Detection of Lint by Using Machine Vision

HAMK
HÄMEEN AMMATTIKORKEAKOULU
HÄME UNIVERSITY OF APPLIED SCIENCES

Thesis presented for the degree: Master of Science in Energy Engineering Technology (Automation)

Häme University of Applied Sciences: Degree Programme in Electrical and Automation Engineering

2016-2017 Spring Semester

Benjamin Filtjens

Preface

My university of choice for Erasmus was Häme University of Applied Sciences (HAMK) in Finland. One of the available research subjects was to research the possibility of detecting lint by using machine vision. My choice went to this subject because it is very research oriented, ideal for a Master's student. It is also beneficial to increase my knowledge in machine vision since it is a topic gaining increased importance for automation students. Also, getting programming experience in MATLAB is also very beneficial.

While four months is a very limited time to complete a research topic it was a very educational experience and my expectations were met. It was interesting to apply my theoretical background and gain increased knowledge about machine vision. My previous experience of working with MATLAB was very limited so my knowledge about MATLAB increased massively.

Firstly, I would like to thank my supervisors Raine Lehto and ing. Geert Leen for their help throughout the thesis. Secondly, I would like to thank Jeroen Lievens for correcting the various assignments I had to make for my home university. I would also like to thank the University of Leuven and HAMK for the opportunity of going on Erasmus and letting me do my thesis in such an international environment. Lastly, I would like to thank all the lecturers at HAMK who were always very friendly and willing to help.

HAMK
HÄMEEN AMMATTIKORKEAKOULU
HÄME UNIVERSITY OF APPLIED SCIENCES

ABSTRACT

Degree programme in Electrical and Automation Engineering
Valkeakoski

Author	Benjamin Filtjens	**Year** 2017
Subject	Detection of Lint by Using Machine Vision	
Supervisor(s)	Raine Lehto and Geert Leen	

ABSTRACT

This thesis, commissioned by Häme University of Applied Sciences, researches the possibility of detecting lint by using machine vision. Due to the small particle size and high movement speed of the lint, various issues occur. Firstly, to detect the small lint particles a sufficient resolution is required. Secondly, since the lint has a high movement speed a high framerate is required to fully represent all the lint passing by. Lastly, a short exposure time is required to prevent inaccuracy due to motion blur. The goals of this thesis are to research the most optimal machine vision components, if the hardware currently available can detect the small particles with a sufficient framerate and a method to prevent motion blur.

The most optimal components were found by performing a literature study. Calculations were made to test if the currently available hardware can fulfil the goals. A colleague created a short duration strobe light to prevent motion blur. Lastly, a practical test setup and MATLAB program were created to verify the theoretical conclusions and detect the lint.

The strobe light uses four high power white LEDs with a flash duration of one microsecond. The calculations have concluded that the currently available hardware is capable of fully representing the lint passing by at a minimum particle size of 45 microns. Analyses of the MATLAB program verified that the theoretical calculations were correct.

Keywords MATLAB image processing lint frame rate

Pages 75 pages including appendices 8 pages

HAMK
HÄMEEN AMMATTIKORKEAKOULU
HÄME UNIVERSITY OF APPLIED SCIENCES

Degree programme in Electrical and Automation Engineering
Valkeakoski

Author	Benjamin Filtjens	**Year** 2017
Subject	Detection of Lint by Using Machine Vision	
Supervisor(s)	Raine Lehto and Geert Leen	

ABSTRACT

Deze thesis, uitgevaardigd door Häme University of Applied Sciences, onderzoekt de mogelijkheid tot het detecteren van stofdeeltjes door gebruik te maken van machine visie. Door de kleine dimensies en hoge voortbewegingssnelheden van de deeltjes treden er allerlei problemen op. Zo is een hoge resolutie noodzakelijk om de deeltjes te detecteren. Ook moet de framerate van de camera voldoende snel zijn om alle deeltjes die voorbij bewegen te detecteren. Ten laatste, is een korte sluitertijd van de sensor noodzakelijk om motion blur te voorkomen. Het doel van deze thesis is om de meest optimale machine visie onderdelen te onderzoeken, de mogelijkheid om het lint te detecteren met de hardware die beschikbaar is te onderzoeken en om een oplossing te zoeken voor motion blur.

De meest optimale machine visie setup werd gevonden met een literatuurstudie. Berekeningen zijn gemaakt om de beschikbare hardware te testen. Een flits van zeer korte duur is door een collega student gemaakt om motion blur te voorkomen. Ten laatste, is er een praktische opstelling en een MATLAB-programma gemaakt om de theoretische conclusies te verifiëren en het stof te detecteren.

De flits gebruikt vier hoogvermogen witte leds met een flitsduur van één microseconde. De berekeningen toonden aan dat de beschikbare hardware in staat is om alle deeltjes te filmen met een minimum grote van 45 micrometer. Het Matlab programma verifieerde dat de theoretische berekeningen correct waren.

Keywords MATLAB image processing lint frame rate

Pages 75 pages including appendices 8 pages

CONTENTS

1 INTRODUCTION 9

1.1 Theoretical Framework 9
1.2 Problem Analysis 9
1.3 Goals 10
1.4 Materials and Methods 11

2 LITERATURE RESEARCH 12

2.1 Industrial Background: Linting 12
2.2 Sensor 14
2.2.1 Charge Coupled Device (CCD) 14
2.2.2 Complementary Metal Oxide Semiconductor (CMOS) 15
2.2.3 Line Scan Sensor 15
2.2.4 Conclusion 16
2.3 Exposure Control 17
2.3.1 Global Shutter 17
2.3.2 Rolling Shutter 18
2.3.3 Best of Both Worlds 19
2.3.4 Conclusion 20
2.4 Lens 21
2.5 Considerations when Selecting a Camera and Lens 22
2.5.1 Spatial Resolution, Field of View and Frame Rate 22
2.5.2 Pixel and Sensor Format 23
2.5.3 Focal Length 24
2.6 Available Hardware at HAMK 24
2.7 Light 25
2.7.1 Features of Interest 26
2.7.2 Light Types 26
2.7.3 Quantum Efficiency and Spectral Output 27
2.7.4 Illumination Techniques 28
2.7.5 Micro Second Strobe Lighting 29
2.8 Programming Environment 29
2.9 Contrast Improvement 30
2.10 Filter 33
2.10.1 Edge Detection 33
2.11 Image Segmentation 34
2.11.1 Frame Differencing 34
2.11.2 Thresholding 34
2.11.3 Gaussian Mixture Models 35

3 VISION SETUP DESIGN: CALCULATIONS 36

3.1 Calculations setup 1 36
3.1.1 Field of View 36
3.1.2 Working distance 37
3.1.3 Depth of Field 38
3.1.4 Magnification 40
3.1.5 Images per Second 40

3.1.6 Flash Exposure Time ... 41
3.1.7 Sensor Readout Time ... 42
3.1.8 Summary ... 43
3.2 Calculations setup two ... 43
3.3 Conclusion ... 44

4 VISION SETUP OVERVIEW ... 45

4.1 Strobe Light ... 46
4.1.1 Strobe and Camera Exposure Synchronization ... 46
4.1.2 Monostable Multivibrator ... 48
4.1.3 MOSFET and LED driver ... 49
4.1.4 LED ... 49
4.2 Program Overview ... 50

5 TEST SETUP RESULTS ... 53

5.1 Flash Synchronization ... 53
5.2 Pre-recorded Performance ... 54
5.2.1 Gaussian Mixture Models ... 54
5.2.2 Histogram Binary Imaging ... 57
5.2.3 Frame Differencing ... 59
5.3 Image Analysis ... 60
5.3.1 Smallest Detected Feature ... 61
5.4 Frames per Second ... 62

6 CONCLUSION ... 63

REFERENCES ... 64

Appendices

Appendix 1 Recording Settings
Appendix 2 Gaussian Mixture Models
Appendix 3 Automatic Thresholding by Using Otsu’s Method
Appendix 4 Frame Differencing
Appendix 5 Image Analysis

LIST OF FIGURES

Figure 1. Number of lint particles per m^2 particle area (Lestiani et al., 2013) 12
Figure 2. Measured number of lint particles per m² vs. number of printed copies (Lestiani et al., 2013) 13
Figure 3. CCD sensor (Qimaging, n.d.)... 14
Figure 4. CMOS sensor (Qimaging, n.d.) 15
Figure 5. Line scan sensor principle (DALSA, n.d.) 15
Figure 6. Line scan and conveyor principle (DALSA, n.d.) 16
Figure 7. Global shutter (Basler, 2017) 17
Figure 8. Rolling mode (Basler, 2017) 18
Figure 9. Rolling shutter flash window (Basler, 2017) 19
Figure 10. Direct-coupled general purpose output line (Basler, 2017).................... 19
Figure 11. Conventional lens angular field of view versus telecentric lens zero angle field of view (Edmund Optics, n.d.-a) 21
Figure 12. The telecentric lens has no depth perception error in comparison to a conventional lens (Edmund Optics, n.d.-a) 21
Figure 13. Circular Object Image Reproduction (Andor, n.d.)................................. 22
Figure 14. Comparison of lighting types (Daryl, 2017).. 26
Figure 15. Comparison of spectral output from various light sources (Daryl, 2017) 27
Figure 16. Quantum efficiency (Basler, n.d.)... 27
Figure 17. Lens transmittance (Basler, 2016)... 28
Figure 18. Back lighting (Daryl, 2017).. 28
Figure 19. Monadic operation: output (Corke, 2011) 30
Figure 20. Example one (Corke, 2011) 31
Figure 21. Example two (Corke, 2011) 31
Figure 22. Contrast improvement 32
Figure 23. Spatial operation (Corke, 2011)... 33
Figure 24. Gray-level density functions of two regions in an image. Left image: clear valley present for easy threshold selection, right image: non-optimal histogram for threshold selection (Huang & Chau, 2008) 35
Figure 25. Depth of Field. A point in the plane Q is focused to a point in plane T. The object can move between region PR, the region between the closest focus point S1 and furthest S2, while still maintaining an optimal focus. (Batchelor, 2012)...................... 38
Figure 26. Relationship between different lens and sensor parameters (National Instruments, 2014) 38
Figure 27. Magnification required when the sensor size and field of view are not similar (Blankinship, 2005) 40
Figure 28. Detailed relationship between lens and sensor parameters (Edmund Optics, n.d.-b) 41
Figure 29. Vision setup overview 44
Figure 30. Vision Setup 45
Figure 31. Strobe Light Diagram... 46
Figure 32. General-purpose output line settings 46
Figure 33. General Purpose Output at 180 Frames per Second 47
Figure 34. Monostable multivibrator circuit (De Brabanter, 2017) 48
Figure 35. 100ηs Pulse Width (De Brabanter, 2017)... 48
Figure 36. MOSFET and LED driver circuit (De Brabanter, 2017) 49
Figure 37. High-power LEDs... 49

Figure 38. Video processing diagram 50
Figure 39. Extensive image analysis diagram 51
Figure 40. Alternative light source 52
Figure 41. Camera and strobe light synchronization (De Brabanter, 2017) 53
Figure 42. 250ηs output pulse (De Brabanter, 2017).. 53
Figure 43. Gaussian mixture models result 54
Figure 44. Gaussian mixture models with an additional noise filter......................... 55
Figure 45. Gaussian mixture models performance analysis..................................... 56
Figure 46. Noise filter performance analysis... 56
Figure 47. Segmentation using Otsu’s thresholding result 57
Figure 48. Poor contrast 57
Figure 49. Segmentation using Otsu’s thresholding performance analysis............. 58
Figure 50. Image Subtraction Results.. 59
Figure 51. Image Subtraction Performance Analysis 59
Figure 52. Image analysis process 60
Figure 53. Smallest Detected Particle.. 61
Figure 54. Calculating Pixel Size... 61
Figure 55. Framerate 62

LIST OF TABLES

Table 1. Basler acA 1920-25um specifications 24
Table 2. Basler C125-0618-5M specifications 24
Table 3. Basler acA 1300-200um specifications 25
Table 4. Basler C125-0818-5M specifications 25
Table 5. Vision setup one, summary 43
Table 6. Vision setup two, summary 43

1 INTRODUCTION

1.1 Theoretical Framework

This thesis was commissioned by Häme University of Applied Sciences (HAMK) located in Valkeakoski, Finland. One of the subjects researched at HAMK involves the measurement of lint that is formed during high speed offset printing and paper manufacturing. Lint is a fibre formed when loosely bonded particles detach from the surface of paper. These fibres accumulate on the offset printing blankets and disturb the transfer from the inking system onto the blanket. Removing the lint from the printing blankets also results in a decrease of production efficiency. (Lestiani, Batchelor, & Banham, 2014)

The increase of printing press speeds and use of recycled paper increases linting problems for the paper manufacturers. To attenuate problems connected to linting, paper manufacturers must be able to monitor the linting level during paper production and adjust this level according to the tolerances of each printer by applying a suitable control strategy (Brouillette, Morneau, Chabot, & Daneault, 2006). Previous research by Amiri, Bégin, Deshaies, & Mozaffari (2004) also concluded that increased pulp quality reduces pulp linting propensity. A real-time lint measurement system can thus help test the quality of the paper during high speed offset printing and help paper manufacturers monitor the linting during paper production. The subject of this thesis is to design a machine vision setup to form an objective measurement of the amount of lint that is formed during high speed offset printing and paper manufacturing.

1.2 Problem Analysis

Previous research has concluded that a single lint measurement taken after a printing trial run does not give an accurate representation of the amount of lint that is formed due to the high variance of the amount of lint across different trial runs (Lestiani et al., 2014). To get an accurate representation, it is therefore necessary to implement a setup that can continuously monitor the amount of lint.

Small lint particles up to ten microns can already be detected by using laser diffraction, which has a volume flow rate of 27 litres per minute. The larger particles, which also pose the most problems during offset printing, cannot be detected yet. Therefore, this thesis focuses on detecting the larger particles with sizes ranging from ten microns to a few millimetres (Nguyen, 2015).

These specifications pose some difficulties in implementing an accurate vision setup. Firstly, there is the effect of motion blur which occurs when the object moves relatively quickly compared to the shutter speed of the camera, so that the image moves while the shutter is open (Wloka & Zeleznik, 1996). The motion blur results in the camera perceiving the lint bigger than it is, which reduces the accuracy of the system. Secondly, continuously monitoring objects of such high speed requires processing many images every second and thus high bandwidth. Lastly, the small size of the lint requires a very high resolution. The high resolution and fast processing speed are two requirements that are in conflict since a higher resolution will increase the sensor readout time (Basler, 2017).

1.3 **Goals**

An optimal machine vision setup must be determined. This overall goal presupposes the following research questions:

- Which type of camera, sensor and lens would be most ideal?
- Which type of lighting has the most optimal spectrum?
- How should the lighting be placed to get the most accurate result?
- Which machine vision software is most ideal?

The number of images that must be taken depends on the speed of the lint and the field of view. With a larger field of view a higher resolution is required to detect the lint. Since the field of view is limited and the lint moves at high speeds, a high number of images need to be taken every second. It should be researched if the camera and lens available at HAMK can capture enough images at a sufficient resolution to detect particles of ten microns. Also, a solution must be found to decrease the exposure time of the sensor and thus prevent motion blur.

The three aforementioned goals must be verified by creating a program and practical test setup.

1.4 Materials and Methods

An extensive literature study took place to learn the current knowledge that is available about machine vision. After this study, it was possible to find the most optimal machine vision elements and calculate if the available hardware is capable to fulfil the goals.

To tackle the problem of motion blur, there are two viable solutions. If budget is not a constraint, then a new camera could be purchased that is capable of high speed imaging. If budget is a constraint, a high-speed strobe light created by a colleague that is much faster than the minimum exposure of the camera could be an option. The high-speed strobe light will reduce the time that the lint is visible to the sensor despite the slow exposure of a regular industrial camera. However, there may be a problem with this method. The flash brightness and duration may not be enough for the photodetectors to gather enough light and thus produce insufficient contrast.

It is necessary to perform image enhancing functions, detect the edges of the lint to form blobs and detect the lint particles. A second literature study therefore focused on software. After this study, it was possible to implement the features in the software environment that was concluded from the first literature study.

2 LITERATURE RESEARCH

2.1 Industrial Background: Linting

Linting is the process where loosely bonded particles detach from the surface of paper. These particles are formed during paper manufacturing and during high-speed offset printing (Nguyen, 2015). Linting has various negative effects. For example, because the lint accumulates on the printing blankets there is a reduction in image quality and in production efficiency due to repeatedly having to stop the press to clean the printing blankets (Lestiani, Batchelor, & Banham, 2013).

The physical size of the particles is quite small. Various trial runs by Lestiani et al. (2013) have shown that the number of particles per square meter decreases exponentially with the particle size (Figure 1).

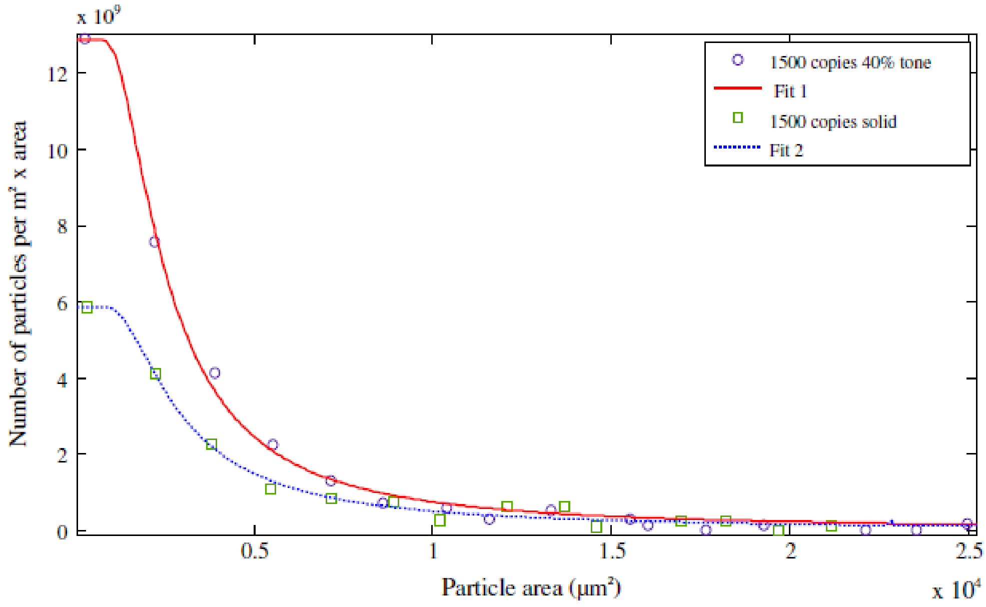


Figure 1. Number of lint particles per m^2 particle area (Lestiani et al., 2013)

Due to resolution constraints of an area scan camera the smallest size distributions, which make up the bulk of the particles (Figure 2) cannot be detected.

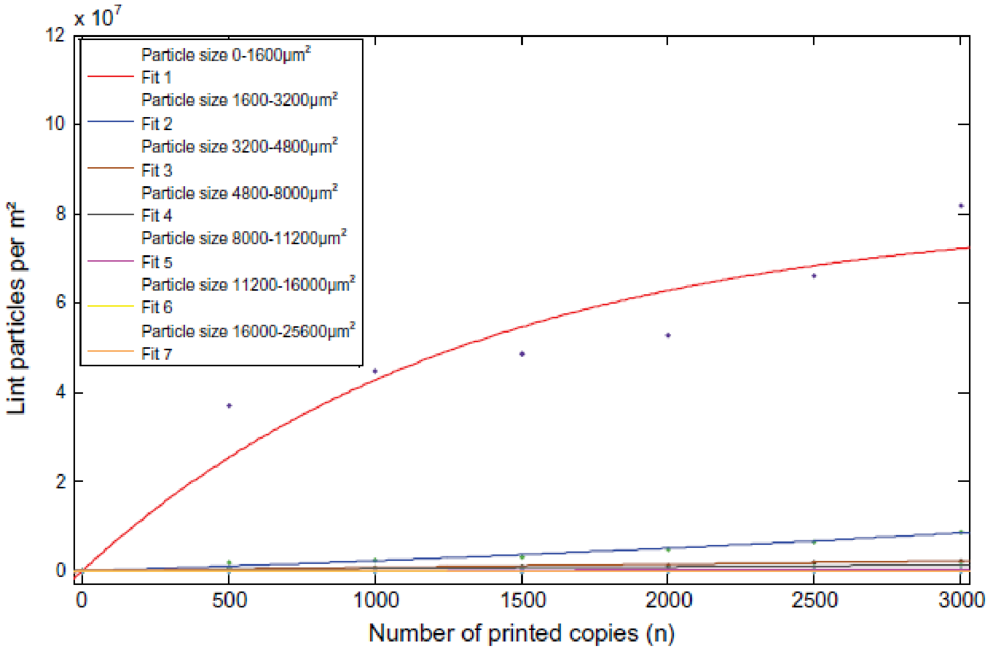


Figure 2. Measured number of lint particles per m² vs. number of printed copies (Lestiani et al., 2013)

These particles are detected by HAMK with a laser diffraction system at a volumetric flow of 27 litres per minute. This thesis is therefore focussed on detecting the larger lint particles starting at ten microns. The minimum speed of the particles is limited to around five metres per second to make sure the particle movement does not stop due to Stokes' law.

Information about the amount of lint formed during paper manufacturing and high-speed offset printing can be beneficial. For example, in order to attenuate problems connected to linting paper manufacturers must be able to monitor the linting level and adjust this according to the tolerances of each printer (Brouillette et al., 2006). Research by Amiri et al. (2004) has also concluded that increased pulp quality and paper coating reduces linting propensity. Thus, an objective lint measurement system can be used to test the quality of the supplied paper.

2.2 **Sensor**

The camera sensor contains the photodetectors which convert the photons of the incoming light to electrons. The amount of charge accumulated is proportional to the photon arrival rate (scene luminance) and the exposure interval (Corke, 2011). In this section, the two most common sensor technologies will be discussed.

2.2.1 Charge Coupled Device (CCD)

A CCD sensor has one common AD convertor for all photodetectors (Figure 3). The charge of every detector must therefore be read out one by one. CCD has the advantage of a high fill factor since there is no loss of space between the photodetectors. The light sensitivity and thus the quality of the image will be higher than with a comparable CMOS sensor. (Batchelor, 2012)

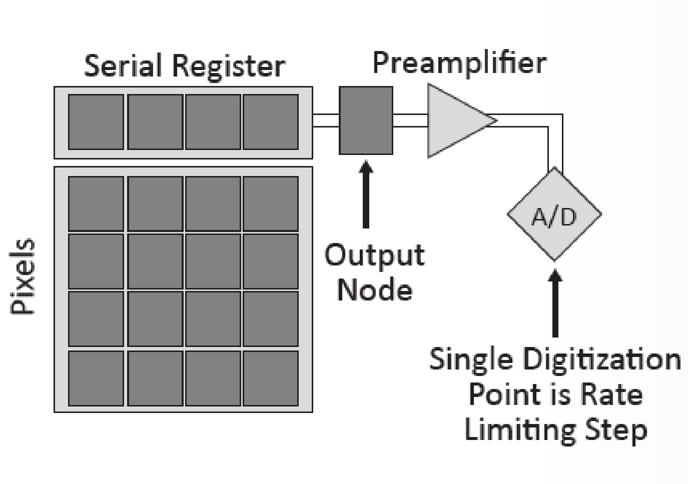


Figure 3. CCD sensor (Qimaging, n.d.)

There are two limitations to the CCD technology however. Firstly, because each line is read out sequentially it is not possible to select the region of interest (ROI). Some CCD cameras do have to capability of selecting the ROI, however doing so doesn’t reduce the sensor readout time as CCDs must always readout complete lines (Stemmer Imaging, n.d.). Secondly, blooming is an effect that happens when two photodetectors gather too much light and get saturated, because of this the charge could be transported to nearby photodetectors resulting in bright stripes in the image.

2.2.2 Complementary Metal Oxide Semiconductor (CMOS)

In a CMOS sensor each pixel is supplied with individual control and readout circuits seen in Figure 4 (Batchelor, 2012). Therefore, the fill factor is lower though this can be corrected with micro lenses that direct the light to the photosensitive parts. This layout has the advantage that every pixel can be independently selected so the region of interest can be easily defined and that the frame rate is higher.

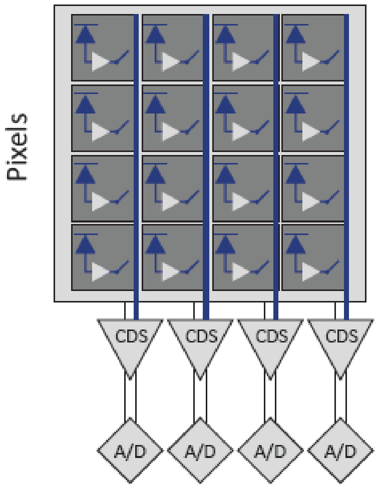


Figure 4. CMOS sensor *(Qimaging, n.d.)*

2.2.3 Line Scan Sensor

A line scan sensor consists of only one line of photodetectors (Figure 5). This means there is no need for vertical transport and therefore much less time to transfer the sensor image information into the readout register is required. One-line sensors are characterised by a very short exposure time of about 25µs. Meaning a very bright light is required to expose the sensor. For these reasons a line scan sensor is often used for inspection of small objects moving at high speeds. (Batchelor, 2012)

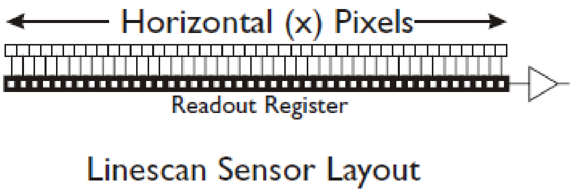


Figure 5. Line scan sensor principle *(DALSA, n.d.)*

This would have led to believe it to be an ideal solution for this subject. However, synchronization of the movement between the object and the camera is required to ensure a constant aspect ratio since an object is formed by adding the lines together at a known speed. This would make a line scan sensor ideal for objects on a conveyor.as seen in Figure 6 (DALSA, n.d.)

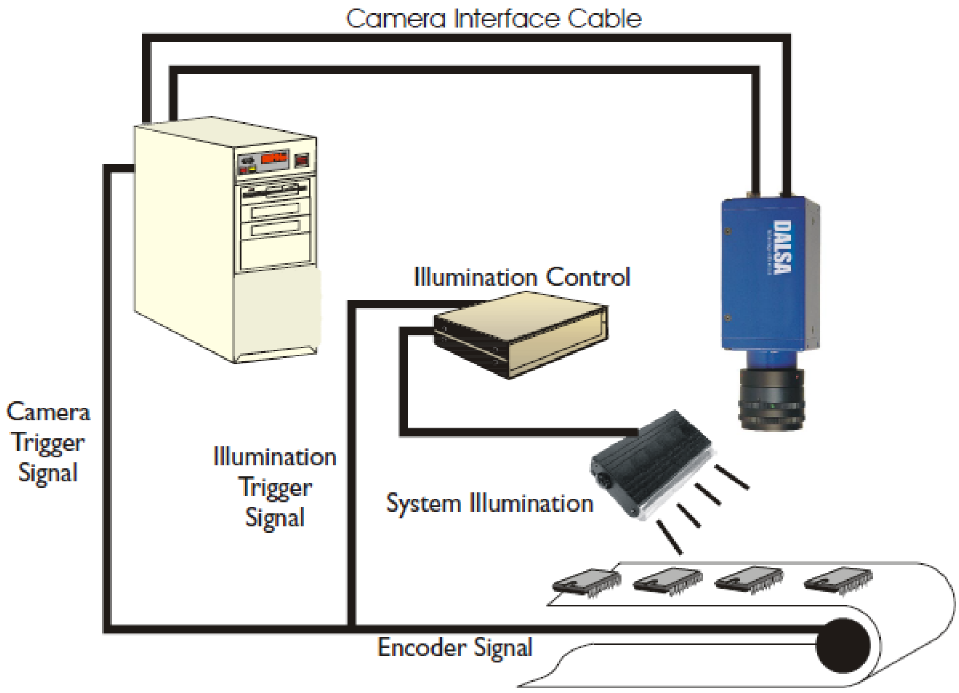


Figure 6. Line scan and conveyor principle *(DALSA, n.d.)*

For this subject, the object speed is not constant or measured since different sizes of lint will move at different speeds. Therefore, a line scan sensor cannot be used.

2.2.4 Conclusion

The advantage CCD has over CMOS is image quality. This however is not a defining characteristic for this subject. The advantages that CMOS possess such as higher framerate due to all camera functions being placed on the image sensor, the ability to define a region of interest and no blooming are of higher importance. Therefore, the sensor of choice will be a CMOS sensor.

2.3 **Exposure Control**

Since the minimum speed of the lint is about five meters per second a lot of images need to be taken every second. Therefore, the sensor exposure and readout time is very important. The shutter mode is thus of significant importance in the selection of a sensor. With a CMOS sensor, there are two shutter modes available, the global shutter and the rolling shutter.

2.3.1 Global Shutter

When the frame start trigger is given the exposure starts for all lines that were selected as the ROI. At the end of a frame the exposure ends for all lines and the pixel readout begins until all pixel data is read (Figure 7). Therefore, all of the pixels expose at the same time and the image brightness is uniform throughout the frame, less problems also occur for objects in motion. (Qimaging, n.d.)

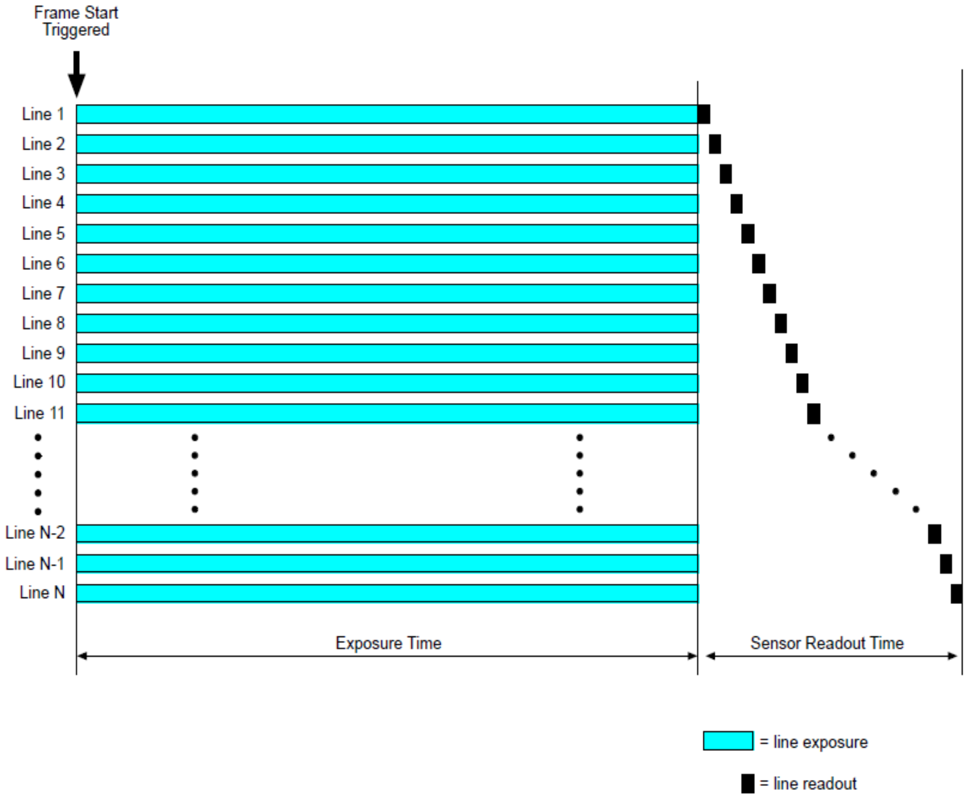


Figure 7. Global shutter (Basler, 2017)

Once the exposure is complete, each pixel simultaneously transfers its charge to a transistor waiting to be digitized. The global shutter mode has no spatial aberrations for fast moving objects, however it has a significantly reduced frame rate. An advantage that CMOS has over CCD is that it can maximize frame rate by not waiting for the entire frame readout to complete. In global shutter mode this advantage is lost since in this mode the overlapping capability is gone and the frame rate is reduced. (Qimaging, n.d.)

2.3.2 Rolling Shutter

Rather than waiting for an entire frame readout, each individual frame is now able to begin the next frame's exposure after completing the readout of the previous frame (Figure 8). The time delay (temporal offset trow) means that the exposure for each row is the same, but the time they start exposure is not. (Basler, 2017) The frame rate is determined by how quickly the readout process can be completed, decided by the speed of the A/D convertor and the number of rows on the sensor (Qimaging, n.d.).

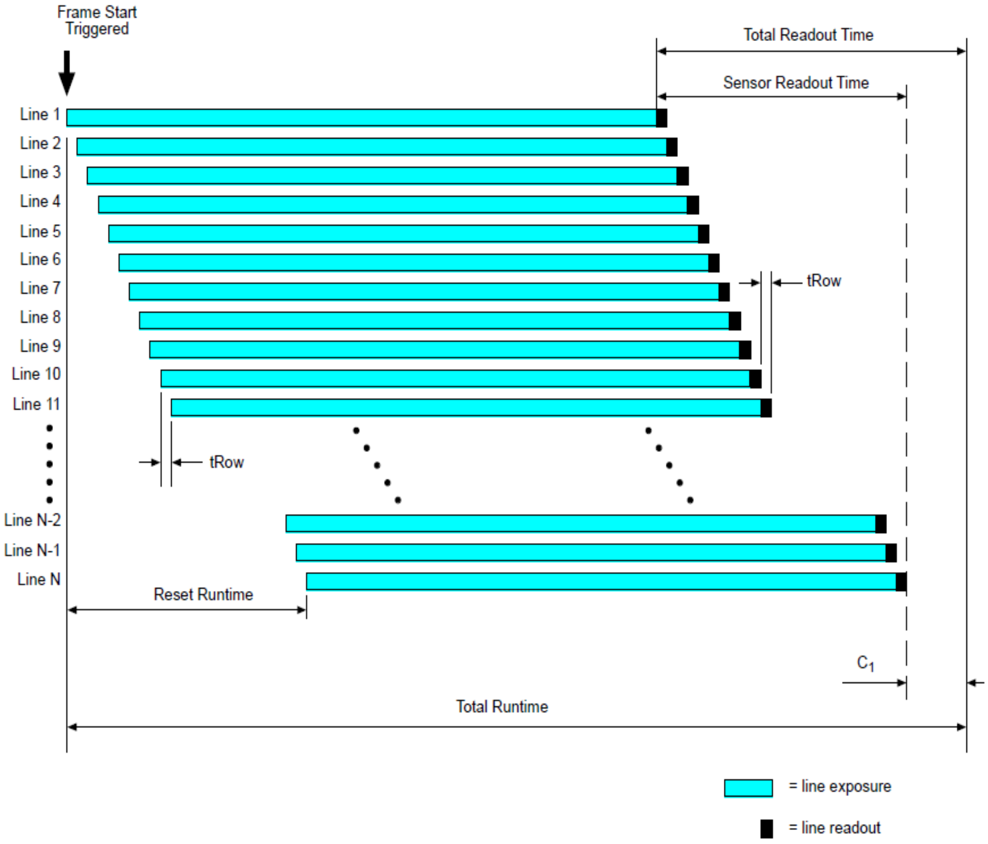


Figure 8. Rolling mode *(Basler, 2017)*

The overlapping behaviour and time delay, while fast, has some disadvantages. When imaging moving objects, as is the case of this subject, considerations concerning the size and speed of the object must be made since spatial distortions due to the time delay between each row's exposure are possible. (Qimaging, n.d.)

The total readout time is defined by the sensor readout time plus the exposure overhead c1 (Equation 1), which is to prepare the sensor for the next acquisition (Basler, 2017).

$$Total\ readout\ time = \ trow \cdot ROI\ height + c1 \qquad (1)$$

2.3.3 Best of Both Worlds

To maximize performance, it’s possible to achieve global exposure with a rolling shutter by using strobe lighting (Figure 9). This way the advantage of high framerate from rolling shutter mode is kept while spatial distortions are avoided thanks to the global exposure created by using a correctly timed flash. (Qimaging, n.d.)

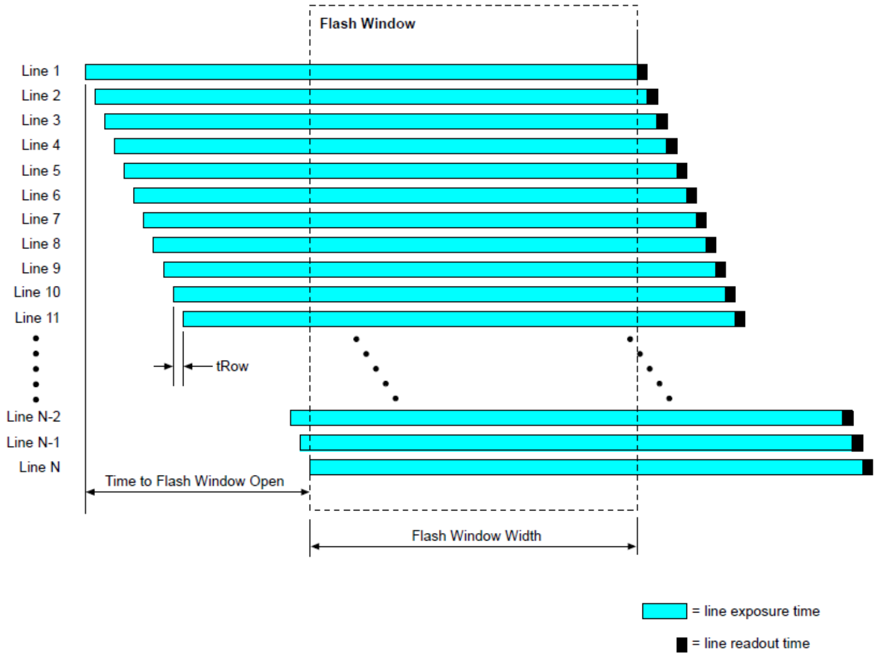


Figure 9. Rolling shutter flash window (Basler, 2017)

Precise synchronization between the camera and flash lighting is therefore required. To ease the synchronization Basler cameras, offer an output that signals the ideal time to start the flash lighting i.e. the time all rows are exposed in case of a rolling shutter or when the exposure becomes active in case of a global shutter. The schematic of the output line can be seen in Figure 10.

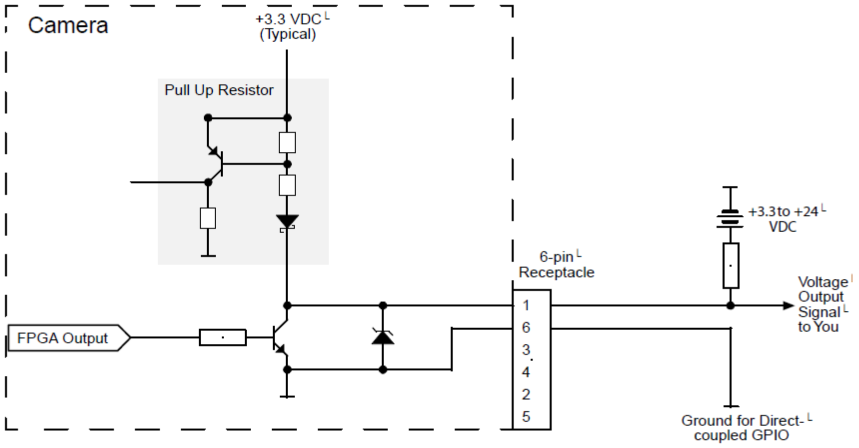


Figure 10. Direct-coupled general purpose output line (Basler, 2017)

2.3.4 Conclusion

To conclude, the ideal camera would have to be an area scan camera with a CMOS sensor capable of using the rolling shutter mode and with a sufficient resolution to detect the smallest feature. Since HAMK has previous experience with Basler cameras a Basler camera will be selected.

Since a rolling shutter requires the use of strobe lighting to avoid spatial distortions most CMOS sensors implement a global shutter. Thus, there are a lot of high framerate cameras with a global shutter. These use a fast readout mode which improves framerate whilst sacrificing image quality. Both readout modes are therefore possible assuming the framerate is sufficient.

2.4 **Lens**

The two most commonly used lenses are the conventional and the telecentric lens (Figure 11).

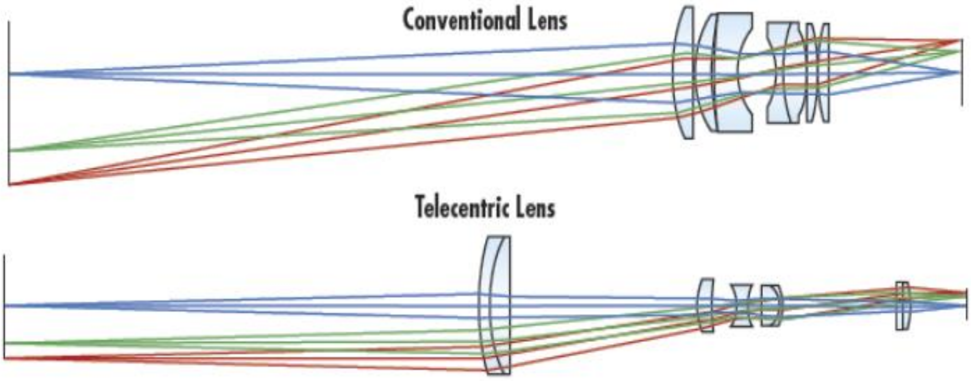


Figure 11. Conventional lens angular field of view versus telecentric lens zero angle field of view (Edmund Optics, n.d.-a)

A conventional lens behaves like the human vision. If the distance between an object and lens increases the angular field of view makes it so the magnification decreases. The angular field of view therefore decreases accuracy due to a perspective error (Edmund Optics, n.d.-a). Since the lint moves in a glass tube with a depth of three millimetres, the size of the lint will vary due to the depth perception error.

A telecentric lens has a constant, non-angular field of view. At any distance, the field of view will remain the same. Therefore, the magnification of the objects does not change in respect to the depth (Figure 12). (Edmund Optics, n.d.-a)

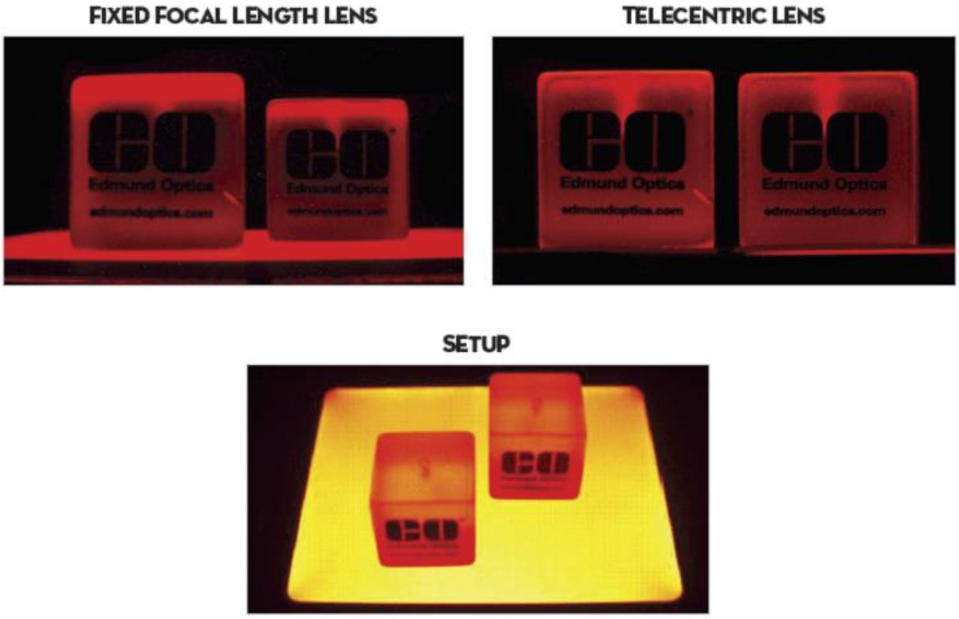


Figure 12. The telecentric lens has no depth perception error in comparison to a conventional lens *(Edmund Optics, n.d.-a)*

There are other advantages when using a telecentric lens. Telecentric lenses have a larger depth of field due to symmetrical blurring on either side of best focus, whereas conventional lenses blur asymmetrically. This means that features such as edges retain their centre of mass. Therefore, an accurate measurement of the lint can still be made even if the object is beyond best focus as long as the contrast remains high enough. (Edmund Optics, n.d.-a)

Telecentric lenses also have lower distortion values than conventional lenses. Distortion decreases measurement accuracy since the position of an object appears to be at a different location. When using telecentric lenses there is therefore no need to calibrate out distortion, thus increasing processing speed. (Edmund Optics, n.d.-a)

2.5 **Considerations when Selecting a Camera and Lens**

2.5.1 Spatial Resolution, Field of View and Frame Rate

Image sensors break the image plane into discrete areas, usually squares. The signal from the sensor is thus sampled. When sampling variations occur over many samples the sampling effect is minimal. Unfortunately, when detecting small particles, the signal changes by a large amount from one location to the next, thus the sampling will produce results that no longer resemble the original intensity variations. (Batchelor, 2012)

If a circular object with a diameter smaller than a pixel, which is very probable in lint detection, then the camera will reproduce the object as a square of one pixel (Figure 13). If the object however falls on the vertices of four pixels then it will be reproduced as a square of four dimmer pixels. If the object has the same diameter the image reproduction is still not faithful. (Andor, n.d.)

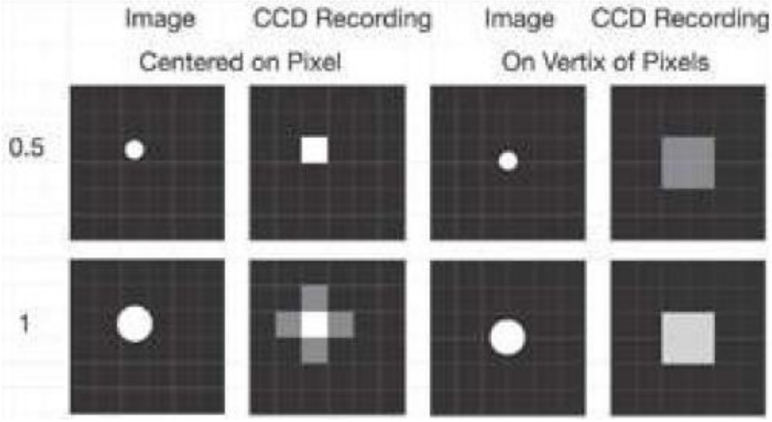


Figure 13. Circular Object Image Reproduction (Andor, n.d.)

Therefore, the Nyquist theorem states that the frequency of the digital signal should be twice that of the analog signal. In imaging terms, the theorem recommends a sampling rate of two pixels relative to the image size of the object. Adequate resolutions can thus only be achieved if at least two samples are made for each resolvable unit. (Andor, n.d.)

The resolutions is thus defined by Equation 2 (National Instruments, 2014):

$$Resolution = 2\ \frac{FOV}{Smallest\ feature} \tag{2}$$

The field of view also decides the number of images that need to be taken every second. Choosing the field of view is thus a conundrum since a wider field of view will decrease the required frame rate, however it will increase the resolution required to detect the smallest feature. For this thesis, the field of view is defined by the mechanical design.

2.5.2 Pixel and Sensor Format

Since a low duration strobe light is researched the amount of light the sensor can gather is rather limited. A large pixel format means there is more room to gather light. Larger pixels also have a higher signal-to-noise ratio and greater dynamic range. A larger sensor format will increase the size the individual pixels can be and is thus best selected as high as possible. (Sally Wiener & Daniel, 2012)

However, due to the Nyquist theorem the imaging pixel size should be half the size of the smallest feature (University of California; Berkeley, n.d.). For ten-micron particles, the pixel size can maximally be 5 microns.

2.5.3 Focal Length

The focal length defines the field of view and working distance and thus also indirectly the smallest detected feature. The shorter the focal length the shorter the distance needed to obtain the same field of view compared to a longer field of view lens. The focal length is thus best kept short. There are however also a few disadvantages to a short focal length lens. Firstly, a short focal length lens can cause distortion and therefore variations in the angle with respect to the working distance. Secondly, short focal length lenses have difficulties covering larger sensor sizes. (Edmund Optics, n.d.-b)

2.6 **Available Hardware at HAMK**

The selection of the optimal lighting is dependent on the camera and lens used. At HAMK there are two camera and lens vision setups available (Table 1-4). The characteristics of both are summarized in the two tables below.

Table 1. Basler acA 1920-25um specifications

Camera Basler acA 1920-25um	
Resolution	1920 x 1080 pixels
Framerate	25 fps
Pixel Width	2,20 µm
Pixel Height	2,20 µm
Sensor Diagonal	4,85 mm
Sensor Width	4,2 mm
Sensor Height	2,4 mm
Shutter type	Electronic rolling shutter
Interface type	USB 3

Table 2. Basler C125-0618-5M specifications

Lens Basler C125-0618-5M	
Focal Length	6 mm
Aperture Range	f/1.8 to f/22
Focus Range	0,1 m to infinity
Wavelength Range	Visible range (400 to 700 ηm)

Table 3. Basler acA 1300-200um specifications

Camera Basler acA 1300-200um	
Resolution	1280 x 1024 pixels
Framerate	200 fps
Pixel Width	4,80 µm
Pixel Height	4,80 µm
Sensor Diagonal	7,87 mm
Sensor Width	6,14 mm
Sensor Height	4,92 mm
Shutter type	Electronic global shutter
Interface type	USB 3

Table 4. Basler C125-0818-5M specifications

Lens Basler C125-0818-5M	
Focal Length	8 mm
Aperture Range	f/1.8 to f/22
Focus Range	0,1 m to infinity
Wavelength Range	Visible range (400 to 700 ηm)

It can already be concluded that with 200 frames per second the second vision setup will likely perform better. However, the theoretical possibility of fulfilling the goals will be calculated for both setups in 3. The camera of setup two also maximizes light gathering performance with the bigger sensor size, yet is small enough at 4.8 microns to use two pixels for the smallest detected feature as stated by the Nyquist theorem for the goal of 10 microns.

2.7 **Light**

The lighting is one of the most essential elements of a vision setup. A good lighting setup will accentuate the object of interest. Therefore, it is important to make a calculated choice on the type and placement of the lighting. This is not an easy task since humans do not see light directly, but rather the reflection of light. It is this process of reflection that decides how an object is seen by the camera. The lighting influences this process. (Corke, 2011)

To make a calculated choice, knowledge of the type of lighting available and its advantages and disadvantages are required. Also, knowledge of the camera quantum efficiency (QE), which is the ratio of collected electrons to the number of incoming photons, is important to maximize the light the camera receives (Batchelor, 2012). For example, if every photon is converted into an electron the QE will be 100 percent. Lastly, knowledge of the different illumination techniques is required (Daryl, 2017).

2.7.1 Features of Interest

The goal of this thesis is to detect the lint particles. To differentiate the lint from the background edge detection will be required. Therefore, maximizing the contrast of the lint against the background is very important.

2.7.2 Light Types

The most widely used lighting types are fluorescent, led and quartz halogen. Less commonly used are xenon and metal halide. Some defining characteristics of each type are seen in Figure 14. (Martin, Practical Guide to Machine Vision Lighting, 2012.)

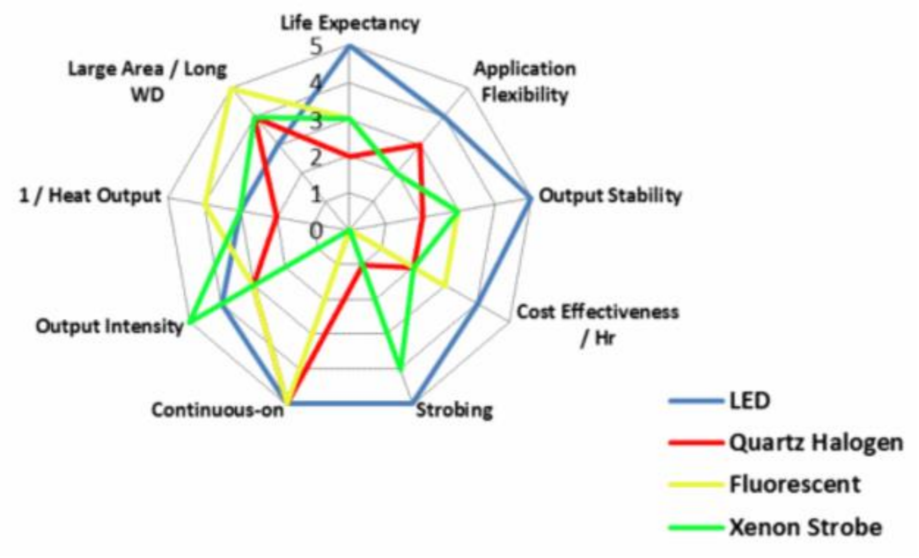


Figure 14. Comparison of lighting types *(Daryl, 2017)*

As can be seen unless high output intensity or a large area are required then LED is most often used. The advantages of led are thus:

- Flexibility: the light colour can be chosen depending on the QE of the sensor and the feature of interest.
- Long life expectancy;
- Strobe lighting is possible;
- Quick response time.

Since output intensity is very important due to the short exposure time another good possibility could be xenon lighting. However, considerations must be made about the ease of implementation and safety since the flash will be created by an electronics student. Xenon requires a very high operation voltage to ionize the gas, is more difficult to implement and has a shorter lifecycle than LED (Mars, 2009). Therefore, for this research LED lighting will be used.

2.7.3 Quantum Efficiency and Spectral Output

In applications requiring high light intensity it is useful to match the spectral sensitivity of the sensor to the output of the LED light (Daryl, 2017). Figure 15 shows the spectral output of various light sources.

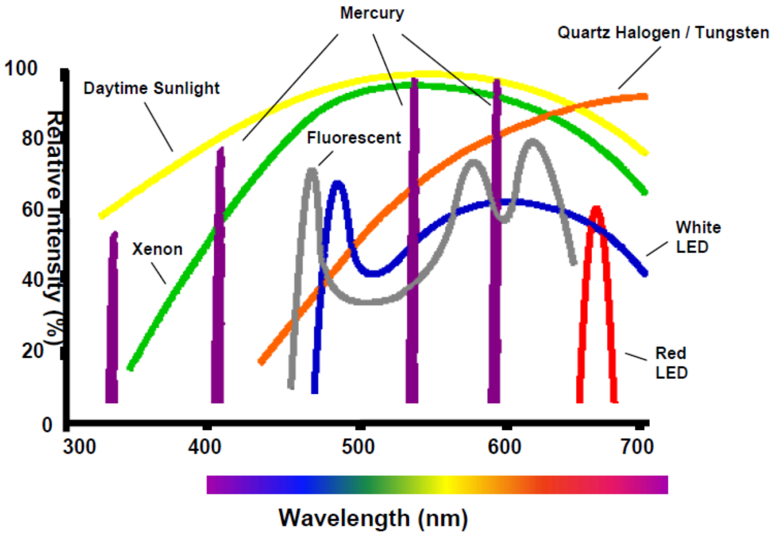


Figure 15. Comparison of spectral output from various light sources *(Daryl, 2017)*

The spectral sensitivity for different wavelengths of the sensor can be seen in Figure 16.

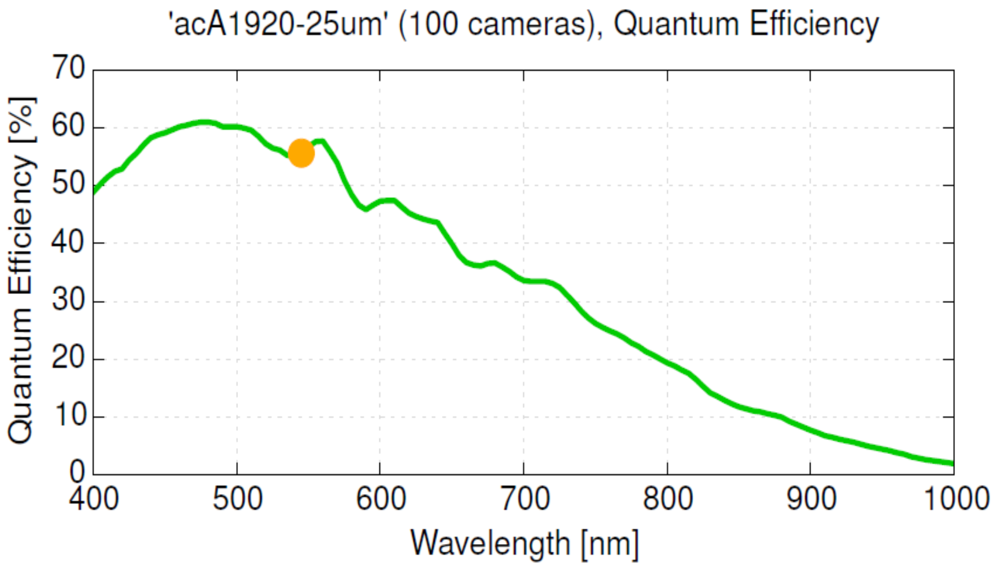


Figure 16. Quantum efficiency (Basler, n.d.)

The sensor is most sensitive in the 450-550ηm wavelength. This wavelength is well represented by white LEDs. Figure 17 shows the spectral transmittance of the Basler lens.

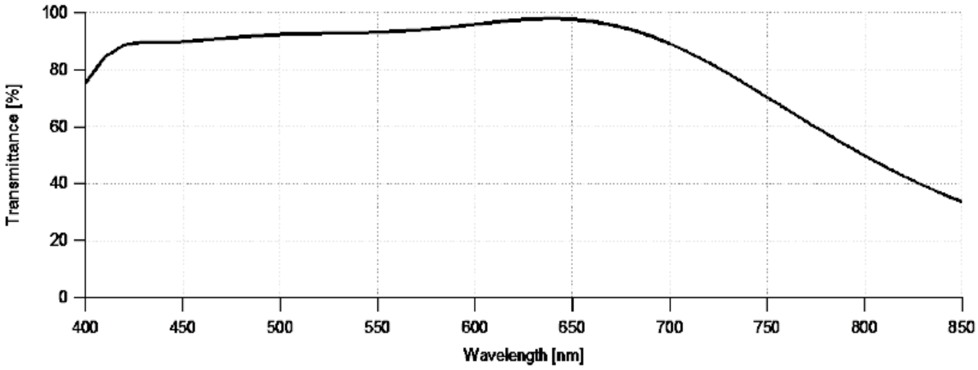


Figure 17. Lens transmittance (Basler, 2016)

The lens is most sensitive in the visible spectrum. A spectrum also well represented by white LEDs. It can be concluded that for this thesis white LEDs are the most ideal type of lighting.

2.7.4 Illumination Techniques

Back lighting generates instant sharp contrast as it creates dark silhouettes against a bright background (Figure 18). The disadvantage is that surface details are lost, this however is not a concern for detecting the lint. (Daryl, 2017)

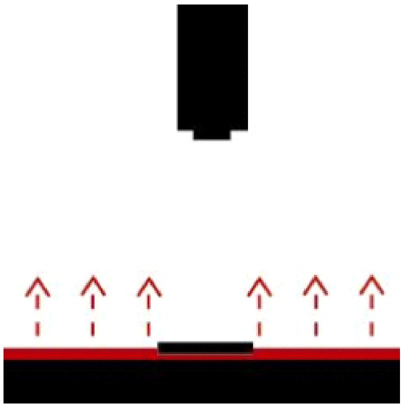


Figure 18. Back lighting *(Daryl, 2017)*

Also, Pau & Olafsson (1991) state that back lighting is the most obvious illumination method for width and length measurements. Which is beneficial for further research in calculating the mass lost from linting.

2.7.5 Micro Second Strobe Lighting

Since the sensor is exposed to the strobe light for only one microsecond it is likely that a single LED cannot sufficiently expose the sensor. A way must therefore be found to increase the brightness. Since a LED will generally fail from too high of a temperature rather than overcurrent, a possibility researched by Willert, Stasicki, Klinner, & Moessner (2010) is to temporally overdrive the LED. Overdriven LEDs with short duration, pulsed current beyond their continuous current damage threshold can generate light pulses sufficient to illuminate and image micron-sized particles (Willert et al., 2010).

When overdriving a LED, the output doesn't increase linearly with the current since some energy is lost in heat. An estimate by Wilson, Gustafson, Lincoln, Murari, & Johansen (2014) concludes that ten times overdriven LEDs (ten times the maximum current) would result in an increase of output intensity by 550 percent. There is no way to calculate the output intensity increase. Therefore, the ideal overdrive amount would have to be tested by increasing the current in small steps. A second possibility is to use multiple LEDs since the output intensity increases linearly with the number of LEDs placed.

2.8 **Programming Environment**

There are many machine vision programming environments available. For this thesis, only those that are available to HAMK free of charge are considered. Commercial software such as Mvtec Halcon, Cognex Visionpro, etc. are not considered.

An open source software package is OpenCV which can be programmed using C++ or Python, though only C++ is supported by Basler. Programming on hardware level like C++ has the advantage of faster performance (Fleyeh, 2014).

Another option are academic software packages like Mathworks Matlab and National Instruments Labview. While OpenCV is generally better for implementation, Matlab and Labview are more interesting for research due to ease of programming (Fleyeh, 2014). For example, working with matrices is much easier in Matlab than in C++ and images are essentially matrices. This thesis is research oriented and not about industrial implementation therefore an academic software package like MATLAB will be used.

2.9 Contrast Improvement

Most segmentation techniques divide the pixels in binary values by using some form of threshold. Sharp contrast is thus advisable to increase the ease and accuracy of finding a suitable threshold. An image lacks contrast when there are no sharp differences between black and white. (Corke, 2011)

To improve contrast the grey scale distribution must be altered. This is generally done with a monadic operation. A monadic operation means that each output pixel is a function of the corresponding input pixel (Equation 3). (Corke, 2011)

$$O[u, v] = f(I[u, v]) \tag{3}$$

The result is an image of the same size as the input image, as can be seen in Figure 19.

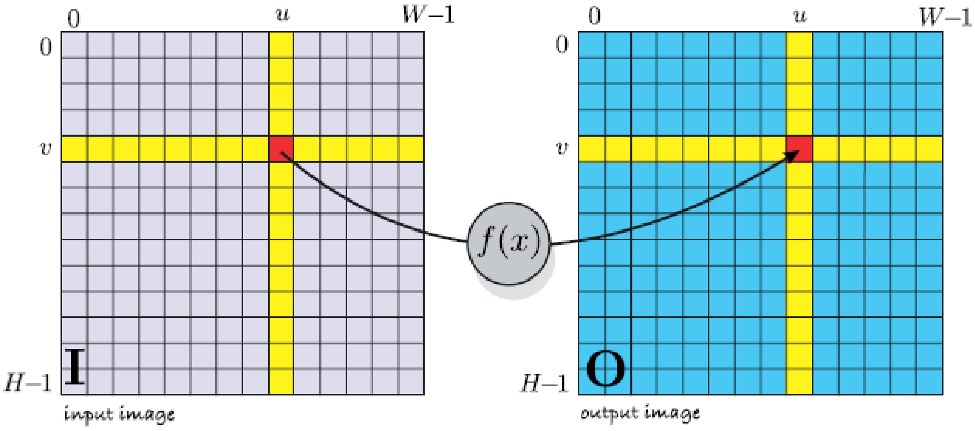


Figure 19. Monadic operation: output (Corke, 2011)

The following two common examples highlight why contrast adjustment might be necessary. The first example has a histogram which shows three significant peaks (Figure 20). However, dependent on where the threshold is chosen there may be nine or more peaks. Contrast normalization is necessary to give a clear idea on where to select the threshold. (Corke, 2011)

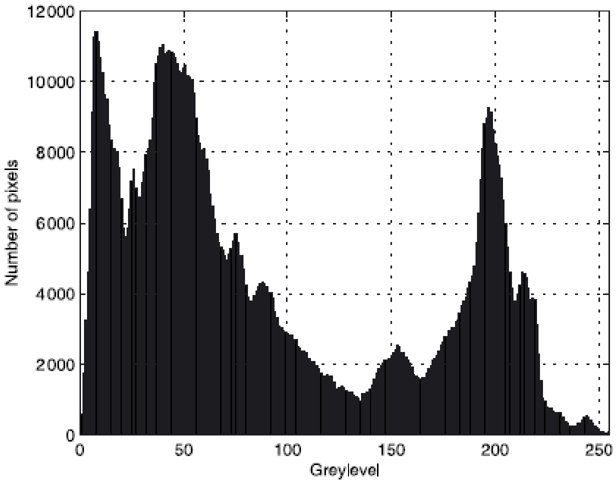


Figure 20. Example one (Corke, 2011)

The second example shows an image where the pixels do not span the full range of available grey levels (Figure 21). A linear mapping such as histogram normalization can be used which ensures that the pixel intensities are distributed linearly. (Corke, 2011)

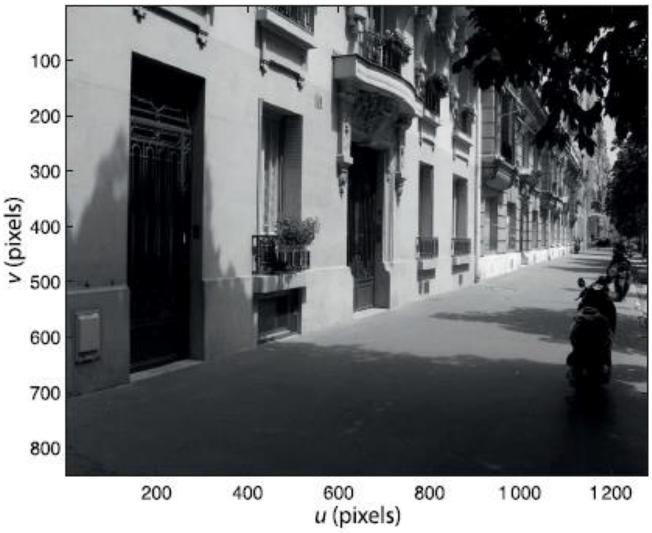


Figure 21. Example two (Corke, 2011)

Since the optimal threshold can be selected with Otsu’s method which separates an image into two clusters of pixels so that the variance of values within each cluster is minimized and the variance of the values between the two clusters is maximized. Otsu’s method thus assumes that the histogram has just two peaks. Using Otsu’s method on the image above would therefore result in poor pixel classification. (Corke, 2011)

To improve contrast a function with a slope higher than one can be used, seen in Figure 22.

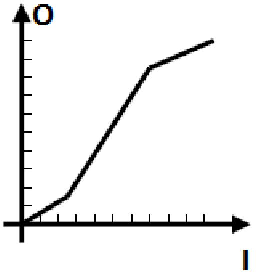


Figure 22. Contrast improvement

As can be seen it will project a small region of input pixels over a larger range of output pixels, thus normalizing the contrast.

2.10 **Filter**

Filters are applied as a linear spatial operator (Corke, 2011).

$$O[u,v] = f(I[u+i, v+j]) \quad (4)$$

Each pixel in the output image is a function of all pixels in a region surrounding the corresponding pixel in the input image as seen in Figure 23 and Equation 4 (Corke, 2011).

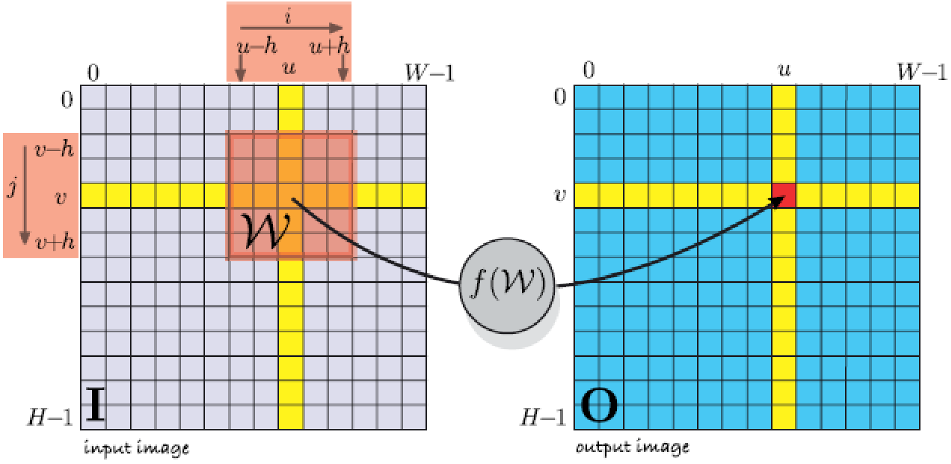


Figure 23. Spatial operation (Corke, 2011)

2.10.1 Edge Detection

Edges are characterised as abrupt changes in pixel intensity. Edge detection is the process of identifying these sharp discontinuities. (Maini & Aggarwal, 2009) Contrast improvement may be required since effects such as refraction and poor focus result in boundaries with a gradual change in intensity, thus no abrupt change (Argyle & Rosenfeld, 1971). There are two main principles for edge detection. Firstly, the gradient based method which looks for edges at the maxima and minima of the first derivative. Secondly, the Laplacian method which looks for zero crossings in the second derivative. Since a derivative will also increase the noise some form of noise filtering is likely necessary. (Maini & Aggarwal, 2009)

There are several types of edge detectors of which the most optimal generally depends on the type of edge that must be detected. Research by Maini & Aggarwal (2009) concluded that Canny edge detection yields the best results under almost all scenarios, though it is also the most computationally expensive.

2.11 **Image Segmentation**

Image segmentation is the process of partitioning an image into multiple segments. It is a necessary process to locate the lint in the images (Kaewtrakulpong & Bowden, 2001). The following sections will review all the related image segmentation techniques used in this thesis.

2.11.1 Frame Differencing

Frame differencing uses the absolute difference between frames to divide an image into changed and unchanged regions. Since only the lint moves, the changed region is associated with the lint and the unchanged with the background. (Lee & Eddins, 2003)

Another possibility, which is also the easiest to implement, would be to make an image of the background without the moving lint particles. Then all that is required is to subtract the lint recording frame by frame from the static background image. The resulting frames will only have the moving lint particles. (OpenCV, n.d.)

2.11.2 Thresholding

Thresholding is a segmentation technique for images based on the colour or grayscale value. It transforms an image into a binary image by transforming each pixel according to whether it is inside or outside the specified range. Black pixels correspond to the background and white pixels to the foreground (or vice versa). (Huang & Chau, 2008) In an ideal case the grey-value histogram has a deep sharp valley between two peaks (Figure 24 left). Since backlight will be used this is reversed as the lint will be represented as dark valleys and the bright background as a peak. The threshold can then be chosen at the bottom of the valley. In other words, sharp contrast between lint and background is required. (J. M. Prewitt and M. L. Mendelsohn, 1966)

Because not much is known about the test setup in Tampere it could be that due to non-ideal factors such as stray lighting and shadows that this ideal case is not possible (Figure 24 right) and the threshold will be difficult to select. Therefore, two threshold techniques will be researched. The first which assumes the ideal case with the threshold selected as a grey value just above the lint's grey value. Also, another method will be tested which overcomes the difficulties when no clear threshold is present.

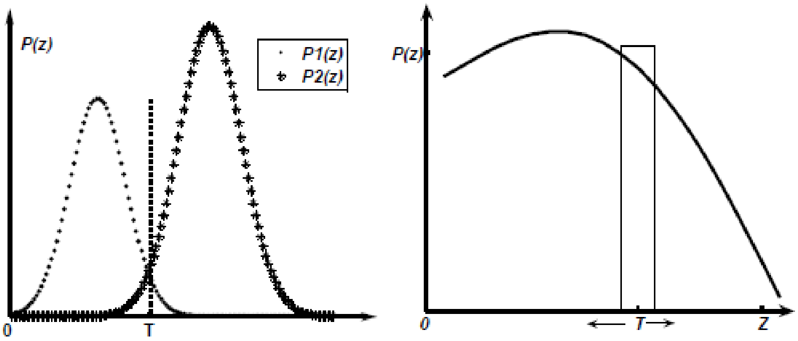


Figure 24. Gray-level density functions of two regions in an image. Left image: clear valley present for easy threshold selection, right image: non-optimal histogram for threshold selection (Huang & Chau, 2008)

One of these techniques is called valley sharpening which is proposed by Weszka, Nagel, & Rosenfeld (1974). Valley sharpening restricts the histogram to the pixels with the largest absolute values of derivative. A second method proposed by Watanabe & Group (1974) selects the threshold at the grey level with maximal amount of difference. MATLAB however uses Otsu's method proposed by Otsu (1979). It automatically selects a threshold from a grey level histogram which is selected by the discriminant criterion. This method is a very simple procedure with limited computing time and an optimal threshold is selected automatically and stable. (Otsu, 1979)

2.11.3 Gaussian Mixture Models

The Gaussian mixture-based background/foreground segmenting algorithm used in MATLAB is based on research by Kaewtrakulpong & Bowden (2001). It uses a method that models each background pixel by a mixture of Gaussian distributions. The weights of each mixture represent the time that those colours stay in the scene. The background colours are those that stay in the scene longer and are more static. (OpenCV, n.d.)

3 VISION SETUP DESIGN: CALCULATIONS

The literature study concluded some characteristics that are important when selecting the hardware for this subject. Since buying new hardware is expensive it is beneficial to look if the research cannot be completed with sufficient accuracy when using hardware currently available at HAMK. There are three hardware components that complete a vision setup. Firstly, the lighting which will be designed and created by a bachelor electronics student. Secondly, the camera which will be one of the two Basler CMOS cameras. Lastly, the lens which is unfortunately not a telecentric lens.

The laser diffraction system used to detect the smaller lint particles has a volumetric flow of 27 l/min, of which the value cannot be changed. The volumetric flow is defined by Equation 5.

$$q_v = A \cdot v \tag{5}$$

The area and speed of the lint particles can be altered slightly if the volumetric flow remains the same. The speed does have a minimum speed of 5 m/s to make sure the heavier particles do not stop moving due to Stokes' Law. An area of 90 mm^2 would lead to a speed of 5 m/s. The area of the tube is defined by the depth of field and the height.

3.1 Calculations setup 1

3.1.1 Field of View

The field of view is decided by the focal length of the lens and the working distance. The field of view is the largest area that the sensor can see. Since the lens available is a fixed focal length lens of 6mm the field of view can only be changed by varying the working distance.

The smallest feature that can be detected in the selected field of view is decided by the resolution of the sensor. Since the vertical resolution is known the ideal vertical FOV can be calculated.

$$\begin{aligned} Vertical\ FOV &= \frac{\text{Vertical resolution } \cdot \text{ Smallest feature}}{2} \\ &= \frac{1080 \cdot 10\mu m}{2} = 5{,}4mm \end{aligned}$$

3.1.2 Working distance

If the tube is made long with short height the particles will overlap decreasing the accuracy of the system. Since the four used LEDs will be placed in a 2x2 formation it makes sense to have the same height and length to even out the light distribution. This is however not entirely possible due to area being limited not the decrease the speed too much. For a 4mm depth of field estimation the maximum size the height of the tube can be is 22,5mm.

As can be seen in the field of view calculations the field of view is not sufficient to cover the complete height of the tube. Since the lens has a fixed focal length the working distance must be increased to make the field of view cover the complete 22,5 mm of the tube height. The relationship between field of view and working distance is given by Equation 6 (National Instruments, 2014).

$$\text{F}ocal\ Lenght\ \cdot\ Field\ of\ View = Sensor\ Size\ \cdot\ Working\ Distance \tag{6}$$

The working distance is thus defined by:

$$\begin{aligned} \text{W}orking\ Distance &= \frac{Focal\ Length\ \cdot\ Field\ of\ View}{Sensor\ Size} \\ &= \frac{6mm\ \cdot\ 22{,}5mm}{2{,}4mm} = 56{,}25mm \end{aligned}$$

The issue is however that no longer two pixels are used for every feature. By increasing the working distance and thus also the field of view the resolution of the sensor is no longer sufficient to detect the small lint particles. For a field of view of 22,5mm a resolution of 4500 pixels would be required. The smallest feature that can realistically be detected therefore must be increased to around 41,67 microns.

3.1.3 Depth of Field

The depth of field defines the maximum depth of the tube since it is the distance the lint can move along the optical axis while still forming a sharply focused image (Batchelor, 2012). It is calculated as the distance between the far and near focal length (Figure 25).

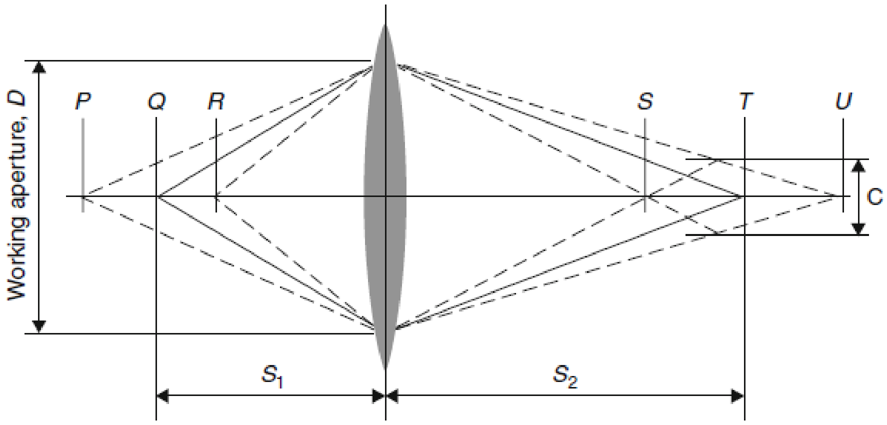


Figure 25. Depth of Field. A point in the plane Q is focused to a point in plane T. The object can move between region PR, the region between the closest focus point S1 and furthest S2, while still maintaining an optimal focus. (Batchelor, 2012)

The focal length is defined as the distance between the lens and it's focal point seen in Figure 26 (Batchelor, 2012).

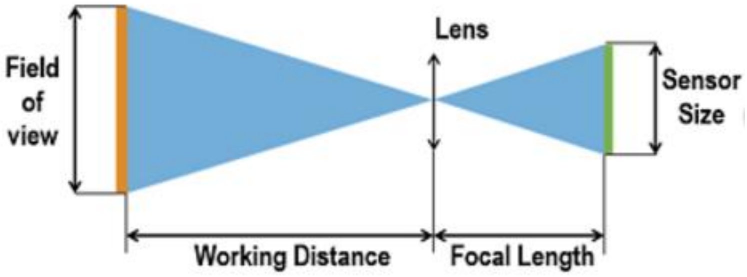


Figure 26. Relationship between different lens and sensor parameters (National Instruments, 2014)

The depth of field must be solved iteratively since it is also depended on the working distance of the camera.

Before calculating the depth of field some other parameters need to be known.

Firstly, the circle of confusion which is a small finite-sized spot size that is a measure of the level of miss-focus (Batchelor, 2012). It can be calculated with the Zeiss formula seen in Equation 7.

$$C = \frac{d}{1730} = \frac{4{,}85}{1730} = 0{,}0028 \tag{7}$$

With d the sensor diagonal.

Secondly, the distance to the closest point of acceptable focus when the lens is focused at infinity, the hyper focal distance seen in Equation 8 (Psarossy, 2013).

$$H = \frac{f^2}{N \cdot C} = \frac{6^2}{22 \cdot 0{,}0028} = 584mm \tag{8}$$

Where N is the aperture. In this calculation, the maximum possible aperture is chosen which will result in the maximum possible depth of field of the camera. This however may not be ideal since it will reduce the exposure of the sensor to light.

The depth of field is defined as the distance between the furthest and closest distance that the lint is acceptably focussed (Equation 9 and 10).

$$S\ close = \frac{H \cdot D}{H + (D - f)} = \frac{584 \cdot 58{,}25}{584 + (58{,}25 - 6)} = 53{,}46mm \tag{9}$$

$$S\ far = \frac{H \cdot D}{H + (D - f)} = \frac{584 \cdot 58{,}25}{584 - (58{,}25 - 6)} = 65{,}46mm \tag{10}$$

Where D is the distance to the lint from the front principal point. The maximum possible depth of field is thus 12mm. With a 2.8 aperture this would be 1,5mm and with a 5.6 aperture 3mm. The previous estimation for the depth of the tube of 4mm is thus acceptable.

3.1.4 Magnification

The field of view does not have the same size as the sensor (Figure 27) so some magnification setting on the lens is required.

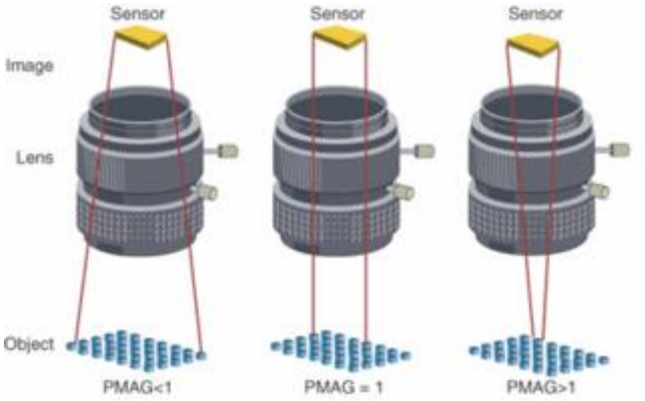


Figure 27. Magnification required when the sensor size and field of view are not similar (Blankinship, 2005)

The primary magnification (PMAG) in Equation 11 describes the ratio between the sensor size and the field of view (Blankinship, 2005).

$$\mathrm{P}MAG = \frac{Sensor\ Size}{Field\ of\ View} = \frac{2{,}4mm}{22{,}5mm} = 0{,}107 \qquad (11)$$

3.1.5 Images per Second

The number of images that need to be taken every second to make a full representation of all the lint that passes by is dependent on the horizontal field of view since the air moves horizontally, and the speed at which the lint moves (Equation 12).

$$\mathrm{H}orizontal\ FOV = \frac{\text{Horizontal resolution } \cdot \text{ Smallest feature}}{2}$$

$$= \frac{1920\ \cdot\ 41{,}67\mu m}{2} = 40mm$$

$$\mathrm{I}mages\ per\ Second = \frac{Speed}{Horizontal\ FOV} \qquad (12)$$

$$= \frac{5\,\frac{m}{s}}{40mm} = 125\ images\ per\ second$$

3.1.6 Flash Exposure Time

The exposure time of the camera is too high to reduce motion blur. Therefore, a short duration strobe light will be designed to decrease the time the lint is visible for the sensor. The ideal pulse duration of the strobe light is calculated in the next section.

The lint moves across the sensor at the known speed of v. While the flash is active for a duration of N, the lint will move a distance of v · N. The field of view is imaged by the lens onto the sensor, because the field of view is larger than the sensor size the lens reduces the size of the field of view by a factor R = BFL/D with BFL the focal length and D the working distance as can be seen in Figure 28. (Facey, sd)

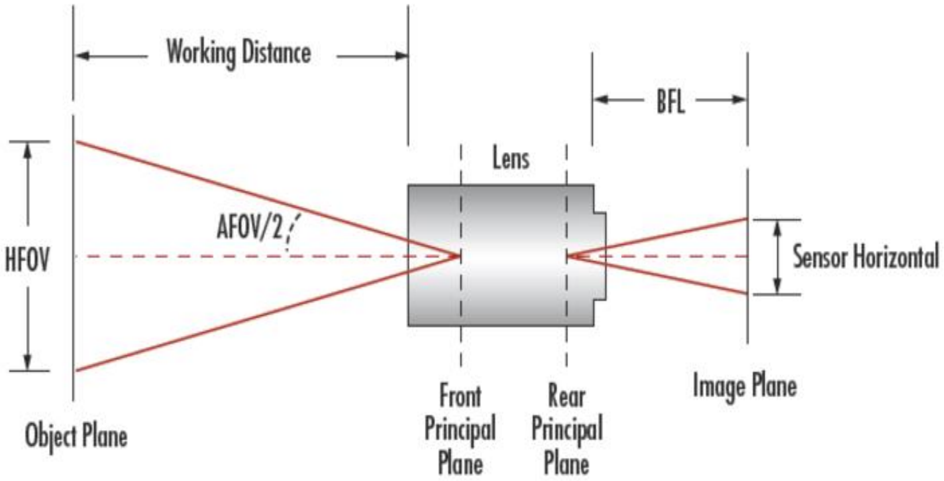


Figure 28. Detailed relationship between lens and sensor parameters (Edmund Optics, n.d.-b)

The distance the lint moves across the sensor is thus R · v · N. This distance needs to be within an acceptable limit to avoid blurring. An example would be to not allow the lint to move more than one pixel, which would be 2,2µm (C in Equation 13) an acceptable accuracy considering the smallest detectable lint particle is 41,67µm. For non-full frame camera's this value needs to be divided by the sensor crop factor (Facey, sd). The crop factor gives the ratio of the dimensions of a sensor relative to that of a 35mm full frame sensor with a diagonal of 43,3mm (Vorenkamp, 2016). The diagonal of the sensor is 4,85mm so the crop factor is 8,93.

So, the distance the lint moves across the sensor should be smaller than C divided by the crop factor (Equation 13).

$$\mathrm{R} \cdot v \cdot N = \frac{BFL}{D} \cdot v \cdot N \leq \frac{C}{8{,}93} \tag{13}$$

Therefore, the flash duration N should be:

$$\mathrm{N} \leq \frac{C * D}{8{,}93 \cdot BFL \cdot v} = \frac{2{,}2\mu m \cdot 56{,}25mm}{8{,}93 \cdot 6mm \cdot 5\frac{m}{s}} = 462\eta s$$

3.1.7 Sensor Readout Time

The readout time is defined by:

$$\begin{aligned} \mathrm{T}otal\ readout\ time &= trow \cdot ROI\ height + c1 \\ &= 35\mathrm{\mu s} \cdot 1080 + 490\mathrm{\mu s} = 38{,}3\mathrm{ms} \end{aligned}$$

This means that with the full resolution, region of interest height of 1080, only 26 images can be taken every second. This is much less than the previously calculated 125 images that are needed every second for a complete representation. As can be concluded the major stumbling block in reaching the goal is the readout time.

For the camera to process enough images the ROI must be decreased which is possible by increasing the size of the smallest detectable feature. The smallest possible feature that can be detected can be easily calculated. Since the vertical field of view cannot be further increased because of the minimum required speed always 125 images need to be taken for a full representation. The ROI required to sufficiently reduce the readout time is:

$$\begin{aligned} \mathrm{T}otal\ readout\ time &= trow \cdot ROI\ height + c1 = 35\mathrm{\mu s} \cdot \mathrm{ROI} + 490\mathrm{\mu s} \\ &= \frac{1}{125} \end{aligned}$$

Thus, the vertical region of interest (ROI) is maxed at 215 pixels. Finally, can be calculated what the smallest possible feature is that can be detected with such a limited vertical resolution:

$$\begin{aligned} Vertical\ FOV &= \frac{\text{Vertical resolution} \cdot \text{Smallest feature}}{2} \\ &= \frac{215 \cdot Smallest\ feature}{2} = 22{,}5mm \end{aligned}$$

This concludes that the smallest feature that can be detected and processed is 210µm.

3.1.8 Summary

The results of these calculations are summarized in Table 5.

Table 5. Vision setup one, summary

Field of View	22,5mm
Working Distance	56,25mm
Maximum Depth of Field	12mm
Magnification	0,107
Images per Second	125
Flash Exposure Time	462ηs
Smallest Detectable Feature	210µm

3.2 **Calculations setup two**

The calculations were performed again for setup two. The results are displayed in Table 6.

Table 6. Vision setup two, summary

Field of View	22,5mm
Working Distance	36,59mm
Maximum Depth of Field	4,3mm
Magnification	0,219
Images per Second	180
Flash Exposure Time	1,06µs
Smallest Detectable Feature	43,95µm

3.3 **Conclusion**

It can be concluded that the goal of detecting particles of ten microns is not possible with the hardware available at HAMK. The best results are possible with setup two, therefore this setup will be used. Some possible improvements could be:

- Change the goal from 10 to roughly 45 microns. This would mean that research must be done to extend the laser diffraction to also detect between 10 and 45 microns.
- Change the mechanical design. A solution could be to, for example, blow the lint on a conveyor so that all the lint moves at the same velocity. If this is the case a line scan camera can be used which is ideal for this subject.
- There are area scan camera's in Basler's catalogue that can acquire images with sufficient resolution and framerate. Though researching each one would be too time inefficient.

A visual overview of the settings of setup two is given in Figure 29.

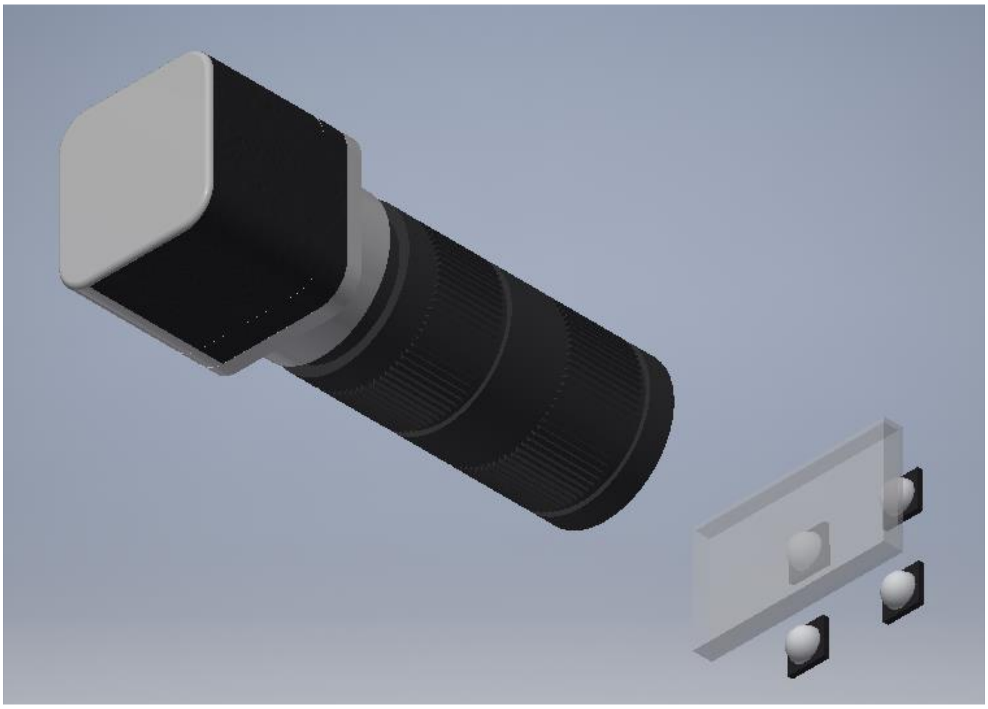


Figure 29. Vision setup overview

4 VISION SETUP OVERVIEW

Vision setup two and a strobe light created by a colleague will be used. Both elements must be synchronized when the exposure of the sensor becomes active. Programs were created to test the framerate, smallest detectable feature, segmentation performances and detect the lint passing by. An overview of the practical test setup is seen in Figure 30.

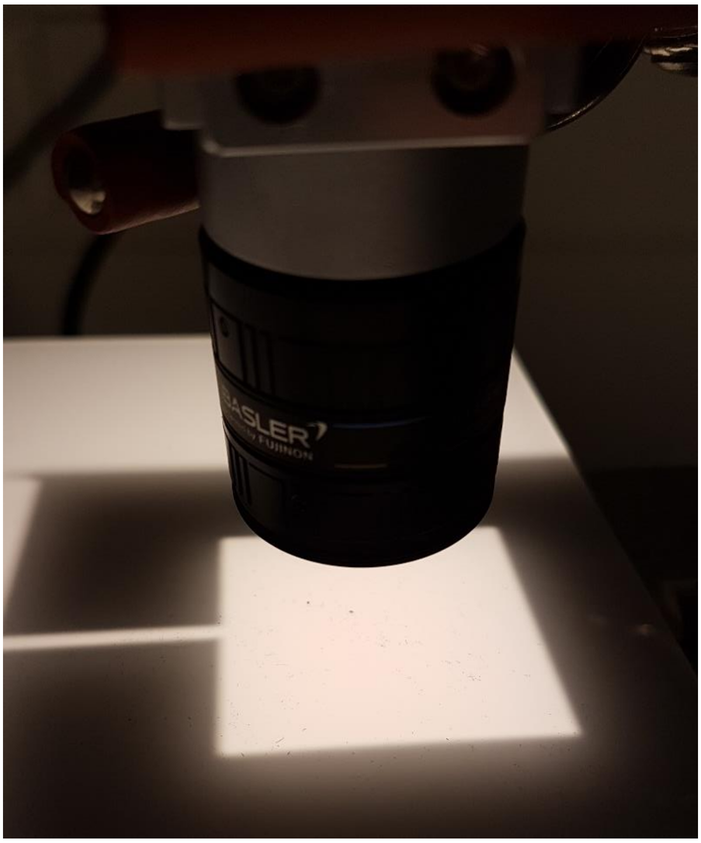


Figure 30. Vision Setup

4.1 Strobe Light

The block diagram in Figure 31 displays the various functions that make up the strobe light.

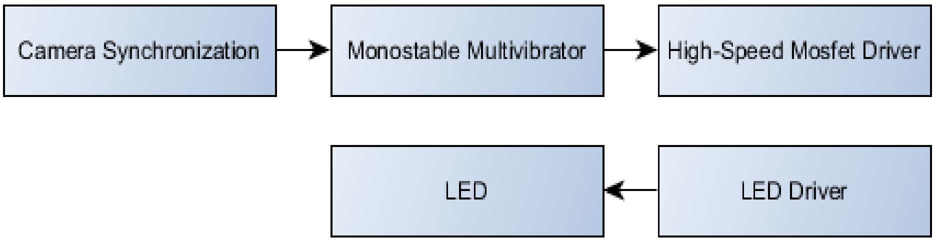
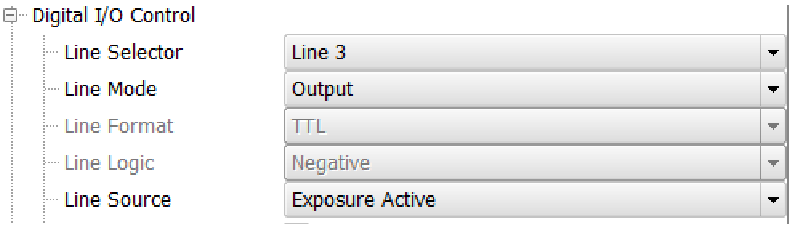


Figure 31. Strobe Light Diagram

Two PCBs were created. The first PCB contains the monostable multivibrator. The second PCB contains the MOSFET and LED driver. In this thesis only a brief overview of each component is given. More in-depth information is given by De Brabanter (2017).

4.1.1 Strobe and Camera Exposure Synchronization

One of the general-purpose output lines (GPO) will be used to synchronize with the strobe light. The output will send a signal when the exposure of the camera becomes active. It can be configured in Basler Pylon as seen in Figure 32.



Digital I/O Control

Line Selector	Line 3
Line Mode	Output
Line Format	TTL
Line Logic	Negative
Line Source	Exposure Active

Figure 32. General-purpose output line settings

The GPO has an internal voltage of 3.3V that will send 180 signals every second to the monostable multivibrator. Since the exposure time is too long, the strobe light will switch on after receiving the signal but turn off at a user defined time as explained in 4.1.2. In Figure 33 the output of the GPO can be seen.

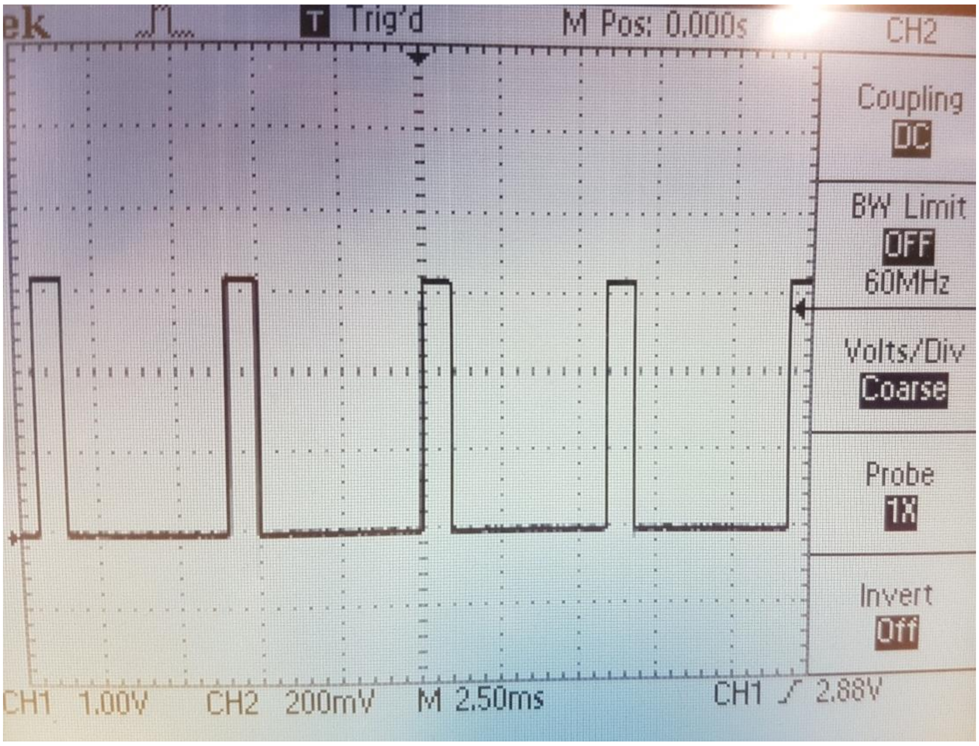


Figure 33. General Purpose Output at 180 Frames per Second

The monostable multivibrator requires an input voltage of five volts. Therefore, an external voltage source and pull-up resistor is used to bring the GPO output to five volts.

4.1.2 Monostable Multivibrator

When the monostable multivibrator gets a trigger from the camera it will give a pulse on the output pin. The pulse width depends on the resistor and capacitance value. The resistor value can be adjusted to change the duration of the output pulse between 100ηs and 1µs. The circuit of the monostable multivibrator can be seen in Figure 34.

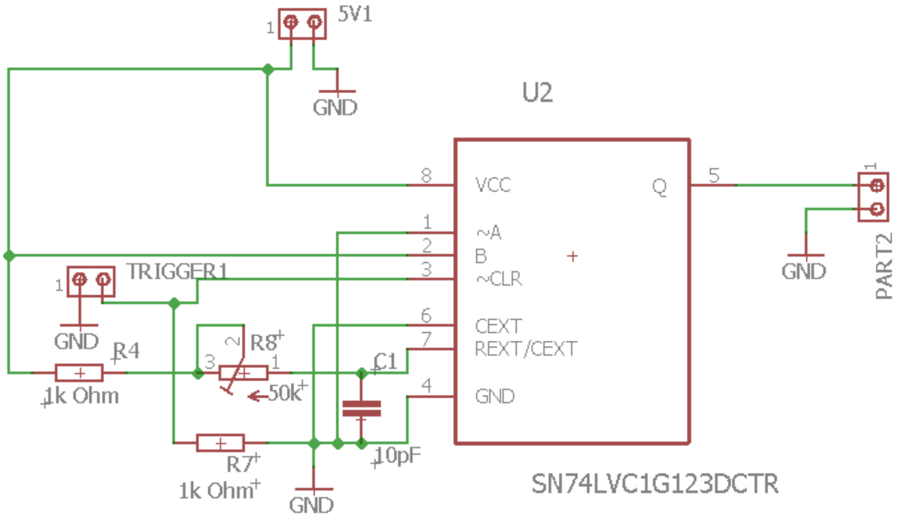


Figure 34. Monostable multivibrator circuit (De Brabanter, 2017)

In Figure 35 the potentiometer is adjusted so the pulse width is at its minimum duration of 100ηs.

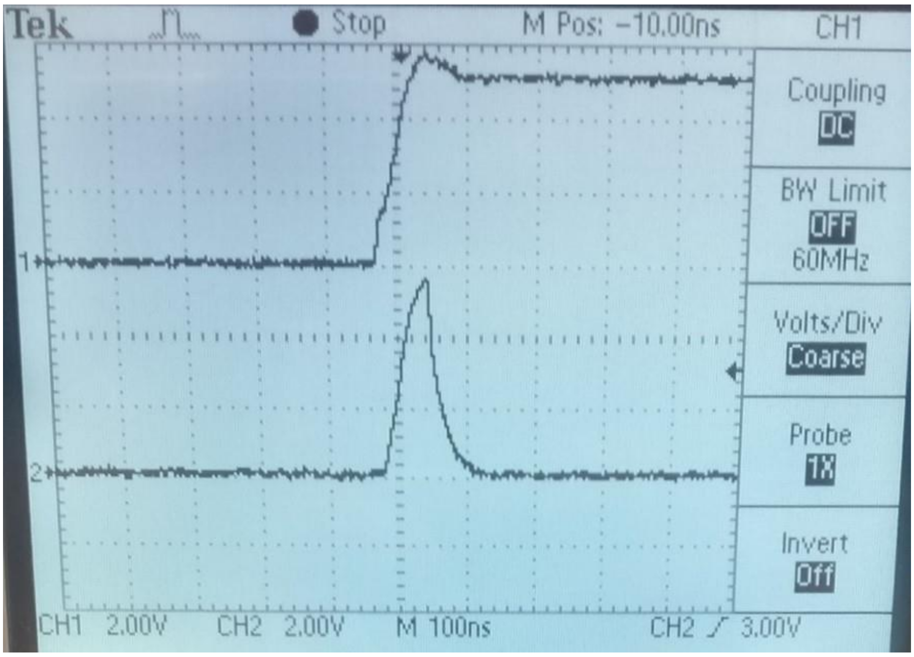


Figure 35. 100ηs Pulse Width (De Brabanter, 2017)

4.1.3 MOSFET and LED driver

The LED driver has a maximum current of 1.5A. Therefore, the R1 resistor is placed to limit the current. The monostable multivibrator however cannot drive this resistor since the output current is limited to 50mA. Therefore, a MOSFET driven by a 12V power source is placed which will wire through the pulses of the monostable multivibrator to the LED driver. The LED driver can deliver sufficient power for four high power LEDs by using an external power source of 70V. The circuit of the MOSFET and LED driver is seen in Figure 36.

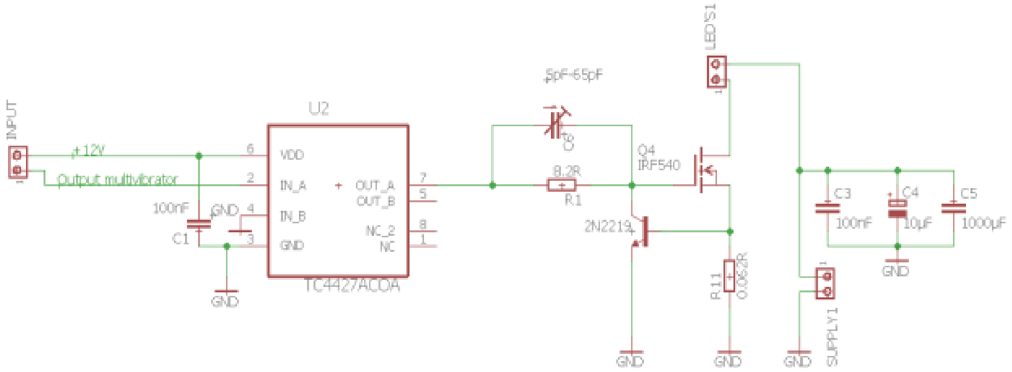


Figure 36. MOSFET and LED driver circuit (De Brabanter, 2017)

4.1.4 LED

Four high power white LEDs (LZ4-40CW08-0065) are used (Figure 37).

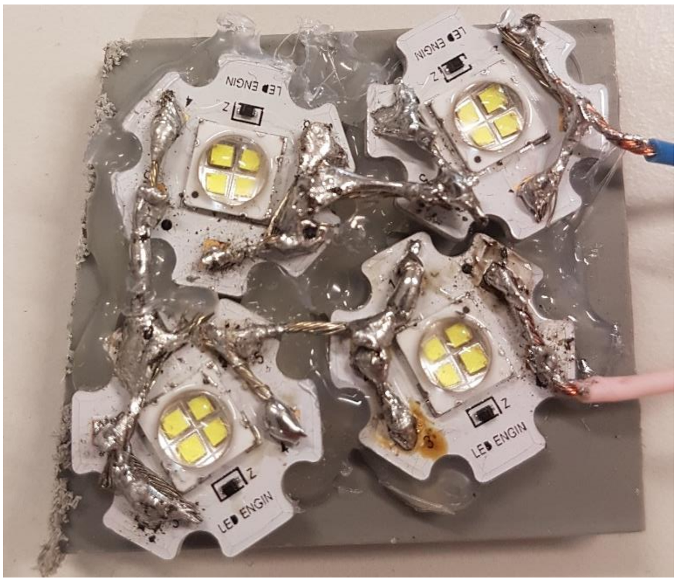


Figure 37. High-power LEDs

At a nominal current of 700mA a single LED will provide 680 lumens. Using four LEDs should thus offer sufficient brightness across the field of view to expose the sensor. If necessary the LEDs can also be overdriven. The pulse duration is defined by the monostable multivibrator.

4.2 Program Overview

Three programs are created. The first program contains the video recording settings. The video will be recorded at 180 frames per second at a user defined recording time. The code for this program is found in appendix 1. The second program is the video processing program. This program is used to process a pre-recorded recording. The code can be found in appendix 2-4 with each appendix containing one of the three different segmentation techniques tested. An overview is given in Figure 38.

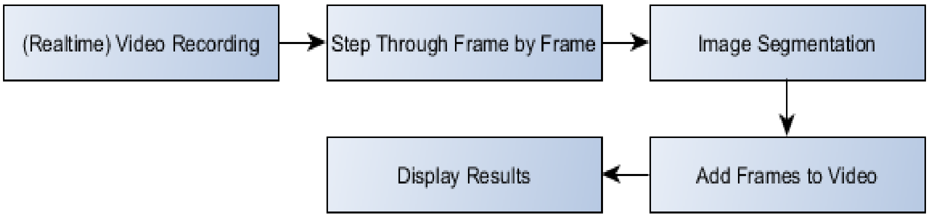


Figure 38. Video processing diagram

As can be seen the frame by frame processing is kept to a minimum. Since the goal is to research the possibility of identifying the passing by lint, processing must be kept to minimum to make real-time processing a possibility. 180 frames per second is still likely too much even with this minimal amount of processing. Therefore, this program is created three times with three different segmentation techniques to compare performances.

For extensive analysis to verify the theoretical conclusions such as the smallest detected feature a third program is created. The diagram of the program can be seen in Figure 39 and the code can be found in appendix 5.

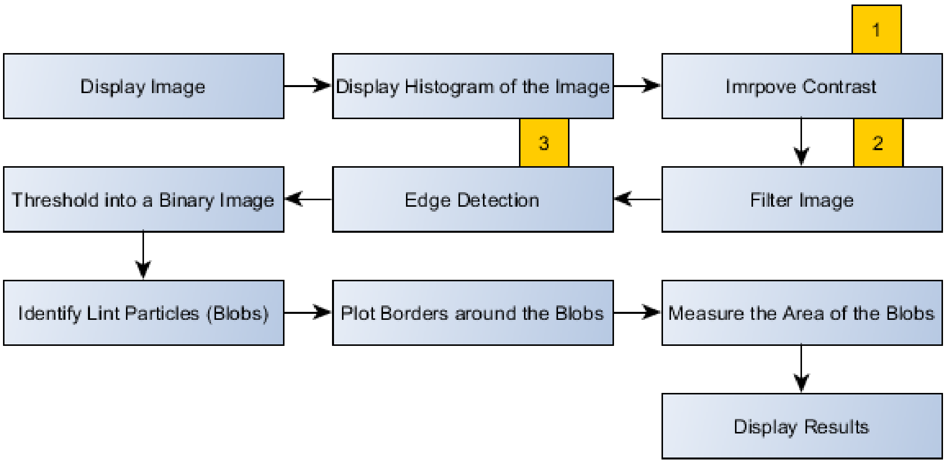

Display Image
Display Histogram of the Image
Imrpove Contrast
1
2
3
Threshold into a Binary Image
Edge Detection
Filter Image
Identify Lint Particles (Blobs)
Plot Borders around the Blobs
Measure the Area of the Blobs
Display Results

Figure 39. Extensive image analysis diagram

As can be seen in the diagram this program has more extensive processing. Processing blocks one, two and three are likely not necessary depended on the recording conditions. It should be noted that these extensive analyses functions can be easily implemented in the for loop that steps through the video frame by frame if more extensive analysis is required in the video processing program.

Unfortunately, some issues occurred with the strobe light which will be explained in the conclusion. The other available lighting system seen in Figure 40 is not optimal.

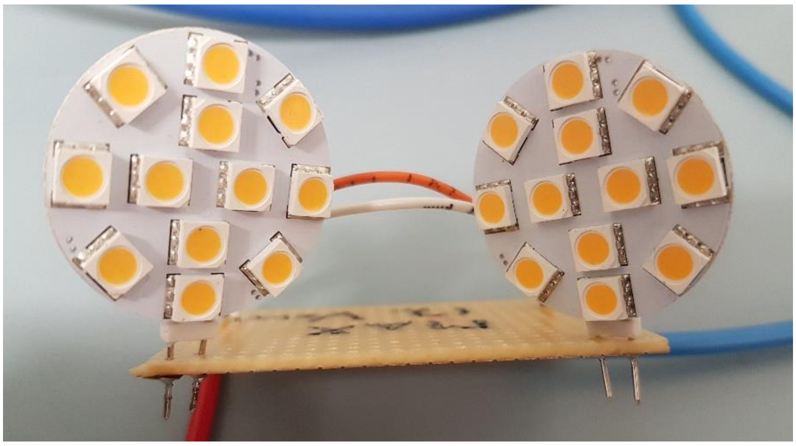


Figure 40. Alternative light source

The light source is angled in such a way that all light is focussed to the middle of the field of view. Therefore, light is not evenly distributed with brightness gradually decreasing away from the centre of the field of view. In image analysis, the darker portion of the field of view was cut to have a good histogram distribution. For the video recording this was not possible. Therefore, histogram segmentation results will be less accurate in the video recorded section (5.2) than in the image analysis section (5.3).

5 TEST SETUP RESULTS

5.1 Flash Synchronization

In Figure 41 the result of the monostable multivibrator being driven by the general purpose direct output of the camera can be seen.

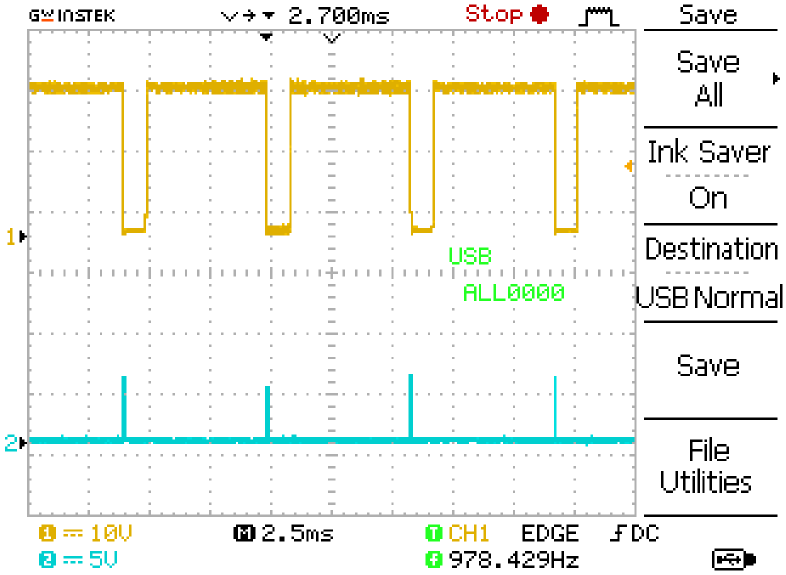


Figure 41. Camera and strobe light synchronization (De Brabanter, 2017)

The yellow signal on channel one is the output of the camera without inversion. The blue signal on channel two is the output of the monostable multivibrator. As can be seen, the output of the monostable multivibrator is successfully synchronized with the exposure active output of the camera at 180 frames per second. In this result, the output pulse is 250ηs as can be seen in Figure 42.

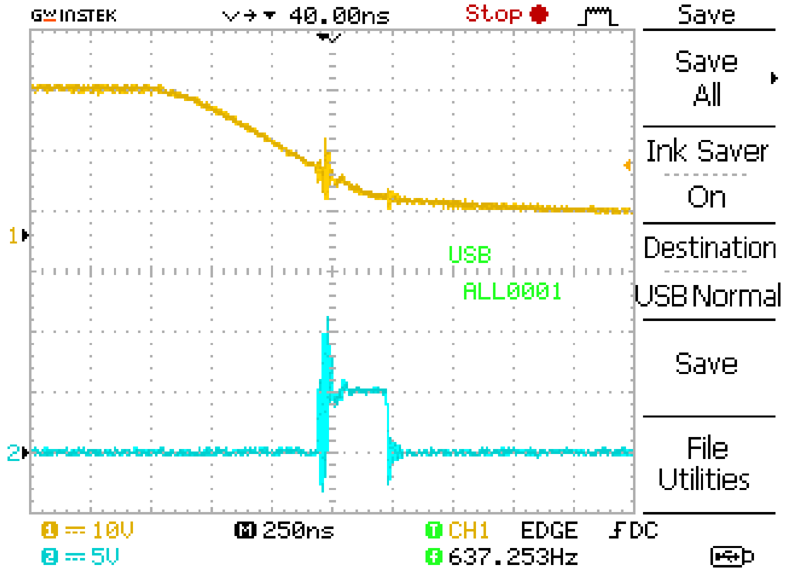


Figure 42. 250ηs output pulse (De Brabanter, 2017)

5.2 Pre-recorded Performance

To test the possibility of real-time recording the performance of three different segmentation techniques was analysed.

5.2.1 Gaussian Mixture Models

Figure 43 displays the output of the Gaussian mixture foreground detector.

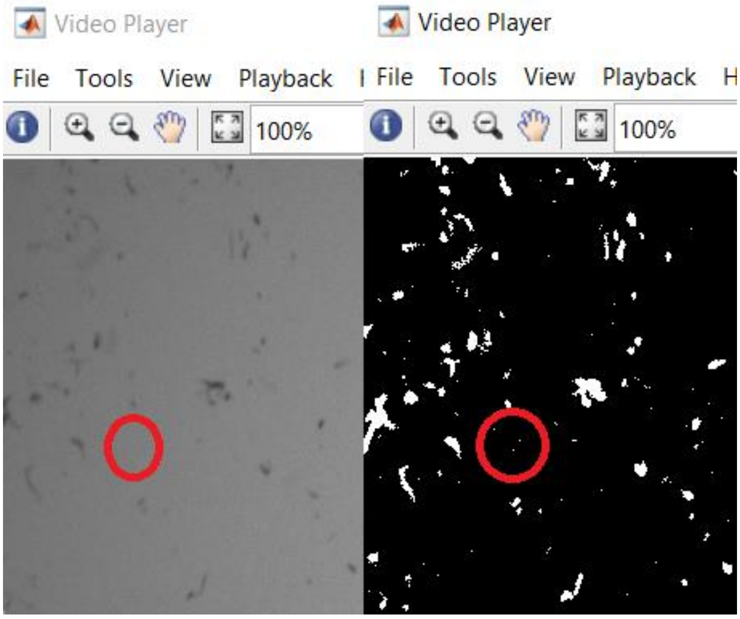


Figure 43. Gaussian mixture models result

The left image displays an unmodified frame of the original recording. The right image is the binary image that is the result of the Gaussian mixture model segmenting. As can be seen, false detections occur due to noise. Therefore, a separate filter is required.

An example of such a filter is to morphologically open the image with a disk of radius one. This will filter out all the noise with a radius smaller than one pixel. It will however also remove all the particles with a radius smaller than one, thus reducing accuracy. For size and mass measurements these particles are inaccurate due to the Nyquist theorem explained 2.5.1 so this loss of accuracy is less of a concern.

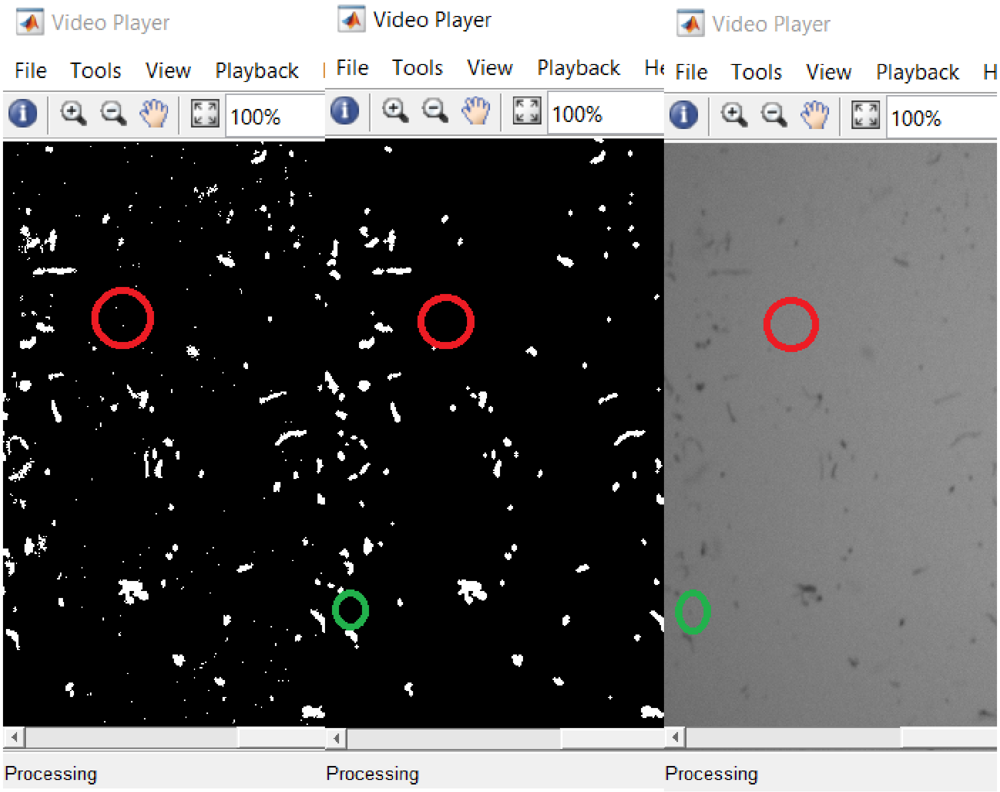
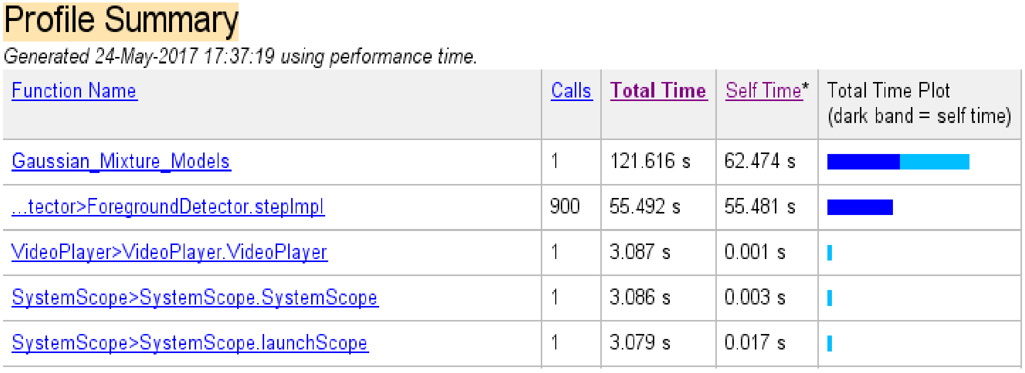


Figure 44. Gaussian mixture models with an additional noise filter

The used filter not only successfully removes all noise (for example, the region in red) it also fills the holes in the detected blobs (Figure 44). The number of particles removed due the filter (for example, the particle in green) is very minimal.

A performance analysis of the Gaussian mixture foreground detector is seen in Figure 45.



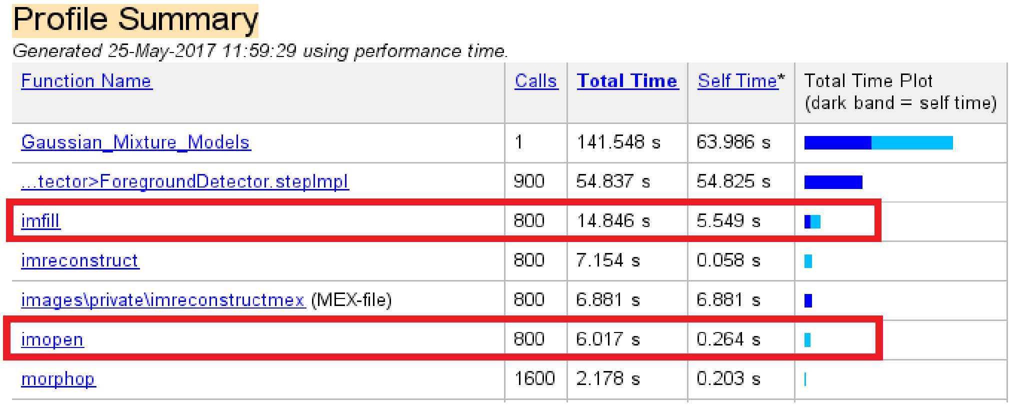
Profile Summary

Generated 24-May-2017 17:37:19 using performance time.

Function Name	Calls	Total Time	Self Time*	Total Time Plot (dark band = self time)
Gaussian_Mixture_Models	1	121.616 s	62.474 s	
...tector>ForegroundDetector.stepImpl	900	55.492 s	55.481 s	
VideoPlayer>VideoPlayer.VideoPlayer	1	3.087 s	0.001 s	
SystemScope>SystemScope.SystemScope	1	3.086 s	0.003 s	
SystemScope>SystemScope.launchScope	1	3.079 s	0.017 s	

Figure 45. Gaussian mixture models performance analysis

In comparison with the other two segmentation techniques, Gaussian mixture models requires significantly more processing time. Therefore, this segmentation technique is not ideal for real-time processing. In Figure 46 the filters are included in the performance analysis.



Profile Summary

Generated 25-May-2017 11:59:29 using performance time.

Function Name	Calls	Total Time	Self Time*	Total Time Plot (dark band = self time)
Gaussian_Mixture_Models	1	141.548 s	63.986 s	
...tector>ForegroundDetector.stepImpl	900	54.837 s	54.825 s	
imfill	800	14.846 s	5.549 s	
imreconstruct	800	7.154 s	0.058 s	
images\private\imreconstructmex (MEX-file)	800	6.881 s	6.881 s	
imopen	800	6.017 s	0.264 s	
morphop	1600	2.178 s	0.203 s	

Figure 46. Noise filter performance analysis

The filters have a further negative impact on the performance.

5.2.2 Histogram Binary Imaging

Figure 47 display the output of histogram binary segmenting using Otsu's method for automatic thresholding.

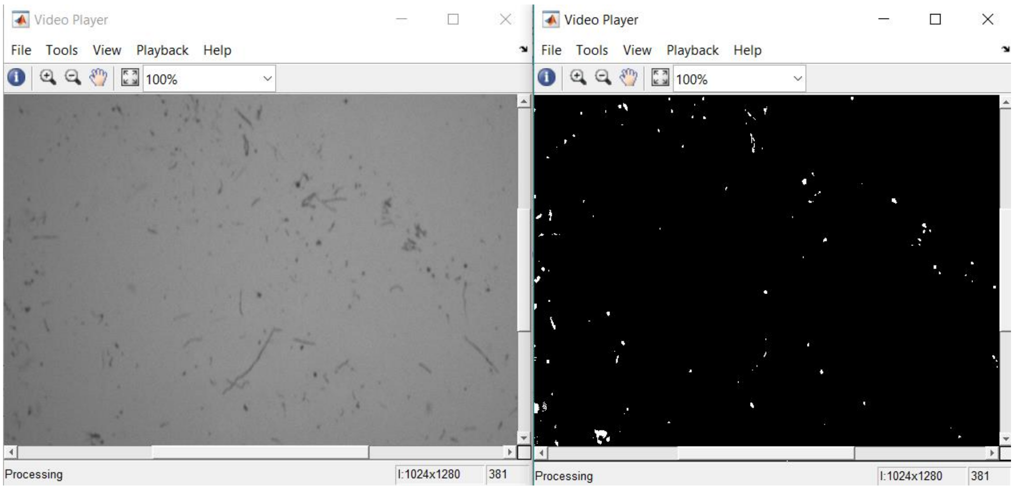


Figure 47. Segmentation using Otsu's thresholding result

The results are very poor with many objects not detected. This is due to the poor contrast of the original video frame as can be seen in the histogram of Figure 48.

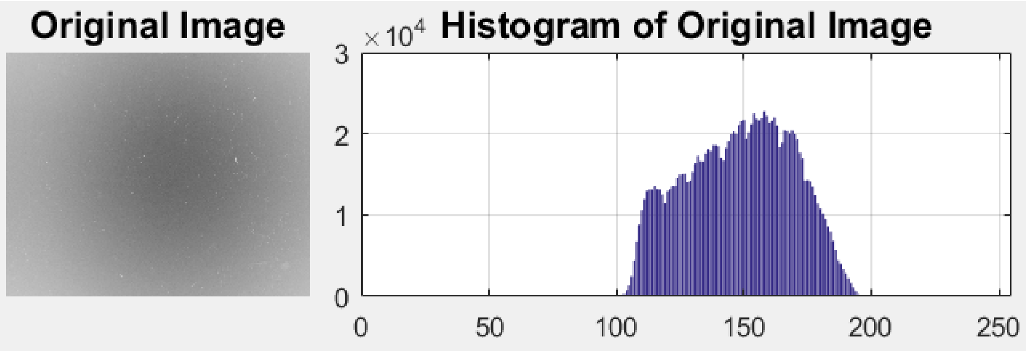
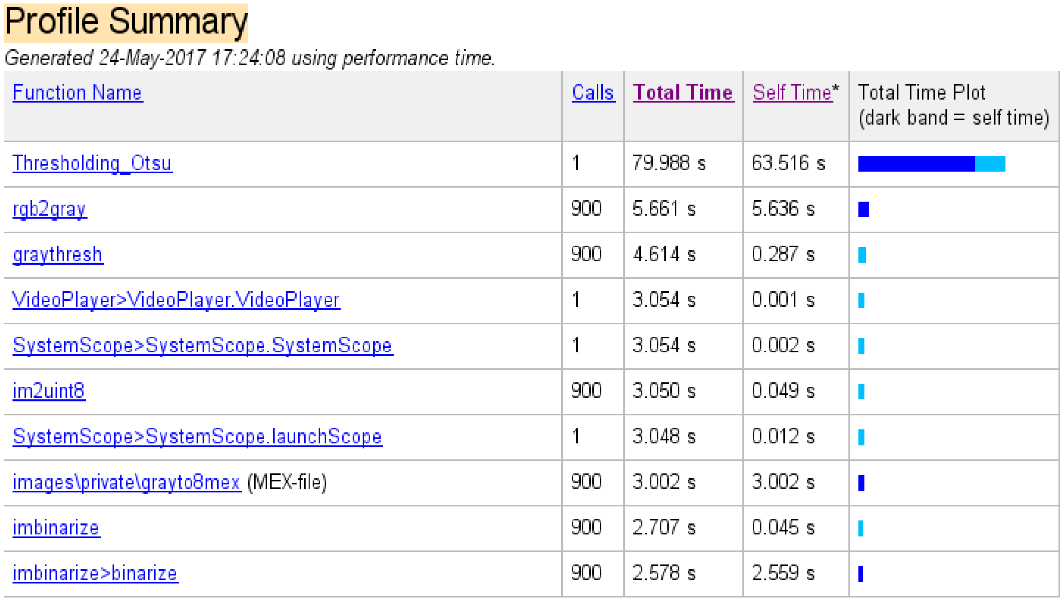


Figure 48. Poor contrast

The poor contrast is a result of poor light distribution of the alternative light source. Thresholding is still a viable method when sufficient contrast is possible as seen in 5.3.

A performance analysis of the segmentation method by using Otsu's thresholding is seen in Figure 49.



Profile Summary

Generated 24-May-2017 17:24:08 using performance time.

Function Name	Calls	Total Time	Self Time*	Total Time Plot (dark band = self time)
Thresholding_Otsu	1	79.988 s	63.516 s	▬▬▬▬
rgb2gray	900	5.661 s	5.636 s	▮
graythresh	900	4.614 s	0.287 s	▮
VideoPlayer>VideoPlayer.VideoPlayer	1	3.054 s	0.001 s	▮
SystemScope>SystemScope.SystemScope	1	3.054 s	0.002 s	▮
im2uint8	900	3.050 s	0.049 s	▮
SystemScope>SystemScope.launchScope	1	3.048 s	0.012 s	▮
images\private\grayto8mex (MEX-file)	900	3.002 s	3.002 s	▮
imbinarize	900	2.707 s	0.045 s	▮
imbinarize>binarize	900	2.578 s	2.559 s	▮

Figure 49. Segmentation using Otsu's thresholding performance analysis

5.2.3 Frame Differencing

Figure 50 displays the output of segmentation by using image subtraction.

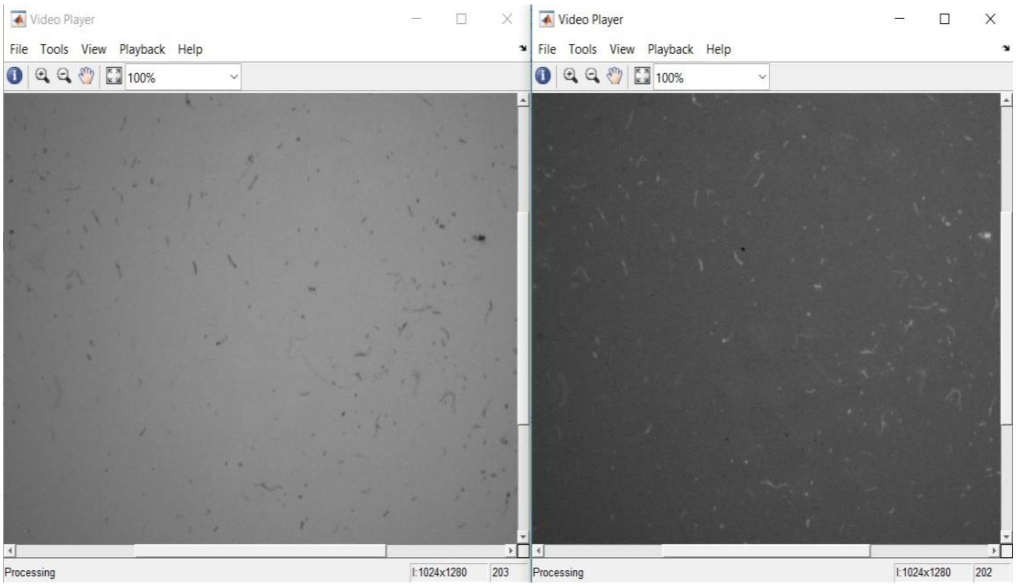
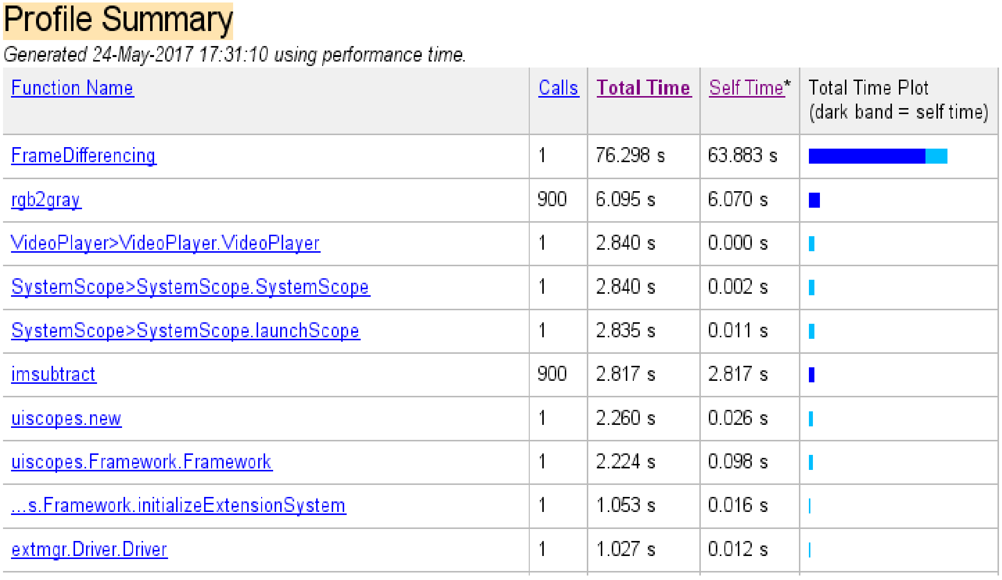


Figure 50. Image Subtraction Results

The results are very good even in non-optimal testing conditions. Though due to non-uniform lighting the result of the image subtraction is not a binary image. A separate step to binarize the image is necessary.

Figure 51 analyses the performance of the image subtraction segmentation.



Profile Summary

Generated 24-May-2017 17:31:10 using performance time.

Function Name	Calls	Total Time	Self Time*	Total Time Plot (dark band = self time)
FrameDifferencing	1	76.298 s	63.883 s	▬
rgb2gray	900	6.095 s	6.070 s	▪
VideoPlayer>VideoPlayer.VideoPlayer	1	2.840 s	0.000 s	▮
SystemScope>SystemScope.SystemScope	1	2.840 s	0.002 s	▮
SystemScope>SystemScope.launchScope	1	2.835 s	0.011 s	▮
imsubtract	900	2.817 s	2.817 s	▮
uiscopes.new	1	2.260 s	0.026 s	▮
uiscopes.Framework.Framework	1	2.224 s	0.098 s	▮
...s.Framework.initializeExtensionSystem	1	1.053 s	0.016 s	|
extmgr.Driver.Driver	1	1.027 s	0.012 s	|

Figure 51. Image Subtraction Performance Analysis

In comparison with the other two segmentation techniques, frame differencing requires significantly less processing time and is thus the best method for real-time processing.

5.3 Image Analysis

To detect the smallest detected particle the image analysis program works as follows (Figure 52).

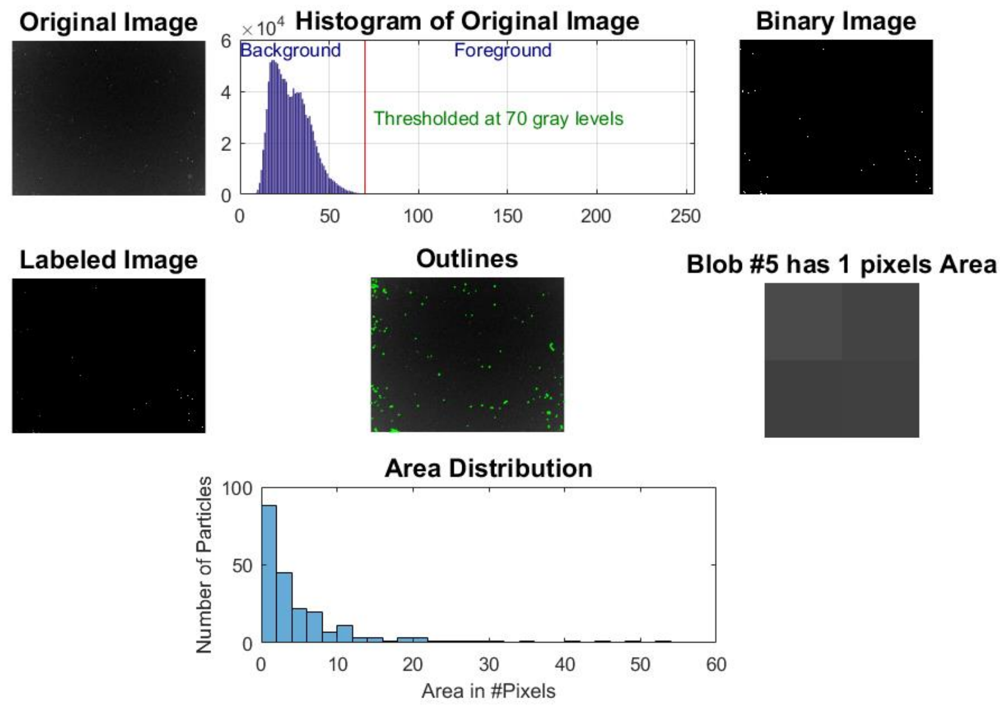


Figure 52. Image analysis process

Firstly, the original image is segmented to a binary image using a manual threshold selected based on the histogram of the original image. Afterwards the particles are identified, the properties of the particles are calculated and they are individually labelled. Then an outline is drawn around the identified particles. Lastly, the size of the particles is displayed and a histogram is created to show the particle distribution.

5.3.1 Smallest Detected Feature

The smallest detected particle has an area of one pixel (Figure 53).

Blob #7 has 1 pixels Area

Figure 53. Smallest Detected Particle

The pixel size of the camera is 4,8 microns. It would be wrong to conclude that the smallest detected particles are 4,8 microns since it does not take in factors such as working distance and magnification. To find the size of one pixel, an image was taken of 5mm squares (Figure 54).

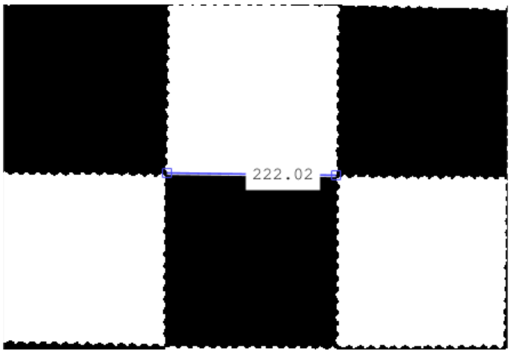
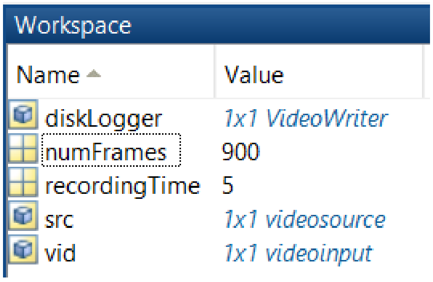


Figure 54. Calculating Pixel Size

The 5mm square is 222 pixels wide. Thus, one pixel is about 23 microns. The previous calculations of 42 microns which factor in the Nyquist theorem is therefore quite accurate. Of course, particles could be detected even smaller than 23 microns as stated in 2.5.1. Though size measurements on features smaller than 46 microns are inaccurate.

5.4 Frames per Second

The camera is successful in taking 180 frames per second (Figure 55).



Workspace

Name	Value
diskLogger	*1x1 VideoWriter*
numFrames	900
recordingTime	5
src	*1x1 videosource*
vid	*1x1 videoinput*

Figure 55. Framerate

As can be seen, a recording of five seconds captures 900 frames, thus 180 frames per second, enough for a full representation of the lint passing by.

6 CONCLUSION

As the theoretical calculations confirmed the available hardware is not capable of detecting particles of 10 microns. However, the results are still positive and particles with sizes as small as 40 microns can be detected accurately while fully representing all the lint passing by at 180 frames per second. The practical tests verified these calculations and confirmed that when neglecting the Nyquist theorem even particles smaller than 20 microns, which is one pixel, can be detected. These detections however do not adhere to the Nyquist theorem and are thus not accurate for size measurements.

Synchronization between the industrial area scan camera and the strobe light was successful. Using a short duration strobe light is thus a capable method to prevent motion blur in setups using an industrial camera with a long exposure time. Unfortunately, due to issues with the LEDs it could not be tested if the strobe duration is sufficient to expose the sensor.

As expected real time processing of the passing by lint is not possible at 180 frames per second in MATLAB. This was however not the goal of the thesis. This thesis was focussed on research, for implementation the programs should be further optimized and programmed in C++ for better performance.

The segmentation technique with the best performance, both in accuracy and processing time, is frame differencing. Automated thresholding by using Otsu’s method suffered the most from the poor contrast of the test-setup. Under more ideal recording conditions this method would still be a viable alternative especially under conditions where the background is not static. Gaussian mixture models require too much processing time for segmentation at 180 frames per second.

REFERENCES

Amiri, R., Bégin, B., Deshaies, S., & Mozaffari, S. (2004). Effects of wood and pulp quality 2003. *Pulp & Paper Canada*, *105*(6), 129–135.

Andor. (n.d.). CCD Spatial Resolution - Understanding spatial resolution. Retrieved May 23, 2017, from http://www.andor.com/learning-academy/ccd-spatial-resolution-understanding-spatial-resolution

Argyle, E., & Rosenfeld, A. (1971). Techniques for Edge Detection. *Proceedings of the IEEE*, *59*(2), 285–287. https://doi.org/10.1109/PROC.1971.8136

Basler. (n.d.). Basler acA1920-25um EMVA. Retrieved from https://www.baslerweb.com/fp-1489067453/media/downloads/documents/emva_data/BD00099201_Basler_acA1920-25um_EMVA_Standard_1288.pdf

Basler. (2016). Basler Lens C125-0818-5M, *0*(2000034830). Retrieved from https://www.baslerweb.com/fp-1491830718/media/downloads/documents/users_manuals/DG00144103_Tech_Spec_for_SAP_2000034832.pdf

Basler. (2017). User's Manual for USB 3.0 Cameras. Retrieved from http://s.baslerweb.com/media/documents/AW00123410000_ace_USB_3_0_Users_Manual.pdf

Batchelor, B. G. (2012). *Machine Vision Handbook*. (B. G. Batchelor, Ed.) (Vol. 2). Springer.

Blankinship, R. (2005). Understanding fundamentals of imaging lenses - Vision Systems Design. Retrieved March 20, 2017, from http://www.vision-systems.com/articles/print/volume-10/issue-1/worldwide-camera-lens-directory/understanding-fundamentals-of-imaging-lenses.html

Brouillette, F., Morneau, D., Chabot, B., & Daneault, C. (2006). Evaluation of new lint reduction additives in wood-containing paper manufacturing. *Pulp and Paper Canada*, *107*(2), 47–50.

Corke, P. (2011). *Robotics , Vision and Control*. (B. Siciliano & O. Khatib, Eds.) (Volume 73). Springer.

DALSA. (n.d.). Application Note Line Scan Imaging Basics - TDI, 11. Retrieved from https://www.teledynedalsa.com/public/mv/appnotes/00541-00_03-32_Linescan_Imaging_Basics.pdf

Daryl, M. (2017). A Practical Guide to Machine Vision Lighting - National Instruments. Retrieved March 20, 2017, from http://www.ni.com/white-paper/6901/en/

De Brabanter, A. (2017). *High-Speed Camera Flash*. Häme University of Applied Sciences.

Edmund Optics. (n.d.-a). The Advantages of Telecentricity. Retrieved March 20, 2017, from https://www.edmundoptics.com/resources/application-notes/imaging/advantages-of-telecentricity/

Edmund Optics. (n.d.-b). Understanding Focal Length and Field of View. Retrieved March 20, 2017, from https://www.edmundoptics.com/resources/application-notes/imaging/understanding-focal-length-and-field-of-view/

Fleyeh, H. (2014). Best for Image Processing - Matlab or OpenCV ? Retrieved March 25, 2017, from https://www.researchgate.net/post/Best_for_Image_Processing-Matlab_or_OpenCV

Huang, Z.-K., & Chau, K.-W. (2008). A new image thresholding method based on Gaussian mixture model. *Applied Mathematics and Computation*, *205*(2), 899–907. https://doi.org/10.1016/j.amc.2008.05.130

J. M. Prewitt and M. L. Mendelsohn. (1966). The analysis of cell images. *Annals of the New York Academy of Sciences*, *128*(3), 1035–1053.

Kaewtrakulpong, P., & Bowden, R. (2001). An Improved Adaptive Background Mixture Model for Real- time Tracking with Shadow Detection 2 Background Modelling. *In Proc. 2nd European Workshop on Advanced Video Based Surveillance Systems, Computer Vision and Distributed Processing, Kluwer Academic Publishers*, *AVBS01*, 1–5. https://doi.org/10.1.1.12.3705

Lee, D., & Eddins, S. (2003). Tracking Objects: Acquiring and Analyzing Image Sequences in MATLAB - MATLAB &amp; Simulink. Retrieved May 1, 2017, from https://nl.mathworks.com/company/newsletters/articles/tracking-objects-acquiring-and-analyzing-image-sequences-in-matlab.html

Lestiani, R., Batchelor, W., & Banham, P. (2013). Investigation of lint particle adhesion in offset printing using Weibull statistics. *Journal of Adhesion Science and Technology*, *27*(4), 339–353. https://doi.org/10.1080/01694243.2012.705535

Lestiani, R., Batchelor, W., & Banham, P. (2014). Effect of paper and printing press variables on the rates of adhesion failure in the linting of offset printing. *Journal of Adhesion Science and Technology*, *28*(19), 1935–1948. https://doi.org/10.1080/01694243.2014.929517

Maini, R., & Aggarwal, H. (2009). Study and Comparison of Various Image Edge Detection Techniques. *International Journal of Image Processing (IJIP)*, *3*(1).

Mars, P. (2009). Comparison of xenon and LED Flash for camera phones. Comparison of xenon flash and high current LEDs for photo flash in camera phones – a review and update. Retrieved from http://www.cap-xx.com/wp-content/uploads/2015/04/CAP-XX_WP_0906_Comparison_of_xenon_flash_and_LED_flash_v3.pdf

National Instruments. (2014). *Calculating Camera Sensor Resolution and Lens Focal Length - National Instruments*. Retrieved from http://digital.ni.com/public.nsf/allkb/1BD65CB07933DE0186258087006FEBEA

Nguyen, D. (2015). *Paper Dust Classification and Measurement Using Machine Vision*. Häme University of Applied Sciences.

OpenCV. (n.d.). OpenCV: Background Subtraction. Retrieved May 1, 2017, from http://docs.opencv.org/3.1.0/db/d5c/tutorial_py_bg_subtraction.html

Otsu, N. (1979). A Threshold Selection Method from Gray-Level Histograms, *20*(1), 62–66. https://doi.org/10.1109/TSMC.1979.4310076

Pau, L. F., & Olafsson, R. (1991). *Fish quality control by computer vision*. M Dekker. Retrieved from https://www.crcpress.com/Fish-Quality-Control-by-Computer-Vision/Pau/p/book/9780824784263

Psarossy. (2013). camera basics - Is there a formula to calculate DOF? - Photography Stack Exchange. Retrieved May 27, 2017, from https://photo.stackexchange.com/questions/24826/is-there-a-formula-to-calculate-dof

Qimaging. (n.d.). Rolling Shutter vs. Global Shutter. Retrieved March 1, 2017, from https://www.qimaging.com/ccdorscmos/triggering.php

Sally Wiener, G., & Daniel, G. (2012). Not all pixels are created equal: the size of a camera's sensor, not the pixel count, determines the quality of a photograph. *IEEE Spectrum*, *49*(5), 22–24.

Stemmer Imaging. (n.d.). Region of interest (ROI). Retrieved April 30, 2017, from https://www.stemmer-imaging.co.uk/en/knowledge-base/region-of-interest-roi/

University of California; Berkeley. (n.d.). Capturing Images. Retrieved March 20, 2017, from http://microscopy.berkeley.edu/courses/dib/sections/02Images/sampling.html

Vorenkamp, T. (2016). Understanding Crop Factor. Retrieved April 4, 2017, from https://www.bhphotovideo.com/explora/photography/tips-and-solutions/understanding-crop-factor

Watanabe, S., & Group, C. (1974). An Automated Apparatus for Cancer Prescreening: CYBEST, 350–358.

Weszka, J. S., Nagel, R. N., & Rosenfeld, A. (1974). A Threshold Selection Technique. *IEEE Transactions on Computers*, *C-23*(12), 1322–1326. https://doi.org/10.1109/T-C.1974.223858

Willert, C., Stasicki, B., Klinner, J., & Moessner, S. (2010). Pulsed operation of high-power light emitting diodes for imaging flow velocimetry. *Measurement Science and Technology*, *21*(7), 75402. https://doi.org/10.1088/0957-0233/21/7/075402

Wilson, S., Gustafson, G., Lincoln, D., Murari, K., & Johansen, C. (2014). Performance evaluation of an overdriven LED for high-speed schlieren imaging. *Journal of Visualization*, *18*(1), 35–45. https://doi.org/10.1007/s12650-014-0220-7

Wloka, M. M., & Zeleznik, R. C. (1996). Interactive Real-Time Motion Blur. *Science And Technology*.

Recording Settings

Clear workspace

```
clear all;
close all
clc;
```

Create the video input object

```
vid = videoinput('gentl', 1, 'Mono8');
src = getselectedsource(vid);
```

Implement the various recording settings

```
recordingTime = 5; % Select the recording durations
numFrames = 180 * recordingTime; % The amount of frames that must be recorded is equal to the framerate * the recording time

vid.FramesPerTrigger = numFrames; % When the recording is triggered it records the amount of frames

% Write the recording to the disk
vid.LoggingMode = 'disk';

diskLogger = VideoWriter('C:\Users\benja\Pictures\MATLAB\Recording_0001.avi', 'Grayscale AVI');

vid.DiskLogger = diskLogger;

diskLogger.FrameRate = 180; % Recorded at 180 frames per second
```

Select the output line for flash synchronization

One setting is missing in MATLAB (selection for when the output becomes high). Therefore, it might be necessary to first use Basler's Pylon for configuring the output lines.

```
%src.SensorReadoutMode = 'Fast'; $ Not necessary for 180 frames per second, use if higher framerate is required (200+).
src.LineSelector = 'Line3';
src.LineMode = 'Output';
```

Start recording

```
start(vid)
```

Published with MATLAB® R2016b

Appendix 2

Gaussian Mixture Models

Clear workspace

```
clear all;
close all
clc;
```

Read video

```
videoReader = vision.VideoFileReader('Recording_0001.avi');
```

Create video players

```
videoPlayer = vision.VideoPlayer;
foregroundPlayer = vision.VideoPlayer;
originalPlayer = vision.VideoPlayer;
```

Background subtraction

Background subtraction by using Gaussian mixture models (computer vision toolbox is required)

```
foregroundDetector = vision.ForegroundDetector('NumGaussians', 2,'NumTrainingFrames',
100);

%First 100 frames are used to learn the background
for i = 1:100
    videoFrame = step(videoReader);
    foreground = step(foregroundDetector, videoFrame);
end
```

Loop through video

```
while ~isDone(videoReader)
    videoFrame = step(videoReader); % read the next video frame

    % Detect the foreground in the current video frame
    foreground = step(foregroundDetector, videoFrame);

    % Noise filter: ideal settings can be found by using the Image Morphology MATLAB
application
    cleanForeground = imopen(foreground, strel('Disk',1));
    cleanForeground = imfill(cleanForeground, 'holes');

    % Show original video, foreground after segmentation and foreground after filter
    step(originalPlayer, videoFrame);
    step(videoPlayer, foreground);
    step(foregroundPlayer, cleanForeground);
end
```

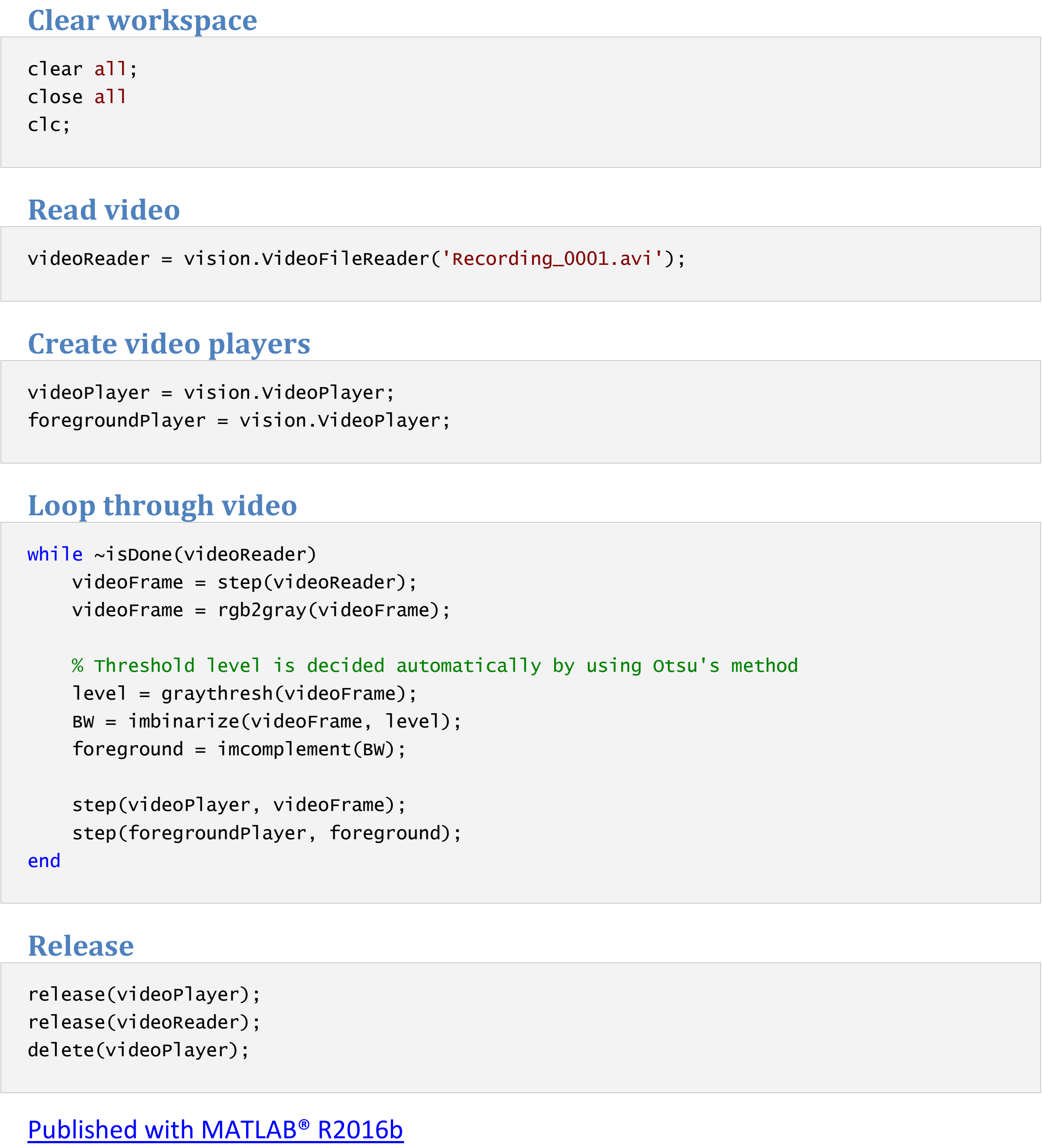
Release

```
release(videoPlayer);
release(videoReader);
delete(videoPlayer);
```

Published with MATLAB® R2016b

Appendix 3

Automatic Thresholding by Using Otsu’s Method



Clear workspace

```
clear all;
close all
clc;
```

Read video

```
videoReader = vision.VideoFileReader('Recording_0001.avi');
```

Create video players

```
videoPlayer = vision.VideoPlayer;
foregroundPlayer = vision.VideoPlayer;
```

Loop through video

```
while ~isDone(videoReader)
    videoFrame = step(videoReader);
    videoFrame = rgb2gray(videoFrame);

    % Threshold level is decided automatically by using Otsu's method
    level = graythresh(videoFrame);
    BW = imbinarize(videoFrame, level);
    foreground = imcomplement(BW);

    step(videoPlayer, videoFrame);
    step(foregroundPlayer, foreground);
end
```

Release

```
release(videoPlayer);
release(videoReader);
delete(videoPlayer);
```

Published with MATLAB® R2016b

Appendix 4

Frame Differencing

Clear workspace

```
clear all;
close all
clc;
```

Read video

```
videoReader = vision.VideoFileReader('Recording_0001.avi');
```

Create video players

```
videoPlayer = vision.VideoPlayer;
foregroundPlayer = vision.VideoPlayer;
```

Loop through video

```
load('background.mat')
background = im2single(background);
while ~isDone(videoReader)
    videoFrame = step(videoReader);
    videoFrame = rgb2gray(videoFrame);

    foreground = imsubtract(background, videoFrame);

    step(videoPlayer, videoFrame);
    step(foregroundPlayer, foreground);
end
```

Release

```
release(videoPlayer);
release(videoReader);
delete(videoPlayer);
```

Published with MATLAB® R2016b

Appendix 5

Image Analysis

Start-up

```
tic; % Start timer
clc; % Clear the command window
clearvars; % Clear variables from prior runs
fprintf('Running Lint Analysis'); % Message to command window
imtool close all; % Close all figures
captionFontSize = 14; % Font for figure titles
```

Check if image processing toolbox is installed

```
hasIPT = license('test', 'image_toolbox');
if ~hasIPT
              % User does not have the toolbox installed.
              message = sprintf('Sorry, but you do not seem to have the Image Processing
Toolbox.\nDo you want to try to continue anyway?');
              reply = questdlg(message, 'Toolbox missing', 'Yes', 'No', 'Yes');
              if strcmpi(reply, 'No')
                            % User said No, so exit.
                            return;
              end
end
```

Import the image with the lint particles

```
originalImage = imread('Backlight.png');
%originalImage = rgb2gray(originalImage); % Convert to grayscale if the input is in rgb
originalImage = imcomplement(originalImage); % Complement the image (white = foreground
lint, black = background)
subplot(4,3,1); % Display Image
imshow(originalImage);
set(gcf, 'units', 'normalized', 'outerposition', [0 0 1 1]); % Maximize figure window
drawnow; % Force to display image right away
title('Original Image', 'FontSize', captionFontSize);
axis image; % Don't stretch to screens aspect ratio
```

Display histogram of original image

```
[pixelCount, grayLevels] = imhist(originalImage);
subplot(4,3,2);
bar(pixelCount);
title('Histogram of Original Image','FontSize', captionFontSize);
xlim([0 grayLevels(end)]); %Scale x axis manually
grid on;
```

Thresholding into a binary image (0 = dark, 1 = bright) by analysing the histogram

Threshold is manually selected by analysing the histogram and performing trial and error runs. An automatic threshold using Otsu's method is possible, though a manual threshold is more accurate.

```
thresholdValue = 70;
binaryImage = originalImage > thresholdValue;
binaryImage = imfill(binaryImage, 'holes'); % Fill holes inside the blobs

% Show the used threshold as a red line in the histogram
hold on;
maxYValue = ylim;
line([thresholdValue, thresholdValue], maxYValue, 'Color', 'r');

% Place a text label on the bar chart showing the threshold.
annotationText = sprintf('Thresholded at %d gray levels', thresholdValue);
% For text(), the x and y need to be of the data class "double" so cast both to double.
text(double(thresholdValue + 5), double(0.5 * maxYValue(2)), annotationText, 'FontSize',
10, 'Color', [0 .5 0]);
text(double(thresholdValue - 70), double(0.94 * maxYValue(2)), 'Background', 'FontSize',
10, 'Color', [0 0 .5]);
text(double(thresholdValue + 50), double(0.94 * maxYValue(2)), 'Foreground', 'FontSize',
10, 'Color', [0 0 .5]);

% Display the binary image
subplot(4,3,3);
imshow(binaryImage);
title('Binary Image','FontSize', captionFontSize);
```

Identify individual blobs (lint particles)

```
labeledImage = bwlabel(binaryImage, 4); % Label connected components in a 2D binary
image
subplot(4,3,4);
imshow(labeledImage, []);
title('Labeled Image','FontSize', captionFontSize);

% Get the properties of the blobs
blobMeasurements = regionprops(labeledImage, originalImage, 'all');
numberOfBlobs = size(blobMeasurements, 1);
```

Plot borders of the lint on the original grayscale image using coordinates returned by bwboundaries

```
subplot(4,3,5);
imshow(originalImage);
title('Outlines','FontSize', captionFontSize);
axis image;

hold on;
boundaries = bwboundaries(binaryImage);
numberOfBoundaries = size(boundaries, 1);

for k=1 : numberOfBoundaries
    thisBoundary = boundaries{k};
    plot(thisBoundary(:,2), thisBoundary(:,1), 'g', 'LineWidth', 1);
end
hold off;
```

Measure the pixels in a blob

Since the measurements are plotted the number of blobs plotted is limited by the number of figures that can be plotted in the subplot.

```
for k = 5 : 11
   thisBlobsBoundingBox = blobMeasurements(k).BoundingBox; % Pixels in current blob
   subImage = imcrop(originalImage, thisBlobsBoundingBox);
   subplot(4,3,k+1);
   imshow(subImage);
   caption = sprintf('Blob #%d has %d pixels Area', ...
                                         k, blobMeasurements(k).Area);
   title(caption, 'FontSize', captionFontSize);
end
```

Plot the distribution of particle area in relation with the number of detected particles

```
particleArea = zeros(numberOfBoundaries,1); % Array containing the area distributions

for i = 1 : numberOfBoundaries
    particleArea(i) = blobMeasurements(i).Area; % X-axis of histogram (area distribution
of all the lint particles)
end

xbins = 0:2:60; % Histogram consists of blocks of two pixel area's and 60 blocks (so
untill 120 pixels area)
histogram(particleArea, xbins);
xlabel('Area in #Pixels')
ylabel('Number of Particles')
elapsedTime = toc; % Elapsed time of running this image analysis program
```

Published with MATLAB® R2016b