525 625	0.899
500 525 660	0.892
525 660	0.851

Table 4-12: Best light sources for SVC.

Light sources	FMI accuracy
525 660	0.923
525 590 660	0.92
473 500 525 625	0.912

Table 4-13: Best light sources for Green Fruits.

Light sources	FMI accuracy
500 525 660	0.887
500 590 660	0.85
473 500 525 625	0.848

Table 4-14: Best light sources for Red Fruits.

Light sources	FMI accuracy
500 525 660	0.856
473 500 525 625	0.85
525 660	0.831

Table 4-15: Best light sources for Yellow Fruits.

The inclusion of more light sources (more than 4) ensures more stable results which performs reasonably across the board, but it also does not offer better performance than other combinations with less light sources. This implies the additional lights do not add more information, but do operate on overlapping regions of the sensors. Each new light source spreads the dataset in one, two or three axes, as can be seen on Figure **4-14** where one wavelength is added at a time. These additions can be destructive, for example adding 473 to 500 525 660 generates overlaps in the previously separable red and yellow (Figure **4-15**).

4.6 Analysis of classifiers using best light source datasets

A more detailed analysis is conducted over 30 cycles with all 45 classifiers on the 500 525 660 and 473 500 525 625 datasets (Figure **4-16**, **4-17**, **4-18**).Using the best two light source configurations GNB, LDA and Random Forests show highest accuracy, with LDA recurrently obtaining perfect results. The OvR and OvO variants of the classifiers always obtain better results than the standard multiclass scenario, which confirms the expected better performance of ensemble classifiers. We found no perceivable advantage in classifier performance by using synthetically generated data points.

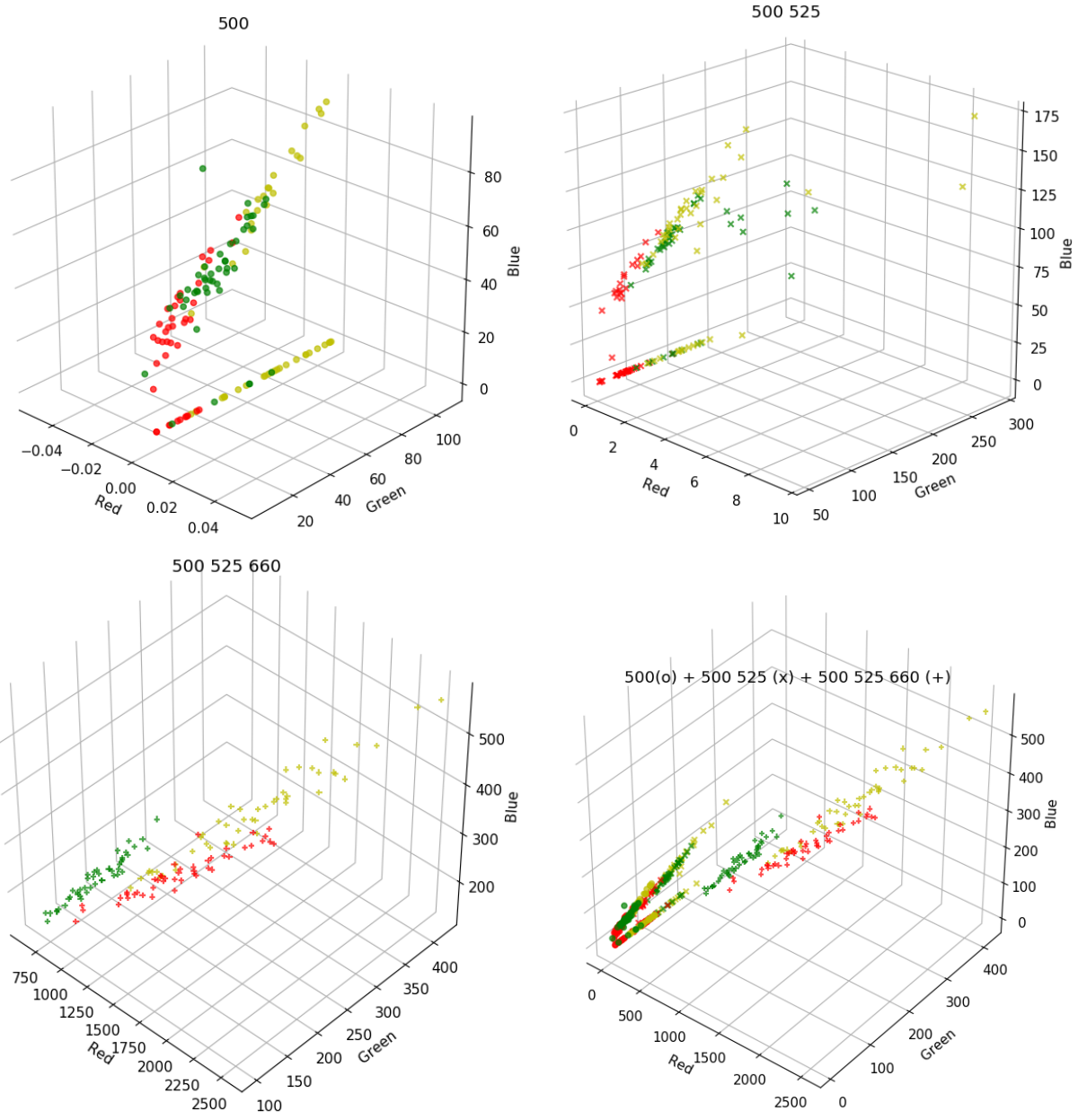

Figure 4-14: RGBC plots for 500, 500 525 and 500 525 660 light source configurations.

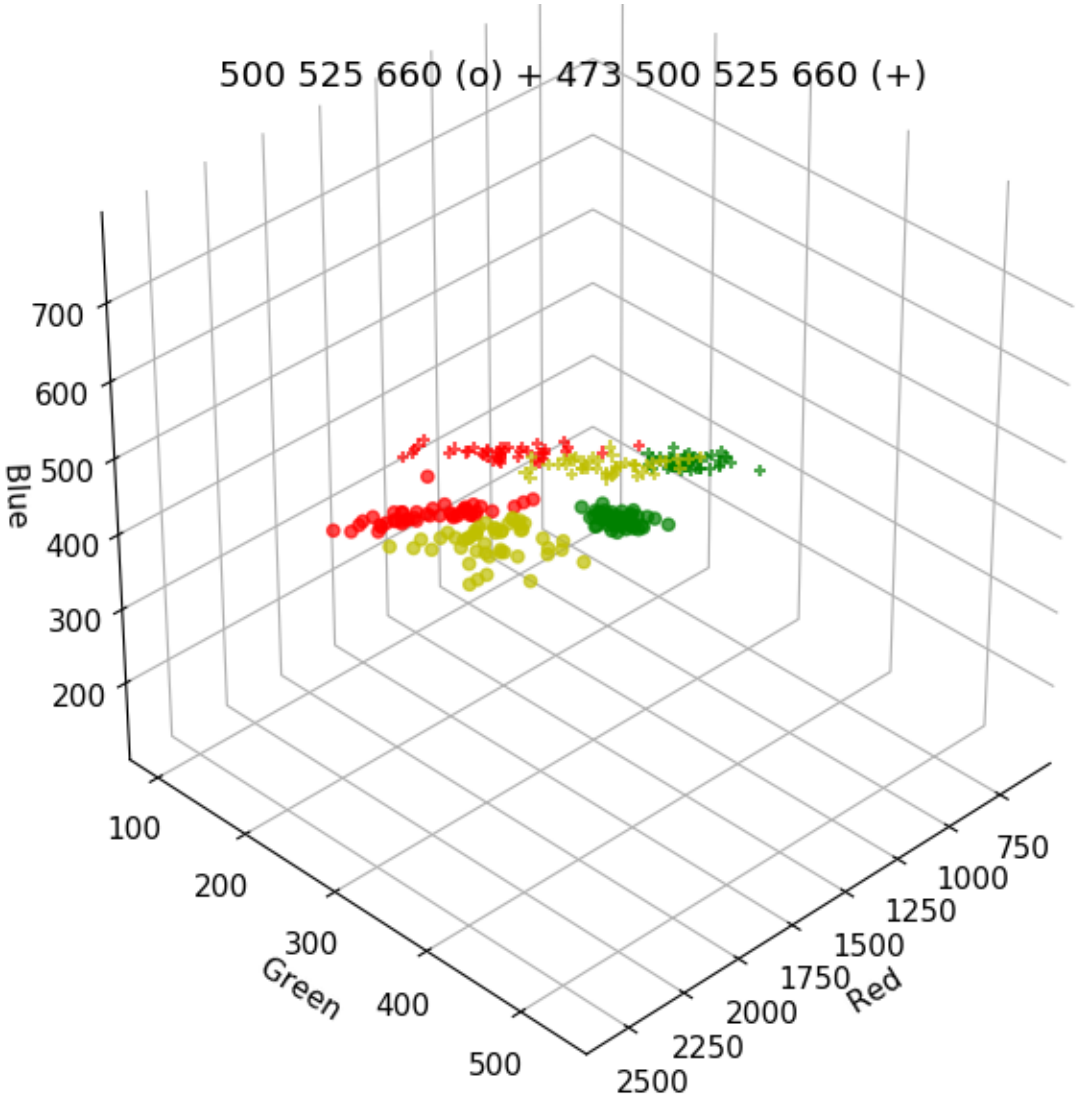

Figure 4-15: 500 525 660 and 473 500 525 660 datasets.

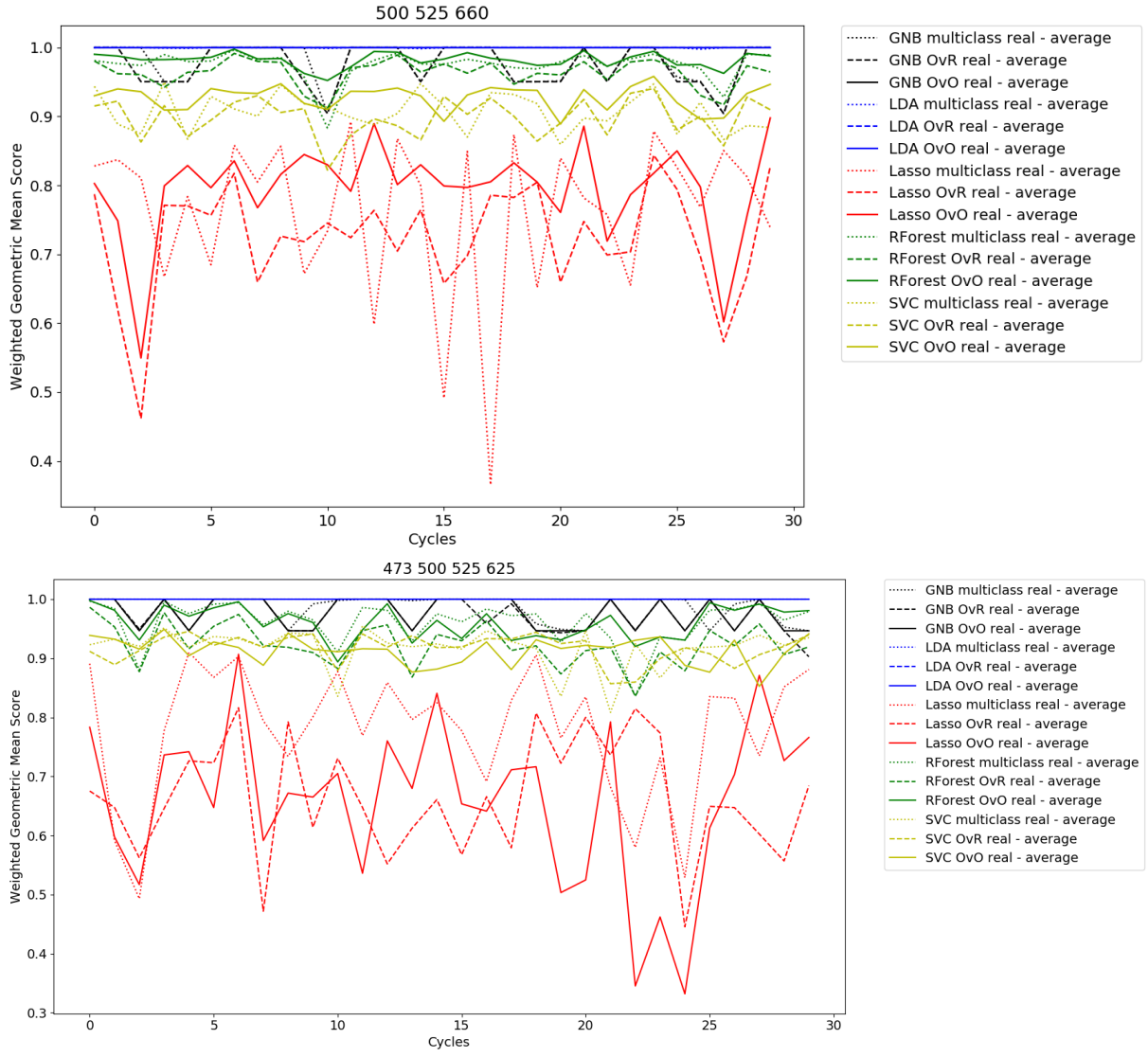

Figure 4-16: FMI weighted over 30 cycles.

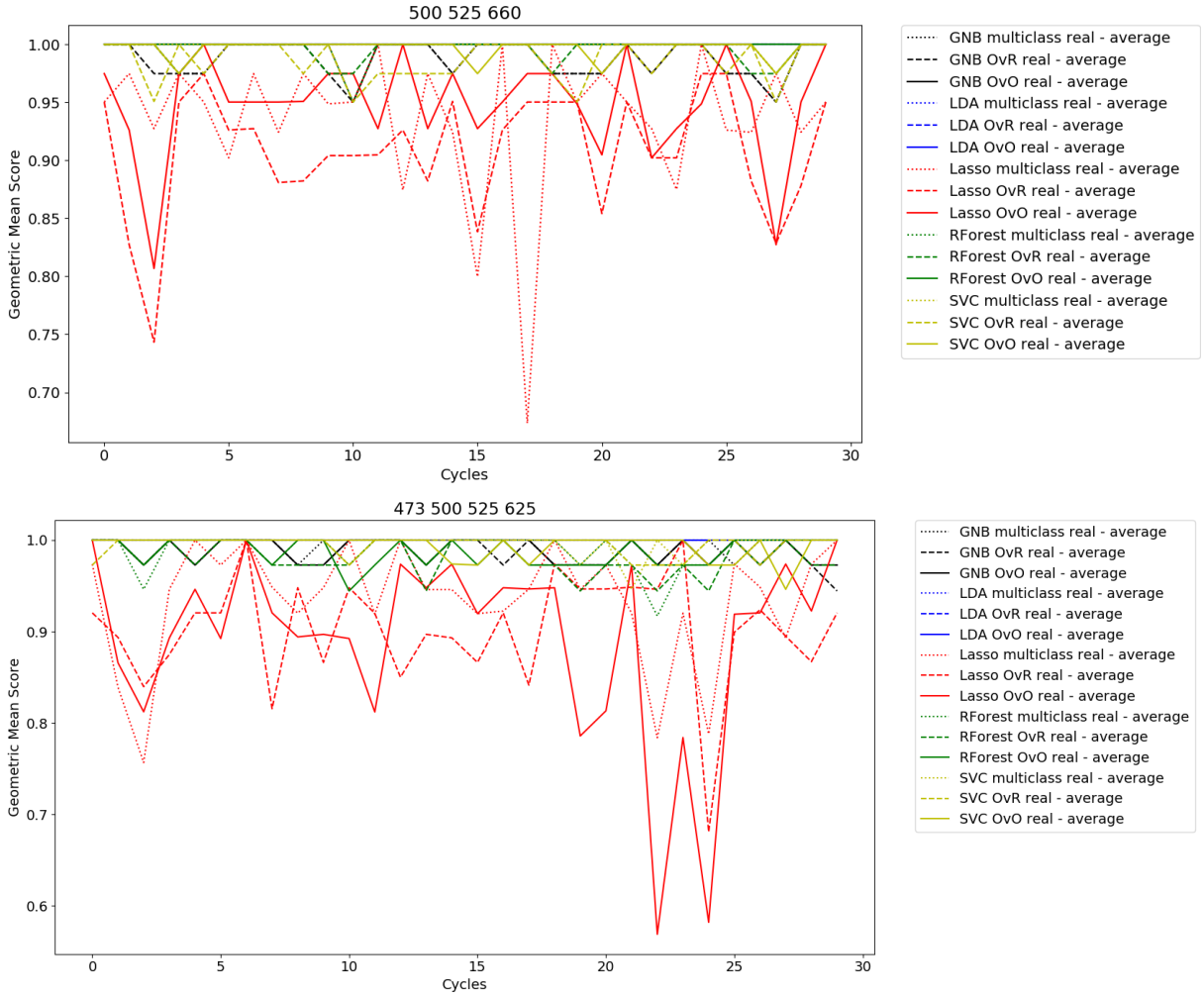

Figure 4-17: FMI accuracy over 30 cycles.

Figure 4-18: Classifier performance comparison (FMI accuracy).

5 MCU Algorithm optimization through feature selection

A color classifier has been trained using machine learning methods as part of the visual inspection unit of the coffee sorting system. Such classifier must be implemented on a STM32F429 MCU for real time classification and sorting in under 5ms, making it desirable to minimize the processing time and resources, which can be achieved through a feature selection process where the number of inputs used is reduced.

5.1 Classifier MCU implementation complexity

Different classification algorithms have been trained to obtain high accuracy in prediction results. We analyze their implementation complexity.

5.1.1 GNB

GNB prediction is based on the estimation of the joint log likelihood. Class priori information is equal (33% for the 3 class problem), thus it reduces to:

```
JLI = []
for i in range(np.size(self.classes_)):
    JLI = - 0.5*np.sum(np.log(2. * np.pi * self.sigma_[i, :]))
    JLI -= 0.5*np.sum(((X - self.theta_[i, :])**2)/(self.sigma_[i, :]), 1)
```

5.1.2 LDA

The LDA classifier class of scikit-learn is built upon a linear regressor whose prediction result is derived from the minimum distance of a given sample to the decision or boundary hyperplane. This operation is a dot product between the weight vector and the input (features) values. The following section explains in greater detail.

```
scores = safe_sparse_dot(X, self.coef_.T, dense_output=True) + self.intercept_
class = argmax(X . Wt)
```

5.1.3 Lasso

Lasso classifier is also built as a linear regressor, the prediction decision function is based on the distance to a hyperplane. This is also calculated as a dot product between the weight vector coefficient and the input (feature) values, as is for LDA.

```
scores = safe_sparse_dot(X, self.coef_.T,dense_output=True) + self.intercept_
```

5.1.4 RF

RForest must implement and store in memory hundreds of individual estimators. These estimators independently calculate the probability of each sample belonging to a given class. An averaged probability is produced for the complete classifier. The forests trained during the research used 100 estimators and 16 decision nodes.

5.1.5 SVC

The prediction is based on the decision function of SVC, which consists of the execution of the decision function used by the Kernel for each sample. The implementation complexity derives from the specific kernel, which changes depending on the best result obtained through cross validation.

5.2 LDA mathematical framework

The Linear Discriminant Analysis is originally a feature space reduction transform technique through eigenvalue decomposition, which allows analysis of components importance by projecting a dataset onto a new space composed by linearly independent combinations of features. In the case where class densities are multivariate normal with equal covariance matrices the samples fall in clusters of equal size and shape. [38]

However, Scikit-learn also implements a classifier based on the underlying mathematical construct which it also calls a Linear Discriminant Analysis. This LDA classifier has a linear decision boundary generated using Bayes' rule to fit class conditional densities, assuming the classes follow a normal distribution and share the same covariance matrix. The knowledge of a covariance matrix allows us to calculate the dispersion of data in any direction or subspace.The specific configuration of the LDA classifier used does not optimize the cost function using eigenvalue decopmosition (which yields the fisher scores), instead it uses least squares optimization to find a weight vector and intercept. This implementation results in a simpler and faster decision function comprised of a dot product and an addition.[29]

We can analyze the Linear Discriminant Classifier starting from Bayes' minmum-error-rate classification discriminant function:

$$g_i(\mathrm{x}) = \ln p(\mathrm{x}|w_i) + \ln P(w_i)$$

For multivariate normal densities, when covariance matrices are identical and class priors are the same it reduces to:

$$g_i(\mathrm{x}) = -\frac{1}{2}(\mathrm{x} - \mu_i)^t \Sigma^{-1} (\mathrm{x} - \mu_i)$$

Expanding the term and eliminating the i independent terms we get the discriminant function:

$$g_i(\mathrm{x}) = \mathrm{w}_i^t \mathrm{x} + w_i$$

Where the weight vector of the cost function can be solved by multiple methods. Using the standard eigenvalue decomposition requires computation of with class and between class scatter matrices, which yields the criterion function known as the Rayleigh quotient. A second method is least squares optimization, where the resulting cost function is similar but avoids the computation of scatter matrices, this makes it a more efficient method for training at the expense of not being able to execute space state transforms. The weight vector is calculated as:

$$\mathrm{w}_i = \Sigma^{-1} \mu_i$$

and the intercept as

$$w_{i0} = -\frac{1}{2} \mu_i^t \Sigma^{-1} \mu_i + \ln P(w_i)$$

which yields the aforementioned decision function

$$\mathrm{y} = argmax(\mathrm{W}^t \mathrm{x} + w_0)$$

The python implementation of uses a single dot product plus an addition. The C implementation uses a recursive loop for each of the characteristics, executing a multiplication and an addition per characteristic.

5.3 Feature selection for MCU implementation

The result of the feature selection process is a rank of characteristics which contribute the most to the decision functions of the classifiers. This analysis can be interpreted as the minimum number of features which yield comparatively similar results, by discovering features which can be eliminated with no accuracy loss. The reduction of the required number of features to consider directly impacts the amount of mathematical operations to be executed, speeding up the real-time decision making process.

Feature selection can be implemented by embedded, wrapper or filter methods Li [19] sums them up in detail. Embedded methods perform feature elimination during the training process, such as in linear regression with L1 penalty (Lasso). This method is not applicable for other classifiers such as LDA. In wrapper methods, the characteristics are added / removed and then evaluated assessing classifiers performance, making them the most accurate alternative as they are evidence-based, but are substantially more computationally intensive as they are NP-Hard. Filter methods attempt to reduce the characteristics by evaluating indices which predict individual or collective performance. Filter methods are classifier independent, instead they evaluate the dataset characteristics.

5.3.1 Feature selection by filter methods

We studied multiple filter methods, but only those with best performance results were included on the report.

Pearson correlation index

In general, it is desired to preserve features that are highly correlated to class labels, while at the same time are not correlated with other features as these add little or no information to the final decision function.

A Pearson correlation analysis is conducted. The objective is to find and eliminate the features with highest correlation to each other. The class is added as a column on the features matrix, and a correlation plot is built (Figure **5-1**).
A filter is designed to remove features with a Pearson correlation score greater than 0.9999. 37 features remaining after Pearson filter: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 13, 16, 17, 18, 19, 20, 21, 22, 23, 24, 31, 32, 34, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 48, 49.

Similarity based methods

Similarity based methods analyze the between-class and within-class data similarity in order to obtain discriminant scores. We analyze Fisher score [34], trace ratio [9], relief [39] for the

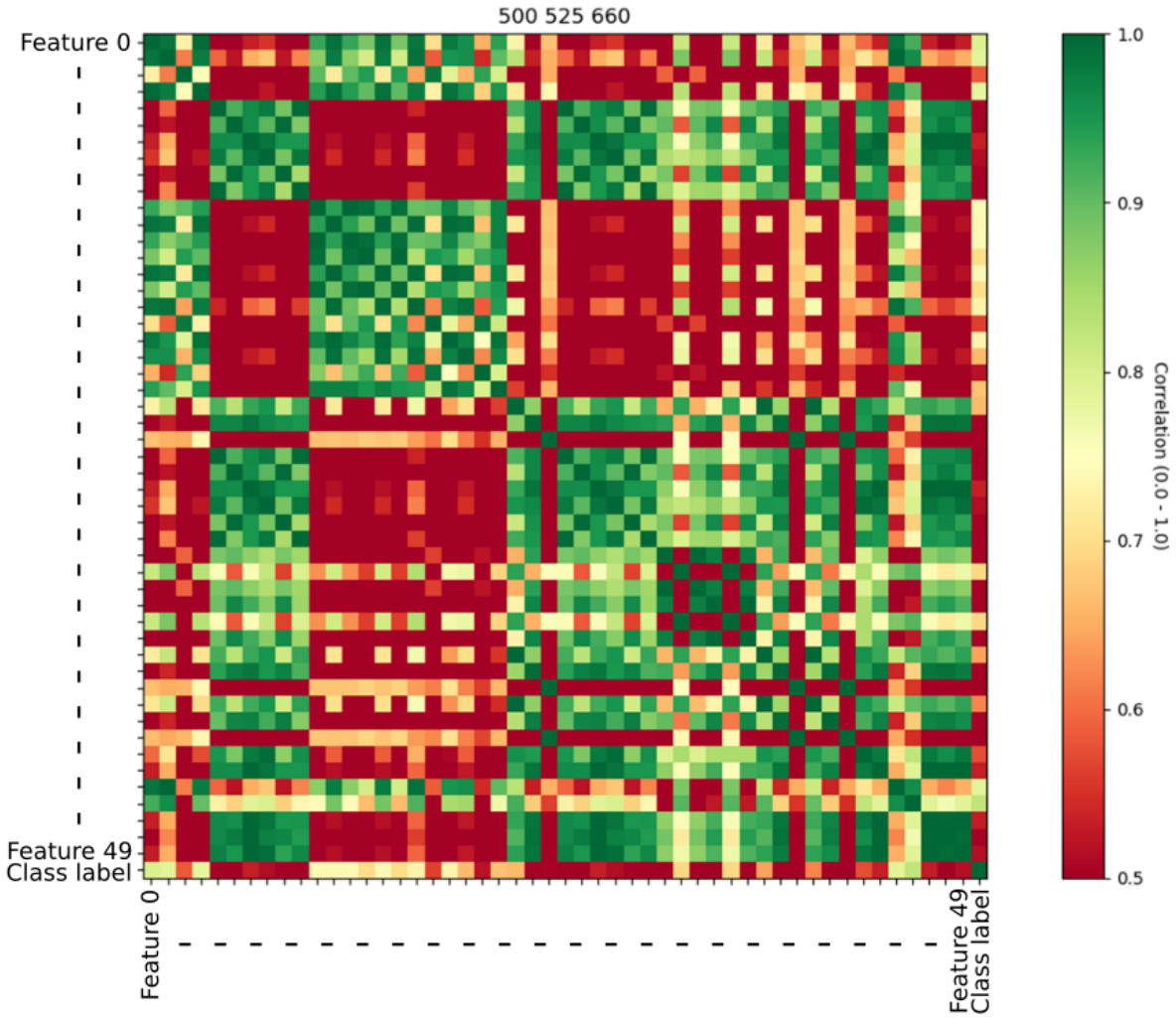

Figure 5-1: Feature correlation coefficient chart.

dataset (Table **5-1**).

Fisher score selects the features such that variation within class is small, and variation between classes is large. The trace ratio criterion evaluates the ratio of the traces of within and between scatter matrices, thus providing a measure of within class similarity and between class dissimilarity. ReliefF attempts to find the best inter-class separation features.

Information theoretical based methods

Information theoretical based methods attempt to find the features which maximize the mutual information of features with regards to class label. Most information theoretical based methods derive from Shannon information equation for conditional information gain, and attempt to find the terms which add more novel data, maximizing the relationship to the label, and minimizing the correlation to other variables.

$$J_{CMI}(X_k) = I(X_k : Y) - \beta \sum_{X_j \in S} I(X_j : X_k) + \lambda \sum_{X_j \in S} I(X_j; X_k | Y)$$

Where J_{CMI} denotes the common mutual information between a given feature and the label, and α and β are penalty terms for feature correlation and inter-class correlation.

Within the methods analyzed, the highest scores are obtained using MIM (Mutual Information Maximization) [20], CIFE (Conditional Infomax Feature Extraction) [22] and CMIM (Conditional Mutual Information Maximization) [11] (Table **5-2**).

Statistical based methods

Statistical based methods attempt to find the features which can generate a statistical difference in classification. We calculate F-score [46] and GINI [16] (Table **5-3**). F-score tests whether a subset of samples is able to separate classes by evaluating the distance between the means and the ratio of the variances for a multiclass problem. GINI Index also attempts to find whether a feature is able to separate instances from different classes.

Analysis of filter methods

The methods utilized yield characteristic ranks. An iterative process is proposed, where the classifier is evaluated eliminating the worst ranked characteristic while monitoring the accuracy; the objective is to find at which point the accuracy stops being perfect. This would become the cutoff point for the least number of features that could be used.

rank	fisher	trace ratio	reliefF	rank	fisher	trace ratio	reliefF
1	30	30	30	26	26	26	24
2	9	9	9	27	5	5	39
3	22	22	43	28	31	31	31
4	7	37	32	29	33	33	33
5	28	43	35	30	3	3	42
6	43	7	7	31	8	8	16
7	37	28	28	32	29	29	41
8	40	40	22	33	16	16	2
9	25	25	37	34	36	36	11
10	4	4	40	35	34	34	14
11	46	46	25	36	14	11	34
12	49	49	4	37	11	14	36
13	47	47	49	38	18	18	18
14	44	44	46	39	10	10	26
15	6	6	47	40	12	12	5
16	27	27	44	41	13	19	10
17	35	35	6	42	15	13	12
18	32	32	27	43	19	15	8
19	48	48	48	44	2	21	29
20	23	23	23	45	21	2	15
21	38	38	1	46	17	17	13
22	1	1	45	47	20	20	19
23	45	45	0	48	24	24	17
24	41	41	3	49	39	39	21
25	0	0	38	50	42	42	20

Table 5-1: Characteristic rank by similarity based methods score

rank	**MIM**	**CIFE**	**CMIM**	**rank**	**MIM**	**CIFE**	**CMIM**
1	43	43	43	26	31	5	12
2	6	4	4	27	33	21	14
3	27	24	34	28	5	35	15
4	47	2	45	29	26	26	16
5	7	45	46	30	8	0	17
6	28	10	0	31	29	34	18
7	9	39	2	32	3	36	19
8	30	17	3	33	18	46	20
9	4	13	5	34	21	23	21
10	25	42	22	35	11	38	25
11	44	20	23	36	14	41	26
12	49	11	24	37	1	48	27
13	22	12	31	38	16	22	28
14	37	8	32	39	19	37	29
15	40	1	10	40	13	40	30
16	48	15	11	41	15	49	33
17	23	3	13	42	45	44	35
18	38	31	48	43	10	25	36
19	41	16	9	44	12	9	37
20	34	14	1	45	2	30	38
21	36	29	49	46	17	7	39
22	46	33	47	47	20	28	40
23	0	18	7	48	24	6	41
24	32	19	6	49	39	27	42
25	35	32	8	50	42	47	44

Table 5-2: Characteristic rank by information theoretical based methods score

rank	**fscore**	**gini**	**rank**	**fscore**	**gini**
1	30	22	26	26	47
2	9	46	27	5	27
3	22	37	28	31	48
4	7	40	29	33	0
5	28	49	30	3	14
6	43	23	31	8	11
7	37	30	32	29	21
8	40	28	33	16	18
9	25	41	34	36	3
10	4	43	35	34	34
11	46	38	36	14	36
12	49	7	37	11	12
13	47	9	38	18	10
14	44	35	39	10	29
15	6	32	40	12	5
16	27	19	41	13	8
17	35	45	42	15	26
18	32	16	43	19	2
19	48	1	44	2	39
20	23	33	45	21	13
21	38	31	46	17	15
22	1	25	47	20	17
23	45	44	48	24	20
24	41	4	49	39	42
25	0	6	50	42	24

Table 5-3: Characteristic rank by statistical analysis based methods score

Rank	Coef	Rank	Coef	Rank	Coef	Rank	Coef	Rank	Coef
1	30	13	41	25	14	37	15	49	34
2	46	14	25	26	29	38	18	50	42
3	7	15	17	27	33	39	3		
4	9	16	16	28	45	40	44		
5	22	17	5	29	11	41	6		
6	20	18	37	30	32	42	24		
7	19	19	0	31	38	43	27		
8	28	20	21	32	31	44	10		
9	26	21	35	33	1	45	12		
10	4	22	8	34	47	46	48		
11	40	23	49	35	13	47	39		
12	**43**	**24**	**2**	**36**	**23**	**48**	**36**		

Table 5-4: Characteristic rank by one out reduction based on feature importance

5.3.2 Feature importance by wrapper methods

Different wrapper algorithms are designed and implemented. These algorithms attempt to systematically reduce or increase the number of features used, evaluating the resulting classifier performance on every step.

Feature selection by one out reduction based on feature importance

LDA classifier implements a feature importance or coefficient weight property, after fitting, the classifiers report the weight or cumulative weight of each of the coefficients used; in other words, how much does a given characteristic contribute to the decision made. To implement a wrapper method by one out reduction, the steps below are followed.

- Start with 50 features.
- Classifier is trained through 30 cycles.
- Scores are averaged.
- The least contributing feature (lowest feature importance or coefficient weight) is eliminated.
- Repeat until 2 features are left.

The results are plotted on Figure **5-2**. The best averaged FMI score per class produces a ranked list of coefficient performance (Table **5-4**).

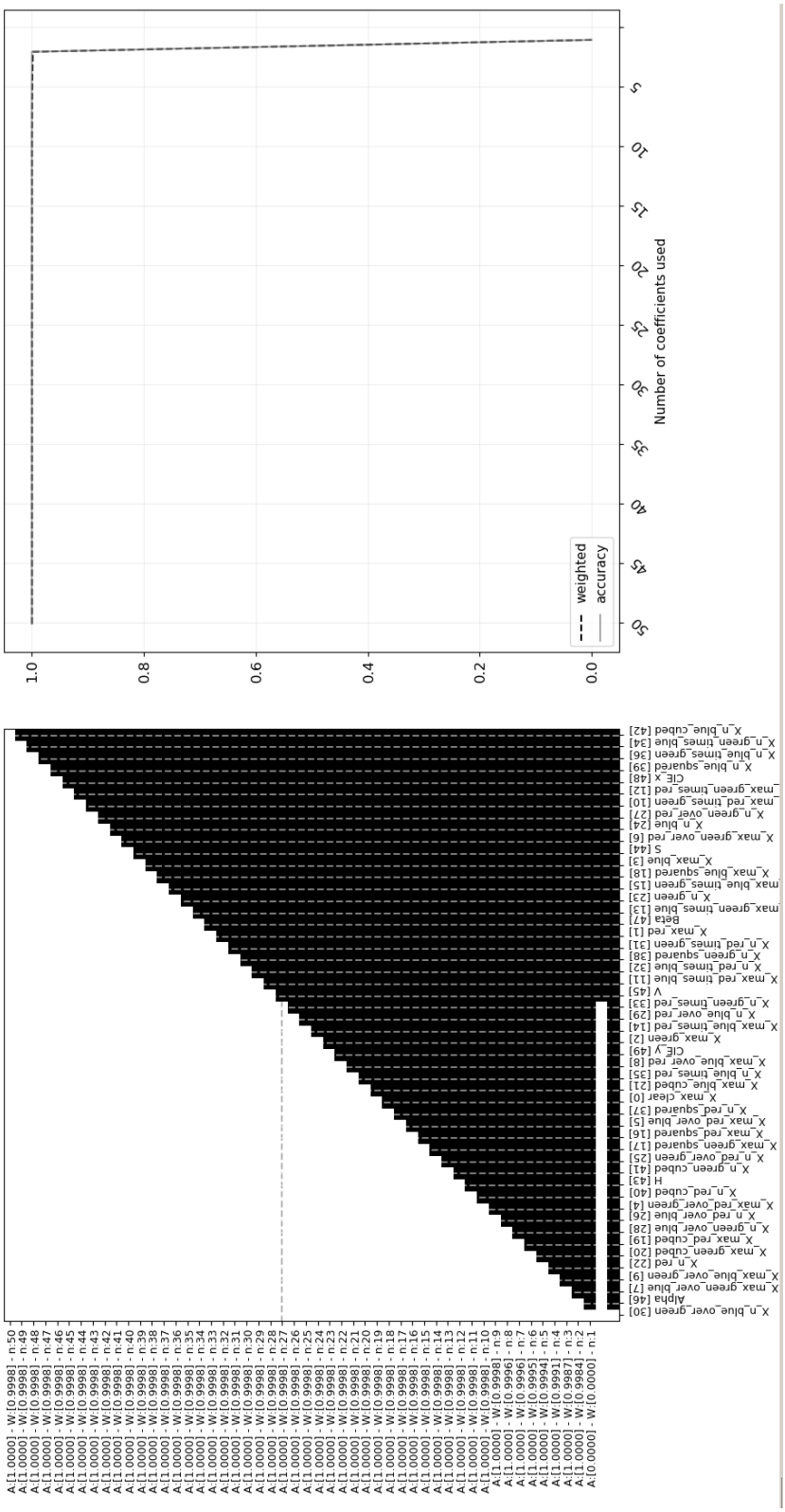

Figure 5-2: LDA performance through removal cycles based on feature importance.

Rank	Coef	Rank	Coef	Rank	Coef	Rank	Coef	Rank	Coef
1	9	13	5	25	13	37	6	49	45
2	3	14	41	26	19	38	21	50	46
3	29	15	23	27	12	39	31		
4	30	16	14	28	10	40	43		
5	20	17	48	29	0	41	42		
6	33	18	8	30	38	42	39		
7	16	19	11	31	28	43	24		
8	1	20	36	32	7	44	35		
9	17	21	34	33	49	45	40		
10	4	22	18	34	44	46	32		
11	26	23	2	35	47	47	37		
12	25	24	15	36	27	48	22		

Table 5-5: Characteristic rank by one out reduction based on classifier performance.

Feature selection by one out reduction based on classifier performance

An alternate route is to train the classifier through an intensive computing procedure where each of the features are eliminated; the configuration with best performance is preserved and the process is repeated.

- Start with 50 features
- Classifier is trained eliminating one feature at a time, until all 50 combinations of 49 features are tested
- The highest score set of remaining features is selected
- Repeat until 2 features are left.

The results are plotted on Figure **5-3**. The best averaged FMI score per class and average produces a ranked list of coefficient performance (Table **5-5**).

Feature selection by one addition based on classifier performance

It is also possible to run a constructive process where features are added instead of removed. This is also an intensive computing process where all possibilities are evaluated. The results are ploted on Figure **5-4**.

- Start with 0 features
- Classifier is trained adding one feature at a time, until all 50 combinations of 1 feature are tested

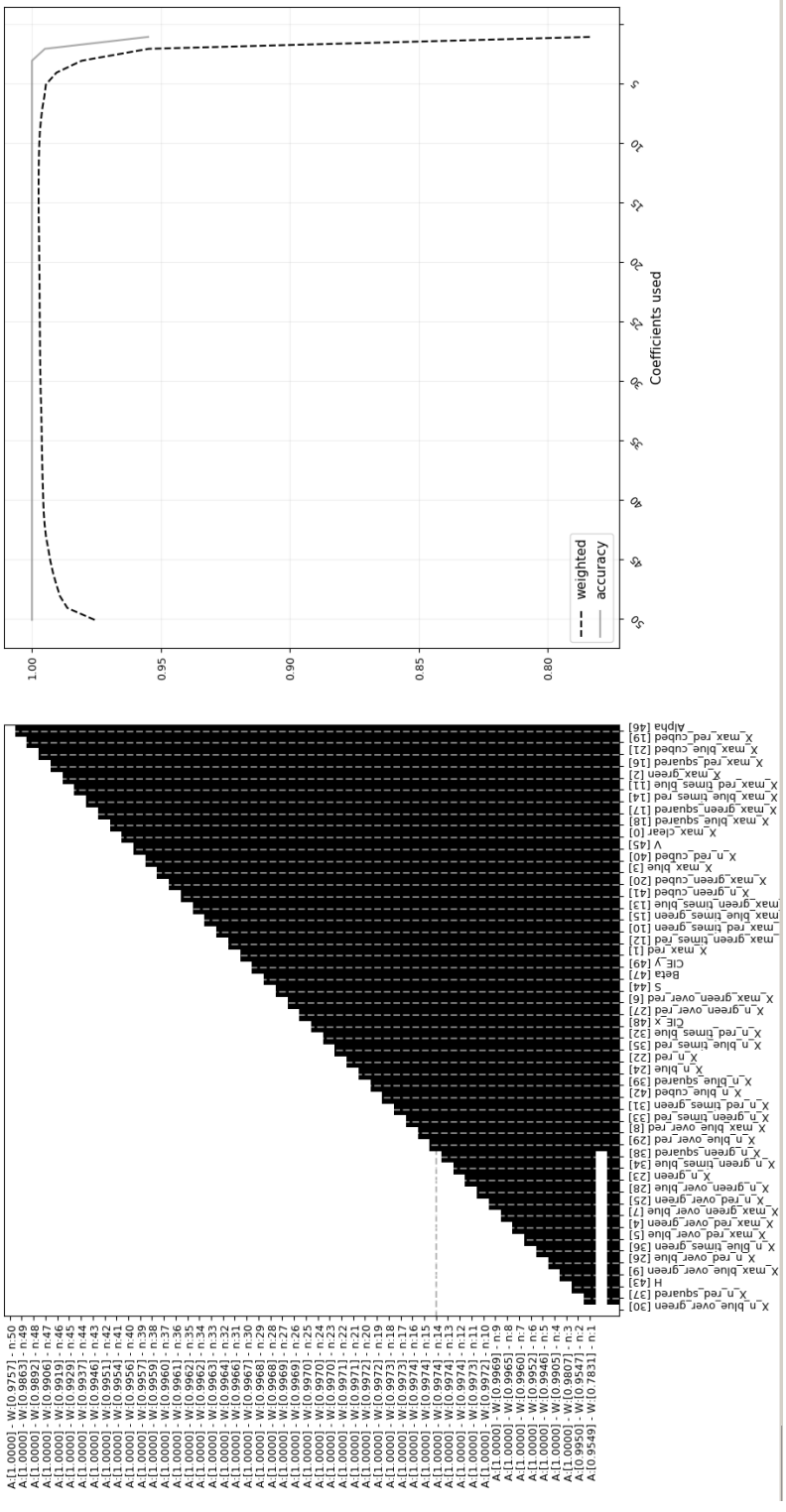

Figure 5-3: LDA performance through feature reduction cycles based on classifier performance.

rank	Coef	rank	Coef	rank	Coef	rank	Coef	rank	Coef
1	43	13	29	25	3	37	12	49	37
2	31	14	20	26	44	38	0	50	22
3	9	15	19	27	21	39	2		
4	8	16	1	28	47	40	27		
5	46	17	16	29	49	41	6		
6	30	18	17	30	18	42	45		
7	41	19	23	31	4	43	42		
8	33	20	11	32	13	44	39		
9	48	21	14	33	15	45	24		
10	7	22	5	34	34	46	40		
11	38	23	25	35	36	47	32		
12	28	24	26	36	10	48	35		

Table 5-6: Characteristic rank by addition based on classifier performance.

- The highest score set of remaining features is selected
- Repeat until 50 features are left.

The best averaged FMI score per class and average produces a ranked list of coefficient performance (Table **5-6**)

Feature selection by addition and reduction based on classifier performance

A dual approach is evaluated by adding and removing features on the same evaluation cycle. This can yield better results than only implementing addition or removal. The path of characteristics added and removed is volatile and thus difficult to plot. A ranked list is shown on Table **5-7**

- Start with 0 features.
- Classifier is trained adding one feature at a time, until all 50 combinations of 1 feature are tested.
- The highest score set of remaining features is selected.
- Classifier is trained removing one feature at a time, until all 50 combinations of 1 feature are tested.
- The highest score set of remaining features is selected.
- Repeat until no further improvement is possible.

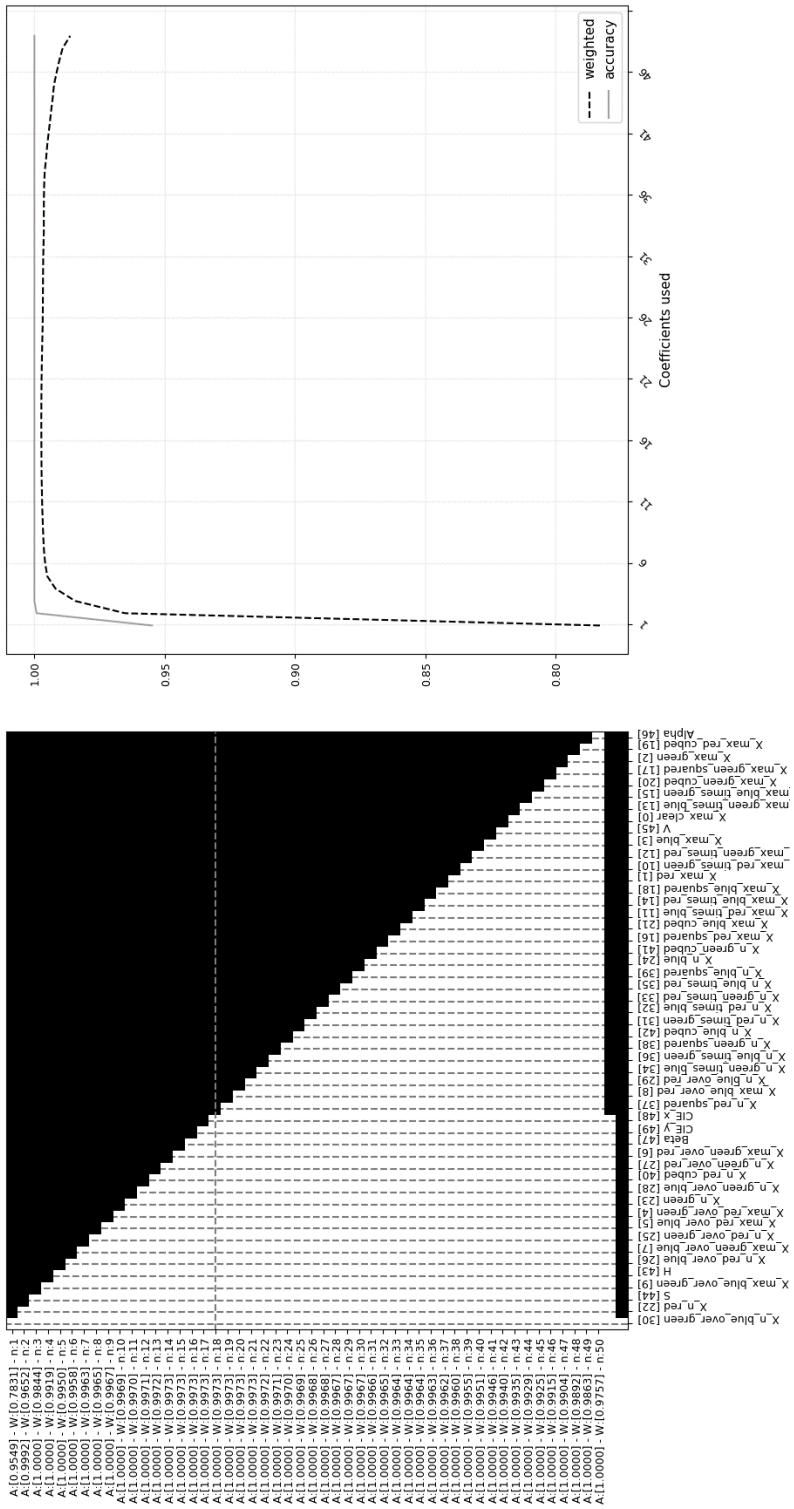

Figure 5-4: LDA performance through feature addition cycles based on classifier performance.

W score	Accuracy	Coefficients
0.46213171	0.6587414	0,
0.99795175	1.0000000	0, 30,
0.99870013	1.0000000	0, 30, 46,
0.99924284	1.0000000	0, 30, 46, 8,
0.99959991	1.0000000	30, 46, 8, 9,
0.99974261	1.0000000	30, 46, 8, 9, 37,
0.99978305	1.0000000	30, 46, 8, 9, 37, 4,
0.99980347	1.0000000	30, 46, 8, 9, 37, 4, 29,
0.99981928	1.0000000	30, 46, 8, 9, 37, 4, 29, 25,
0.99982509	1.0000000	30, 46, 8, 9, 37, 4, 29, 25, 48,
0.99982933	1.0000000	30, 46, 8, 9, 37, 4, 29, 25, 48, 26,
0.99983073	1.0000000	30, 46, 8, 9, 37, 4, 29, 25, 48, 26, 5

Table 5-7: Classifier progression through addition and reduction cycles, features used and performance.

Feature selection by combination of 2 features

An intensive computing brute force approach combination of two features at a time. This yields $50!/(2! * 48!) = 1225$ combinations. For every pair we run 30 cycles with different train and test samples.The colors of the matrix represent the average FMI. A perfect score (100% FMI accuracy) is marked with a black dot (Figure **5-5**). Table **5-8** allows performance comparison of different pairs of features by weighted score. At least 22 pairs of features can preserve a perfect classification score. A high correlation between coefficient 9 and coefficient 30 exists (as their formulas are almost identical), so in practice only half of the pairs (11) achieve perfect accuracy.

5.4 Feature selection results

5.4.1 Filter methods

All filter methods implemented can predict most of the features which yield the best results with larger number of samples. However, none of them predict the best performers when a minimum number of features is desired. The minimum number of features to preserve 100% accuracy is 3. A comparative plot is seen on Figure **5-6**

5.4.2 Wrapper methods

Some wrapper methods manage to achieve perfect score with only 2 features (Figure **5-7**). Minimum number of features that preserve 100% accuracy:

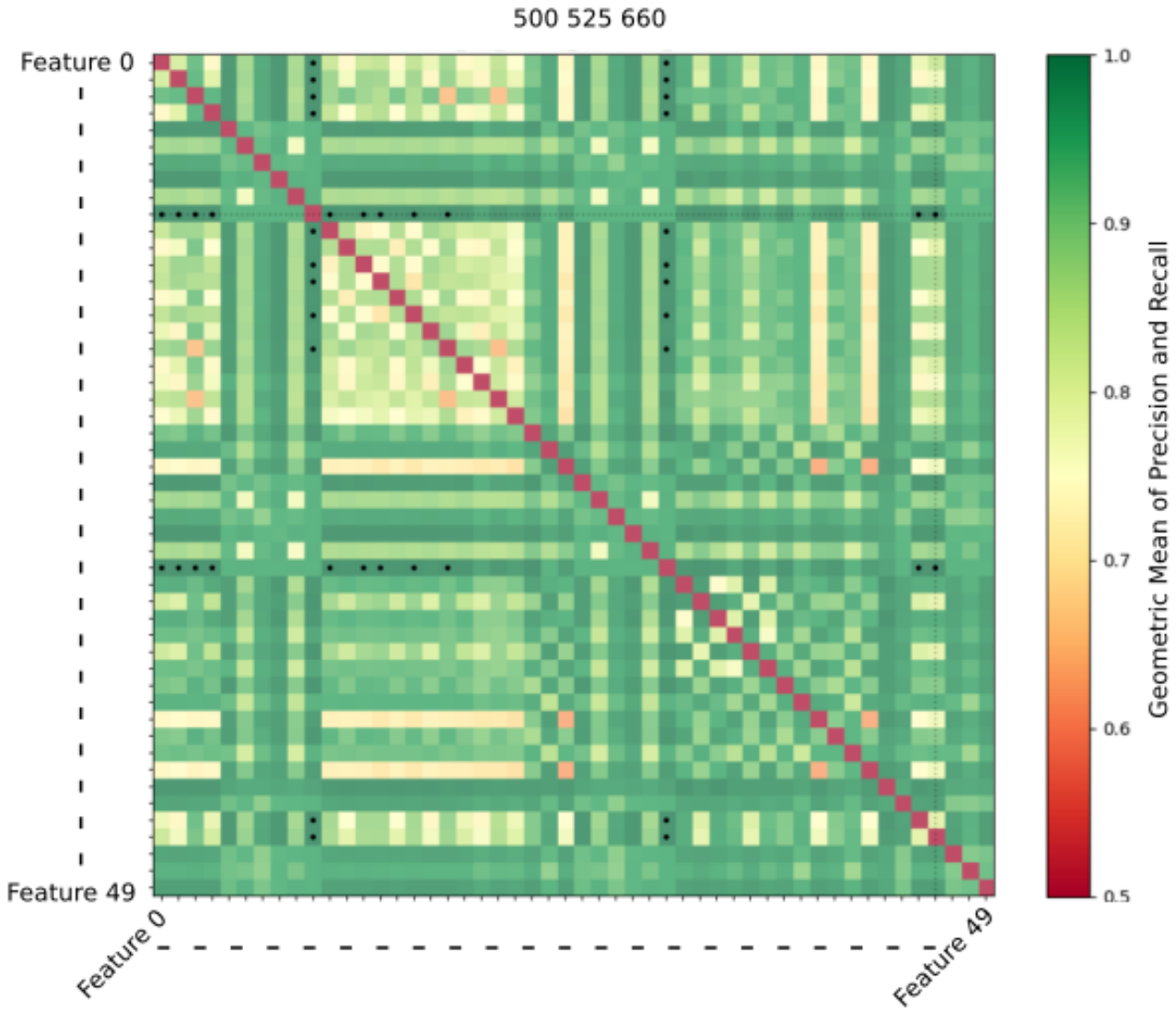

Figure 5-5: Classifier performance by intensive computing of pairs of characteristics. Black dots represent combinations which achieve 100% accuracy. The dotted cross

Feature 1	Feature 2	Accuracy	Feature 1	Feature 2	Accuracy
9	0	1.0000	30	0	1.0000
9	1	1.0000	30	1	1.0000
9	2	1.0000	30	2	1.0000
9	3	1.0000	30	3	1.0000
9	10	1.0000	30	10	1.0000
9	12	1.0000	30	12	1.0000
9	13	1.0000	30	13	1.0000
9	15	1.0000	30	15	1.0000
9	17	1.0000	30	17	1.0000
9	45	1.0000	30	45	1.0000
9	46	1.0000	30	46	1.0000

Table 5-8: Pairs of coefficients which achieve a perfect selection accuracy score.

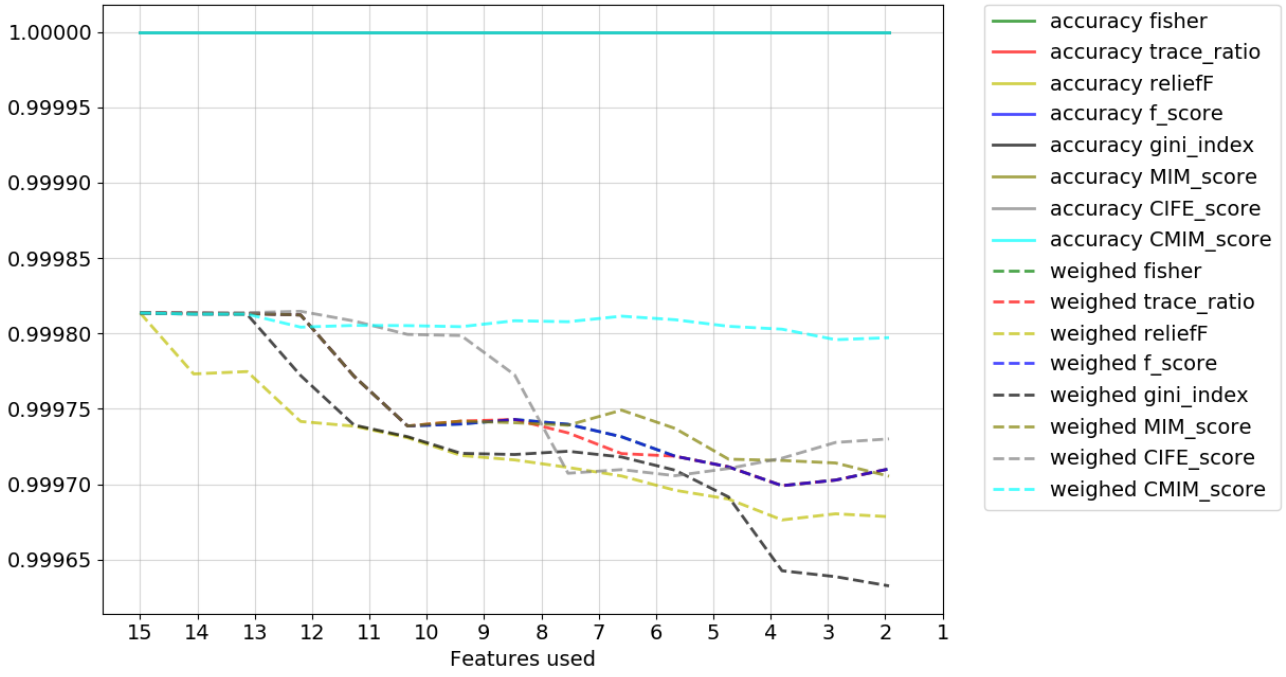

Figure 5-6: Filter methods performance progression as less features are used. Accuracy remains perfect for all methods.

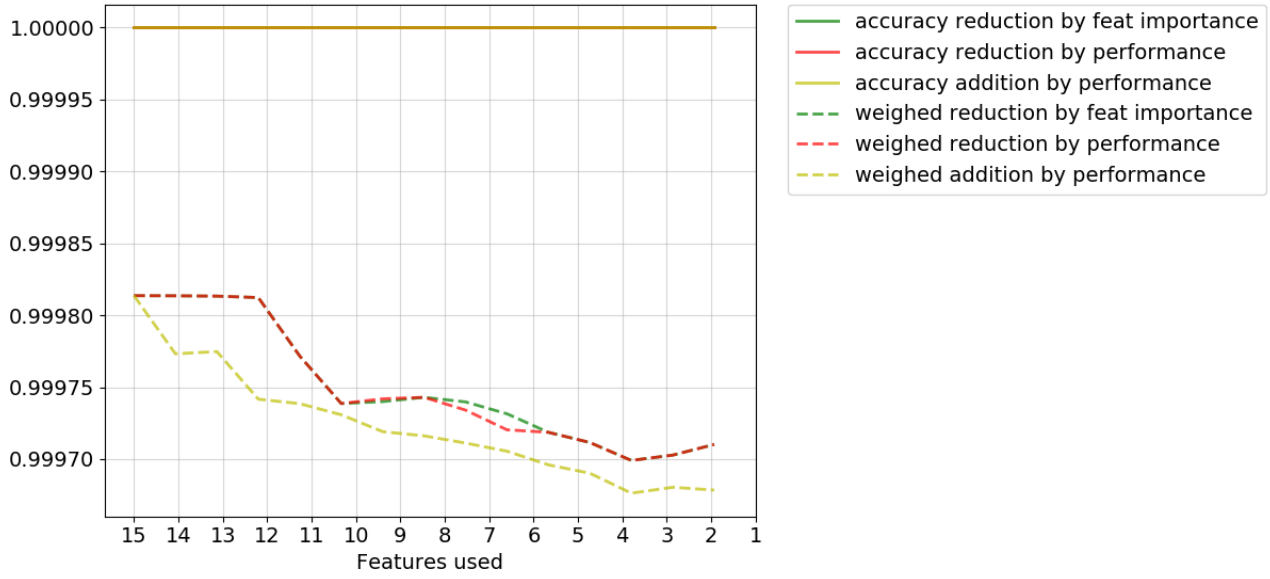

Figure 5-7: Wrapper methods performance progression as less features are used. Accuracy remains perfect for all methods.

- Reduction by Feature importance: 30, 46
- Reduction by Classifier performance: 9, 3
- Addition by Classifier performance: 43, 31, 9, 8

The addition and reduction algorithm allows to find a set of the most efficient 11 features, two of which allow perfect scores (Figure **5-8**).

5.4.3 Overall results

A brute force search finds all combinations of 2 which preserve 100% accuracy. We rank them by weighted score to realize that feature reduction and performance addition + reduction methods obtain what appear to be the highest weighted scores for 2 features (Table **5-9**). Two features suffice to obtain a perfect accuracy score, and reasonably high weighted score, with superior performance than all features.The best weighted score was found using a recursive addition and reduction algorithm evaluated by classifier performance 0.9998387, with 11 coefficients: 30, 46, 8, 9, 37, 4, 29, 25, 48, 26, 5. The original weighted score with 50 coefficients was 0.9980616, which shows that feature elimination improves the classifier performance. The best weighted score that can be achieved using only 2 features is 0.9983633, obtained using 9 and 46, and can be found through feature reduction or performance addition and reduction.

Features	**Accuracy**	**Weighted Score**	**Method(s)**
9,46	1.00000	0.9983633	feature reduction, performance addition + reduction
30,46	1.00000	0.9983633	feature reduction, performance addition + reduction
9,45	1.00000	0.99808698	Brute force only
30,45	1.00000	0.99808698	Brute force only
9,1	1.00000	0.99798986	Brute force only
30,1	1.00000	0.99798986	Brute force only
9,0	1.00000	0.99795175	Brute force only
30,0	1.00000	0.99795175	performance addition + reduction
9,3	1.00000	0.99764584	performance reduction
30,3	1.00000	0.99764584	Brute force only
9,2	1.00000	0.99703991	Brute force only
30,2	1.00000	0.99703991	Brute force only
9,10	1.00000	0.99606363	Brute force only
9,12	1.00000	0.99606363	Brute force only
30,10	1.00000	0.99606363	Brute force only
30,12	1.00000	0.99606363	Brute force only
9,17	1.00000	0.99601202	Brute force only
30,17	1.00000	0.99601202	Brute force only
9,13	1.00000	0.99597521	Brute force only
9,15	1.00000	0.99597521	Brute force only
30,13	1.00000	0.99597521	Brute force only
30,15	1.00000	0.99597521	Brute force only

Table 5-9: Best performing pairs of coefficients and methods used to discover them

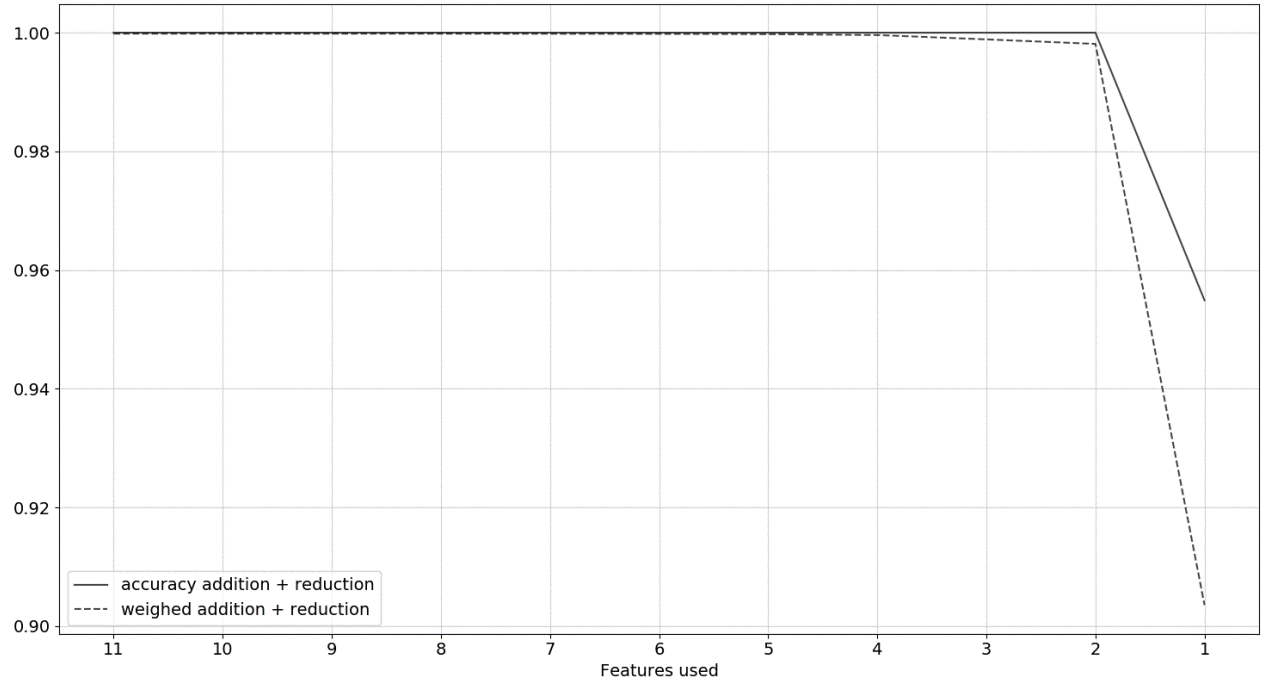

Figure 5-8: Wrapper methods by addition and reduction performance progression as less features are used.

The feature selection process allowed us to find the least number of features that could preserve perfect accuracy without significantly jeopardizing classifier certainty. It also exposed the maximum expected performance, and demonstrated that several combinations perform better than the original set of 50 characteristics. For the on-line device, two configurations will be further tested:

- Highest weighted FMI: 11 features: 30, 46, 8, 9, 37, 4, 29, 25, 48, 26, 5.
- Fastest calculation: 2 features: 9,46.

6 Conclusions

The main objective of the research has been met through the design and prototyping of a high speed sorting system using low cost TCS3200 and AP5130 discrete color sensors and 3 different types of LEDs driven by a STM32F429I. The estimated total cost of the electronic assembly was kept under US$100 in 1000 production volumes, with accuracy superior to those obtained using CMOS sensors and higher cost CPU or FPGA computing platforms. The system implements a cost effective reflective hemisphere and cone construction which allows high light uniformity, lowering errors due to hotspots, avoiding the use of focusing lenses.

The core classification algorithms and highest performance system set-up were obtained and tuned through a PC based training process where fruit data was captured using different light source combinations. After filtering and data augmentation a feature extraction process generates 50 different features. Further study into these characteristics by means of feature selection techniques shows that only a subset of these 50 features are necessary to preserve the accuracy of the system.

The classifiers were implemented on the MCU platform by means of finite state machines and peripheral managers in C using Atollic Truestudio and STM32CubeIDE on top of a hardware abstraction layer of ST Microelectronics. The portable on-line system obtains an accuracy of over 99% using LDA during early laboratory testing, at an average rate of 10 fruits per second (72Kg per hour). The high discriminant capacity system is possible thanks to the coupling of the discrete low-cost sensor and the narrowband LEDs in wavelengths which exploit the areas of maximum sensitivity and lower inter-channel overlap for the specific colors of interest, which are superior to the 94% accuracy obtained with white light. In many cases a couple of LEDs could suffice to obtain separable datasets. A systematic method of LED selection has been validated.

The best performance was found using 3 discrete LEDs in 500, 525 and 660 nm wavelengths and a One vs One Ensemble LDA classifier. The resulting implementation complexity is substantially lower than that of typically higher performance algorithms such as RF and SVC, showing that classifier performance is sensor and LED dependent, and general assumptions should not be madeThis is to say, feature selection processes should only take place after classifier training has taken place.

The main products of the executed work were a functional prototype capable of real time sorting, a patent grant which is in process of preliminary revision and an article which will be submitted by the time this document is published.

6.1 Future work

The sensor array used implements four discrete TCS3200 devices; while this allows for parallel data capture it introduces positional noise, as not all sensors are equally aligned and at the same distance from the fruit. Further research should be conducted on alternate sensors which solve this shortcoming.

The system can easily distinguish different maturation stages through reflectance spectroscopy; other work has shown it is possible to detect other characteristics and defects in fruits.

Thanks to the features selection process, the decision-making stage can take place with a limited number of simple mathematical operations, which opens the door to implementing the existing code in even more modest MCUs, further simplifying the system and lowering the total unit cost.

The recent trend of low cost and high processing power computing platforms which can run full featured operating systems such as embedded distributions of Linux make it possible to consider a similarly low costed device which implements CCD or CMOS modules, opening the door for more advanced machine vision techniques without compromising on simplicity and economy.

Bibliography

[1] Acosta, Luis J.: Colombia to examine selling coffee at its own price, ignoring New York market. In: *Reuters* (2019)

[2] Akmalia, Dina ; Saputro, Adhi H. ; Handayani, Windri: A Non-Destruction Measurement System based on Hyperspectral Imaging for Sugar Content in Banana. In: *IEEEE* (2017)

[3] Arcila, J.: *Sistemas de Producción - Densidad de siembra y productividad de los cafetales*. Cenicafe, 2011

[4] Betancur, J. A.: Segmentación de frutos de cafe mediante métodos de crecimiento de regiones. In: *Rev.Fac.Nal.Agr.Medellin.Vol.59, No.1* (2006), S. 3311–3333

[5] Comercio, Industria y T. d.: *Informacion: Perfiles EconÃ³micosDepartamentales − MinCIT − Caldas.*. Version: 2019

Dahl, Russ: *Light-Emitting Diodes: A Primer*. `https://www.photonics.com/Article.aspx?AID=36706`. Version: 2019

E. Alvarez, F. Alvarez M. T.: *Propiedades físico-mecanicas del fruto y del sistema fruto-pedunculo del café variedad Colombia*. 1999

Farfan, F.: *Sistemas de Producción - Cafés Especiales*. Cenicafe, 2011

Feiping Nie, Yangqing Jia Changshui Z. Shiming Xiang X. Shiming Xiang ; Yan, Shuicheng: Trace ratio criterion for feature selection. In: *AAAI, Volume 2* (2008), S. 671–676

Fen, Dai ; Tiansheng, Hong ; Kun, Zhang ; Ya, Hong: Nondestructive detection of pesticide residue on longan surface. In: *IEEE* (2010)

Fleuret, FranÂ¸cois: Fast binary feature selection with conditional mutual information. In: *The Journal of Machine Learning Research* (2004), S. 5:1531–1555

FNC: *La Gente del Cafe - Estadisticas Caficultores*. `http://www.cafedecolombia.com/particulares/es/la_tierra_del_cafe/la_gente_del_cafe/`. Version: 2010

FNC: *Estadísticas Cafeteras*. `https://federaciondecafeteros.org/wp/estadisticas-cafeteras/`. Version: 2020

FNC: *Precio del Café en Colombia, visited February 24th 2020*. `https://federaciondecafeteros.org/static/files/precio_cafe.pdf`. Version: 2020

G. I. Puerta, Q.: *Influencia de los granos de café cosechados verdes, en la calidad física y organoléptica de la bebida*. 2000

Gini, CW: Variability and mutability, contribution to the study of statistical distribution and relations. In: *Studi Economico–Giuricici della R* (1912)

Jeremy Hodges, Aine Q. Fabiana Batista B. Fabiana Batista: Climate Change Threatens to Make Your Morning Brew More Expensive. In: *Bloomberg News* (2019)

J.J. Carvajal, H.: Colorimetría del fruto de café (Coffea arabica L.) durante su desarrollo y maduración. In: *Rev.Fac.Nal.Agr.Medellin* (2011), S. 6229–6240

Jundong Li, Suhang Wang Fred M. Kewei Cheng C. Kewei Cheng: Feature Selection: A Data Perspective. In: *ACM Computing Surveys* (2016)

Lewis, David D.: Feature selection and feature extraction for text categorization. In: *Proceedings of the workshop on Speech and Natural Language* (1992), S. 212–217

Li, Jing ; Xue, Long ; He, Xiuwen ; Liu, Muhua: Visible and Near infrared reflectance spectroscopy for determining soluble solids content of navel orange. In: *IEEE* (2011)

Lin, Dahua ; Tang, Xiaoou: Conditional infomax learning: an integrated framework for feature extraction and fusion. In: *Computer Vision ECCV* (2006), S. 68–82

Marin-Lopez, S.: Cambios físicos y químicos durante la maduración del fruto de café (Coffea arabica L. var Colombia). In: *Cenicafe* (2003), S. 208–225

Minghua Zhang, Eike L. Adam Hale H. Adam Hale: Feasibility of using remote sensing techniques to detect spider mite damage in stone fruit orchards. In: *IEEE* (2008)

Montes, N.: La visión artificial aplicada al proceso de producción de café. Universidad Nacional de Colombia, 2001. – Forschungsbericht

Montes, N.: Real-time classification of coffee fruits using FPGA. Universidad Nacional de Colombia, 2015. – Forschungsbericht

MÃnguez, I.: *Disappearance of chlorophylls and carotenoids during the ripening of the olive*. 1995

Nila ; Saputro ; Imawan: The prediction system of bruising depth of guava based on VIS-NIR imaging. In: *IEEE* (2017)

Pedregosa, F. ; Varoquaux, G. ; Gramfort, A. ; Michel, V. ; Thirion, B. ; Grisel, O. ; Blondel, M. ; Prettenhofer, P. ; Weiss, R. ; Dubourg, V. ; Vanderplas, J. ; Passos, A. ; Cournapeau, D. ; Brucher, M. ; Perrot, M. ; Duchesnay, E.: Scikit-learn: Machine Learning in Python. In: *Journal of Machine Learning Research* 12 (2011), S. 2825–2830

Poltak Sihombing, Sajadin Sembiring Nogar S. Faddly Tommy T. Faddly Tommy: The Citrus Fruit Sorting Device Automatically Based On Color Method By Using Tcs320 Color

Sensor And Arduino Uno Microcontroller. In: *The 3rd International Conference on Computing and Applied Informatics* (2018)

Putra, Rizaldi T. F.: Application of Color Sensor in the Determination of Tomato Fruit Ripeness (Solanum Lycopersicum, L) in Gravitation Type Fruit Sorting Tool. In: *Indonesian Journal of Agricultural Research* (2019)

Ramos, P. J.: Identificación y clasificación de frutos de café en tiempo real, a traves de la medicion de color. In: *Cenicafe* (2010), S. 315–326

Rangkuti ; Saputro ; Imawan: Prediction of soluble solid contents mapping on averrhoa carambola using hyperspectral Imaging. In: *IEEE* (2017)

Richard O Duda, Peter E H. ; Stork, David G.: *Pattern classification*. John Wiley & Sons, 2012

Sandoval, Z.: Caracterización de café cereza empleando técnicas de visión artificial. In: *Rev.Fac.Nal.Agr.Medellin* (2007), S. 4105–4127

SCAP: *Best of Panama Auction 2019*. `https://auction.bestofpanama.org/en/lots/auction/best-of-panama-eauction-2019?tab=lots`. Version: 2019

Schiffman, Richard: As Climate Changes, Colombiaâ^{TM}s Small Coffee Farmers Pay the Price. In: *Yale Environment 360* (2019)

Scholkopft, Bernhard ; Mullert, Klaus-Robert: Fisher discriminant analysis with kernels. In: *Neural networks for signal processing IX* (1999), S. 267–288

Sikonja, Marko R. ; Kononenko, Igor: Theoretical and empirical analysis of relieff and rrelieff. In: *Machine learning* (2003), S. 53(1–2):23–69

Sitti Wetenriajeng Sidehabi, Intan Sari Areni Ingrid N. Ansar Suyuti S. Ansar Suyuti: The Development of Machine Vision System for Sorting Passion Fruit using MultiClass Support Vector Machine. In: *Journal of Engineering Science and Technology Review* (2018)

Sun, Jason ; KÃ$\frac{1}{4}$nnemeyer, Rainer ; McGlone, Andrew ; Tomer, Nathan: Fruit orientation in NIR transmission for vascular. In: *Crown* (2017)

Tamayo, M.A.: *Análisis de la capacidad discriminante de características de color en imágenes multiespectrales de frutos de café*, Doctorado en Ingeniería - Automática, Diss., 2018

Tetsuya Inagaki, Yoshiaki Shimomura Satoru T. Daisuke Nozawa N. Daisuke Nozawa: Three-Fibre-Based Diffuse Reflectance Spectroscopy for Estimation of Total Solid Content in Natural Rubber Latex. In: *Journal of Near Infrared Spectroscopy* (2016)

Tibshirani, Robert: Regression shrinkage and selection via the lasso. In: *Journal of the RoyalStatistical Society* (1996)

Wang, Xiao ; Xue, Long ; He, Xiuwen ; Liu, Muhua: Vitamin C Content Estimation of Chilies Using Vis/NIR spectroscopy. In: *IEEE* (2011)

Wright, Sewall: The interpretation of population structure by fwith special regard to systems of mating. In: *Evolution* (1965), S. 395–420

Zuliang Wang, Chuanglue Cao Ting Z. Qi An A. Qi An: Design and Implementation of Automatic Sorting Control System for Melon and Fruit Products. In: *Application of Intelligent Systems in Multi-modal Information Analytics* (2019)