



BRNO UNIVERSITY OF TECHNOLOGY

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH TECHNOLOGIÍ

DEPARTMENT OF CONTROL AND INSTRUMENTATION

ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

DEFECT DETECTION ON FIBER MATERIALS USING MACHINE LEARNING

DETEKCE VAD VLÁKNITÉHO MATERIÁLU UŽITÍM METOD STROJOVÉHO UČENÍ

MASTER'S THESIS

DIPLOMOVÁ PRÁCE

AUTHOR

AUTOR PRÁCE

Bc. Matěj Lang

SUPERVISOR

VEDOUCÍ PRÁCE

Ing. Peter Honec, Ph.D.

BRNO 2019

Master's Thesis

Master's study field **Cybernetics, Control and Measurements**
Department of Control and Instrumentation

Student: Bc. Matěj Lang

ID: 174224

Year of study: 2

Academic year: 2018/19

TITLE OF THESIS:

Defect detection on fiber materials using machine learning

INSTRUCTION:

The goal is to propose and build automated station for defect detection in fibered material.

1. Design a scanning bench with XY translation.
2. Test and propose suitable positioning of camera(s), lights, usage of filters.
3. Research currently used machine learning methods in image processing.
4. Build scanning bench station.
5. Test and propose suitable method for defect detection.
6. Choose applicable programming language.
7. Create application for grabbing and processing images.
8. Learn the algorithm on the dataset.
9. Calibrate and evaluate the system.
10. Create final evaluation.

REFERENCE:

SZELISKI R.: Computer Vision, Algorithms and Applications, ISBN 978-1848829343

T. Abbood Wisam, M. Nancy Somer: Automated Surface Defect Detection Using Line Scan & Area Scan Cameras, ISBN 978-3659709388

HLAVAC V., SONKA M., BOYLE R.: Image Processing, Analysis, and Machine Vision, ISBN 978-0495082521

Assignment deadline: 4. 2. 2019

Submission deadline: 13. 5. 2019

Head of thesis: Ing. Peter Honec, Ph.D.

doc. Ing. Václav Jirsík, CSc.
Subject Council chairman

WARNING:

The author of this Master's Thesis claims that by creating this thesis he/she did not infringe the rights of third persons and the personal and/or property rights of third persons were not subjected to derogatory treatment. The author is fully aware of the legal consequences of an infringement of provisions as per Section 11 and following of Act No 121/2000 Coll. on copyright and rights related to copyright and on amendments to some other laws (the Copyright Act) in the wording of subsequent directives including the possible criminal consequences as resulting from provisions of Part 2, Chapter VI, Article 4 of Criminal Code 40/2009 Coll.

Abstrakt

Cílem této diplomové práce je automatizace detekce vad ve vláknitých materiálech.

Firma SILON se již přes padesát let zabývá výrobou jemné vaty z recyklovaných PET lahví. Tato vata se následně používá ve stavebnictví, automobilovém průmyslu, ale nejčastěji v dámských hygienických potřebách a dětských plenách. Cílem firmy je produkovat co nejkvalitnější výrobek a proto je každá dávka testována v laboratoři s několika přísnými kritérii. Jedním z testů je i množství vadných vláken, jako jsou zacuchané smotky vláken, nebo nevydloužená vlákna, která jsou tvrdá a snadno se lámou.

Navrhovaný systém sestává ze snímací lavice fungující jako scanner, která nasnímá vzorek vláken, který byl vložen mezi dvě skleněné desky. Byla provedena série testů s různým osvětlením, která ověřovala vlastnosti Rhodaminu, který se používá právě na rozlišení defektů od ostatních vláken. Tyto defekty mají zpravidla jinou molekulární strukturu, na kterou se barvivo chytá lépe. Protože je Rhodamin fluorescenční barvivo, je možné ho například pod UV světlem snáze rozeznat. Tento postup je využíván při manuální detekci. Při snímání kamerou je možno si vypomoci filtrem na kameře, který odfiltruje excitační světlo a propustí pouze světlo vyzářené Rhodaminem.

Součástí výroby skeneru byla i tvorba ovládacího programu. Byla vytvořena vlastní knihovna pro ovládání motoru a byla upravena knihovna pro kameru. Oba systém pak bylo možno ovládat pomocí jednotného GUI, které zajišťovalo pořizování snímku celé desky.

Pomocí skeneru byla nasnímána řada snímků, které bylo třeba anotovat, aby bylo možné naučit počítač rozlišovat defekty. Anotace proběhla na pixelové úrovni; každý defekt byl označen v grafickém editoru ve speciální vrstvě.

Pro rozlišování byla použita umělá neuronová síť, která funguje na principu konvolucí. Tento typ sítě je navíc plně konvoluční, takže výstupem sítě je obraz, který by měl označit na tom původním vadné pixely.

Výsledky naučené sítě jsou v práci prezentovány a diskutovány. Síť byla schopna se naučit rozeznávat většinu defektů a spolehlivě je umí rozeznat a segmentovat. Potíže má v současné době s detekcí rozmazaných defektů na krajích zorného pole a s defekty, jejichž hranice není tolik zřetelná na vstupních obrazech.

Nutno zmínit, že zákazník má zájem o kompletní řešení scanneru i s detekčním softwarem a vývoj tohoto zařízení bude pokračovat i po závěru této diplomové práce.

Klíčová slova

Vláknitý materiál, Rhodamine, scanner, řádková kamera, fluorescence, transmise, umělá neuronová síť, konvoluční neuronová síť, plně konvoluční neuronová síť, strojové učení, rozpoznávání.

Abstract

Following Master's Thesis is presenting the creation of scanning unit, that will automate the quality check process in company SILON, for their fibered material.

The process of manual detection is discussed and the automated solution is proposed. Several test are shown, that demonstrate effects of different lights on fibres dyed in Rhodamine. Optimal filter for camera is chosen, to achieve images with highest resolution possible and with enough definition.

Next, the software tools for hardware control are presented and tools for building neural networks. Also, some basic info on current state of the art is provided, to explain some of the tools used. The network itself is shown and also its learning process and capabilities of defect detection.

Keywords

Fibre material, Rhodamine, scanner, linescan camera, fluorescence, transmission, artificial neural network, ANN, convolutional neural network, CNN, fully convolutional network, FCN, machine learning, image recognition.

LANG, M. Defect detection on fibre materials using machine learning. Brno: Brno University of Technology, Faculty of Electrical Engineering and Communication, 2019. 46 s. Supervisor Ing. Peter Honec, Ph.D.

Prohlášení

Prohlašuji, že svou semestrální práci na téma „Defect detection on fibre materials using machine learning“ jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce. Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Vysoké Učení Technické, Brno _____

Acknowledgement

I would like to thank my supervisor, Peter Honec and to Pavel Číp, for useful insight and consultation.

Brno University of Technology _____

Contents

Introduction	11
1 Application	12
1.1 Current process	13
1.2 Proposed improvement	14
2 Frame & Drive	17
2.1 Frame	17
2.2 Linear Motor	18
3 Camera & Lighting	19
3.1 Linescan camera	19
3.2 Lens	21
3.3 Lighths	22
3.4 Filter	23
4 Neural Networks	24
4.1 Neural Networks	24
4.2 Convolutional Neural Networks	24
4.3 Fully Convolutional Networks	26
5 Software Implementation	28
5.1 Motor Control	28
5.2 Image Aquisition	28
5.3 Image Preprocessing	29
5.4 Neural Network	30
6 Tests & Project Future	33
Conclusion	36
Bibliography	37
Appendices	39
List of appendices	46

List of Pictures

1.1	Sample of tesil fibre. [2]	12
1.2	Carding machine straightening colored wool. This is an industrial machine, the laboratory uses much smaller one. [15]	13
1.3	This is a sample of corded fiber with many defects (bundles)between two glass panes.	14
1.4	Absorption spectrum of Rhodamine B. [11]	15
1.5	Fluorescence spectrum of Rhodamine B. [11]	15
1.6	Absorption (blue) and fluorescence (orange) spectra of Rhodamine B. [11]	16
2.1	Scanner setup with frame (violet), motor (yellow) and “C” arm (red) highlighted.	17
2.2	Second prototype of the scanner. The front legs were removed to allow free movement of the “C” beam.	18
3.1	Close-up to camera, and both transmission and fluorescence lights.	19
3.2	Placement of Transmission and Fluorescence lights. Side view.	22
3.3	605 nm.	23
3.4	635 nm.	23
3.5	470 nm, 645 nm.	23
4.1	Example of kernels in first layer of CNN. [17]	25
4.2	Example of CNN architecture (VGG16). [4]	26
4.3	Example of fully connected network from Stanford lectures [5].	27
4.4	Description of transposed convolution from Stanford lectures [5].	27
5.1	Motor control GUI.	29
5.2	GUI controlling both camera and linear drive.	30
5.3	Model accuracy and model loss.	32
6.1	Layer of rhodamine colored fibres between two glass sheets.	33
6.2	Image under transmission light(left) and fluorescence light (right).	34
6.3	Image under transmission light(left) and fluorescence light (right).	34
6.4	Image under transmission light(left) and fluorescence light (right). Here would be the thick fiber completely undetectable without rhodamine coloring.	35
6.5	Prediction of the network. Larger verision available as figure A.11.	35
A.1	Linear actuator by STANDA®.	39

A.2	Standa linear actuator schematics.	39
A.3	Linescan camera – front.	40
A.4	Linescan camera – back.	40
A.5	Nikon lens 50 <i>mm</i>	41
A.6	Set of close-up lenses.	41
A.7	Light filter temporarily fixed to the back side of a lens. Note professional attachment.	42
A.8	COM port selection screen.	43
A.9	Operating software of STANDA [®] motor.	43
A.10	Example of the training dataset. Red color is for true labels, green and blue are for fluorescence and transmission respectively.	44
A.11	Prediction of the network.	45

INTRODUCTION

SILON s.r.o has been manufacturing TESIL[®] polyester fibres for over 50 years and was creating them using exclusively PET bottle flakes since 2002 [1].

Quality control is essential part of every industrial process. TESIL[®] has been, until now, inspected by hand, using carding machines. The carded fiber is laid on lit surface and quality inspector checks for any thick and/or bundled threads. Final number is counted and then extrapolated to whole batch.

The machine consists of a linescan camera, two lighting units, linear actuator and a glass cover, creating a scanning unit. Two prototypes have been built, demonstrating scanning capabilities, resolution and field of view of camera. The lights are tested with different filters and different configurations to find optimal solution.

Software tools used to control the drive and the camera are also presented and discussed.

The tools for creating neural network are presented, along with the network itself and the result is tested and evaluated.

APPLICATION

CHAPTER

1

The company's Silon s.r.o name comes from first product – polyamide filament yarn – that was used for ladies' stockings. Today, Silon is a manufacturer of polymer compound granulates and polyester fibres for use in construction, automotive industry and for hygiene and medical products. The fibres are made exclusively from used PET bottle flakes since 2002.

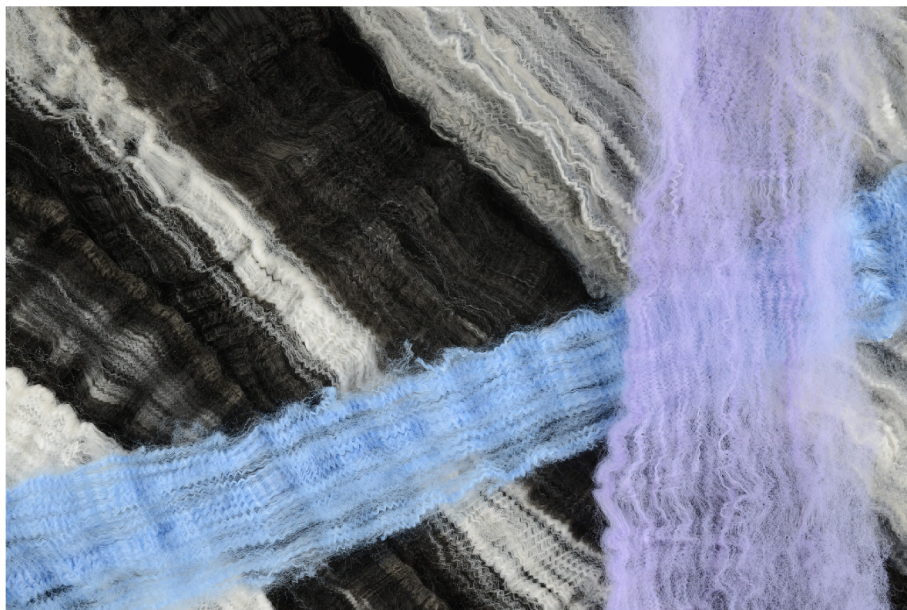


Figure 1.1: Sample of tesil fibre. [2]

The quality control is essential part of manufacturing. It consists of several steps, according to [3]:

- basic mechanical fiber properties (strength, elongation at break, fineness),
- color shade inspection,
- crimp characteristics,
- staple length,
- friction and cohesion measurements,
- staple length,
- shrinking.

1.1 Current process

At this time is the process of quality control fully manual. The lab assistant's job is first to prepare the sample to be examined. This is done in a carding machine. It draws and straightens bundles of raw fibre that has come off of the production line. Those fibres are then inspected for nondrawn fibres (that are thicker and tend to break), and bundles of material that is stuck together. The laboratory sample between two glass panes can be seen on figure 1.3.



Figure 1.2: Carding machine straightening colored wool. This is an industrial machine, the laboratory uses much smaller one. [15]

The fibre is often dyed by fluorescent coloring named Rhodamine B. This chemical compound and dye, red in color, is used for quality control of polyester fibres. It sticks better to nondrawn and bundled fibres due to different molecular structure. Fluorescence spectrum has one peak at 564 nm [11]. That corresponds to yellowish green in visible spectrum. It's the color that will be emitted after excitation with absorbed light. Its absorption spectrum has two notable peaks; first one around 358 nm , the second one at 543 nm . First wavelength is in ultraviolet spectrum and is used in human inspection, because UV light, being invisible, will not prevent from seeing the emitted light. Second peak is much higher, but also very close to emitted wavelength, so it's more challenging to isolate those two colors and this wavelength is not usable by humans. For reference see figures 1.4, 1.5 and 1.6.

The search is very time-consuming, often taking twenty minutes to half an hour to sort through the sample and find all defects. It's also very monotonous and tiring to constantly look at UV lit surface, even when the worker is required to wear safety goggles with UV filter. That means that many defect might not be found, degrading quality control process.

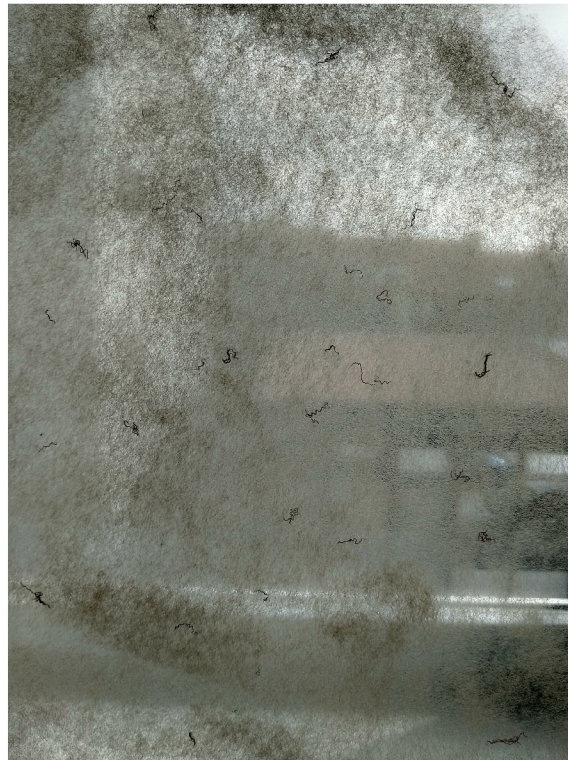


Figure 1.3: This is a sample of corded fiber with many defects (bundles) between two glass panes.

1.2 Proposed improvement

Main goal of this thesis is to find and propose solution to partial automation of defect searching process. The idea is to build a scanning bench with high resolution camera and lights. The sample will be pressed between two glass panes in a uniform layer. This should reveal any bigger bundles (as seen on figure 1.3, where the sample has extra defects added) and allow the sample to be properly scanned. The picture will be taken with a high resolution line-scan camera, similarly to standard scanners. The image will be then saved on computer for further analysis.

The proposed method is to combine both fluorescence and transmission images and treat them as two channels of one image. Analogically to usual RGB image, with this having two channels, both taken as grayscale pictures. These images will then be fed into neural network that is supposed to discriminate between defect and non-defect on the pixel-base level, thus creating another layer of image with highlighted areas where the defects are present.

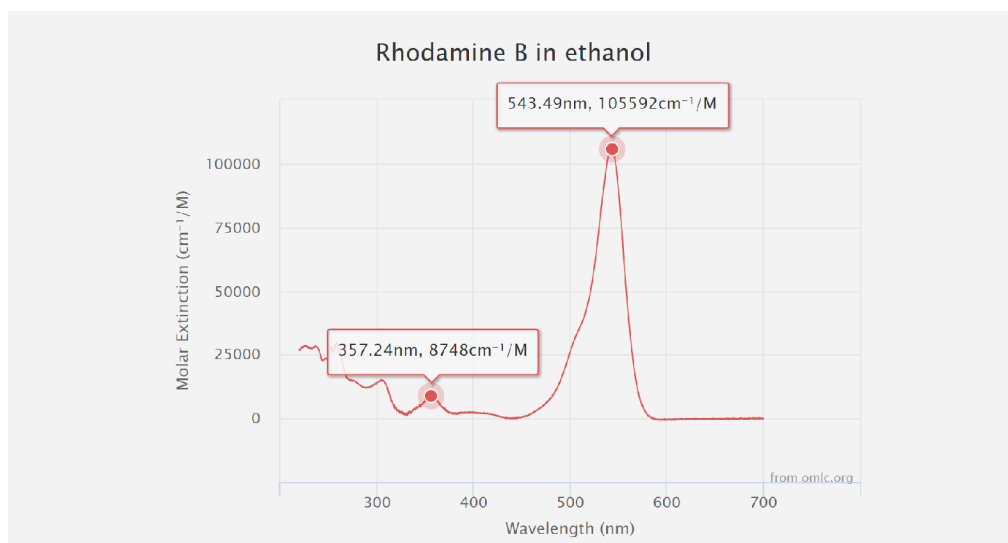


Figure 1.4: Absorption spectrum of Rhodamine B. [11]

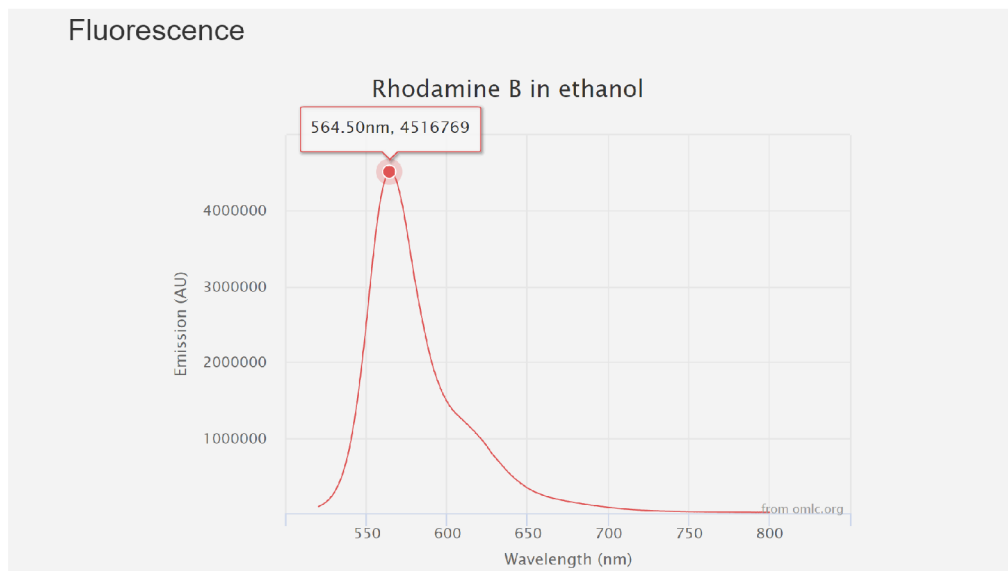


Figure 1.5: Fluorescence spectrum of Rhodamine B. [11]

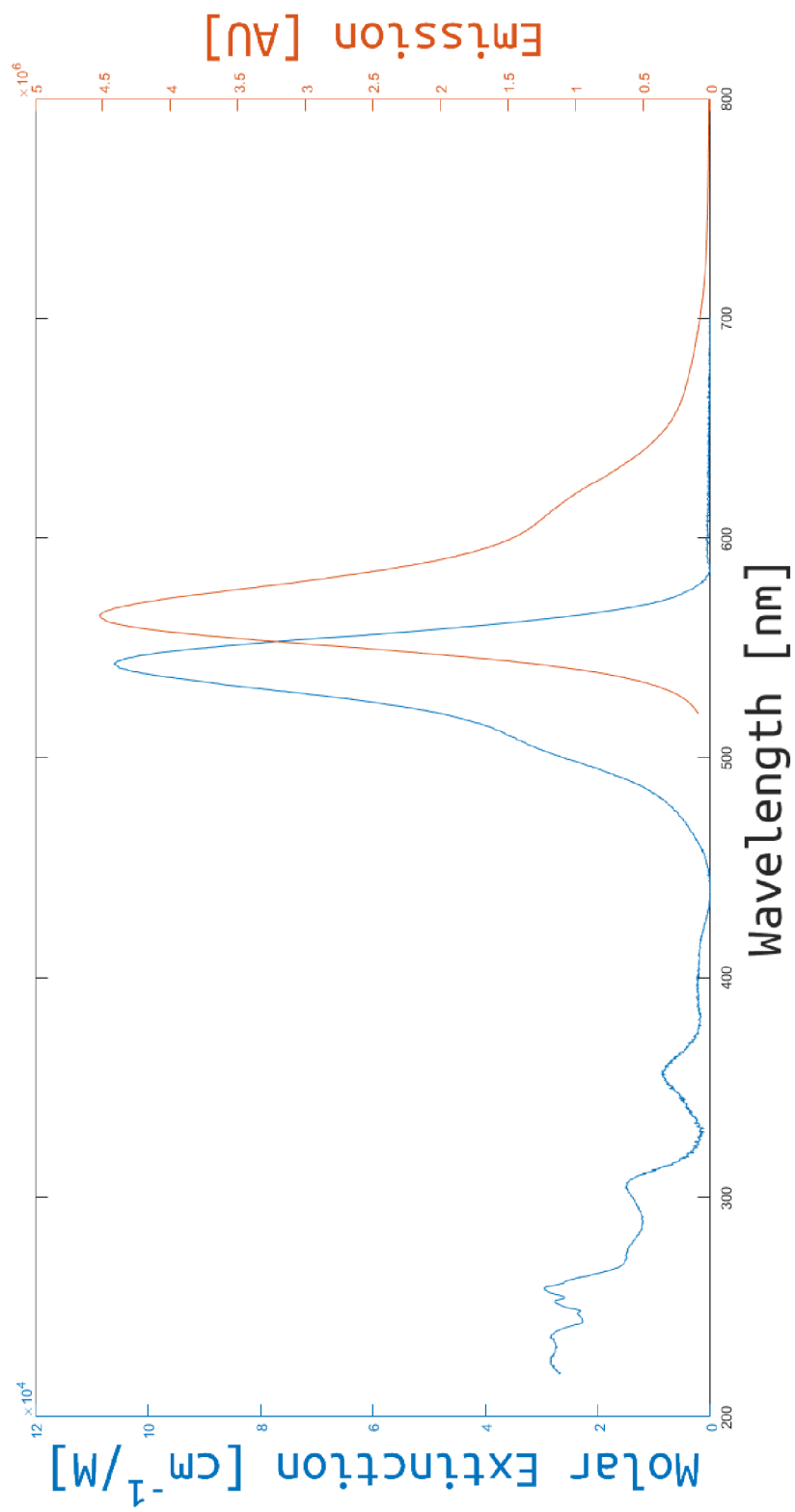


Figure 1.6: Absorption (blue) and fluorescence (orange) spectra of Rhodamine B. [11]

FRAME & DRIVE

CHAPTER 2

During the development there were two prototypes built, both of them similar in design, the second being more streamlined and lighter. Both scanners use MiniTec[®] aluminium profiles and feature one linear motor, two light units and one line-scan camera.

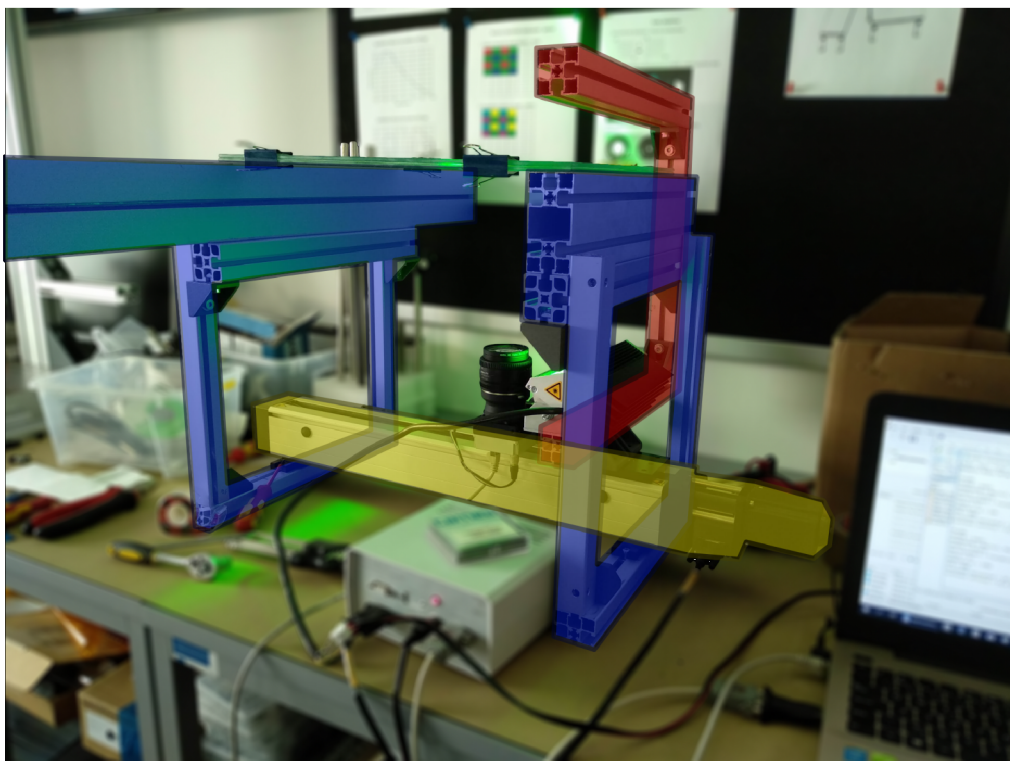


Figure 2.1: Scanner setup with frame (violet), motor (yellow) and “C” arm (red) highlighted.

2.1 Frame

As was mentioned, whole construction is made from MiniTec[®] aluminium profiles. The linear motor is serving as a centerpiece to the whole machine. On both sides there are aluminium stands supporting glass working plane laid on top of them. Second version has only two legs, so the top light can move freely across the whole length. On the linear actuator is affixed “C” arm that serves as a support for transmission light. The very first prototype (not featured on pictures) had camera on top of the moving beam, but vibrations during movement caused degradation of resulting image, so the camera has been moved to the bottom.

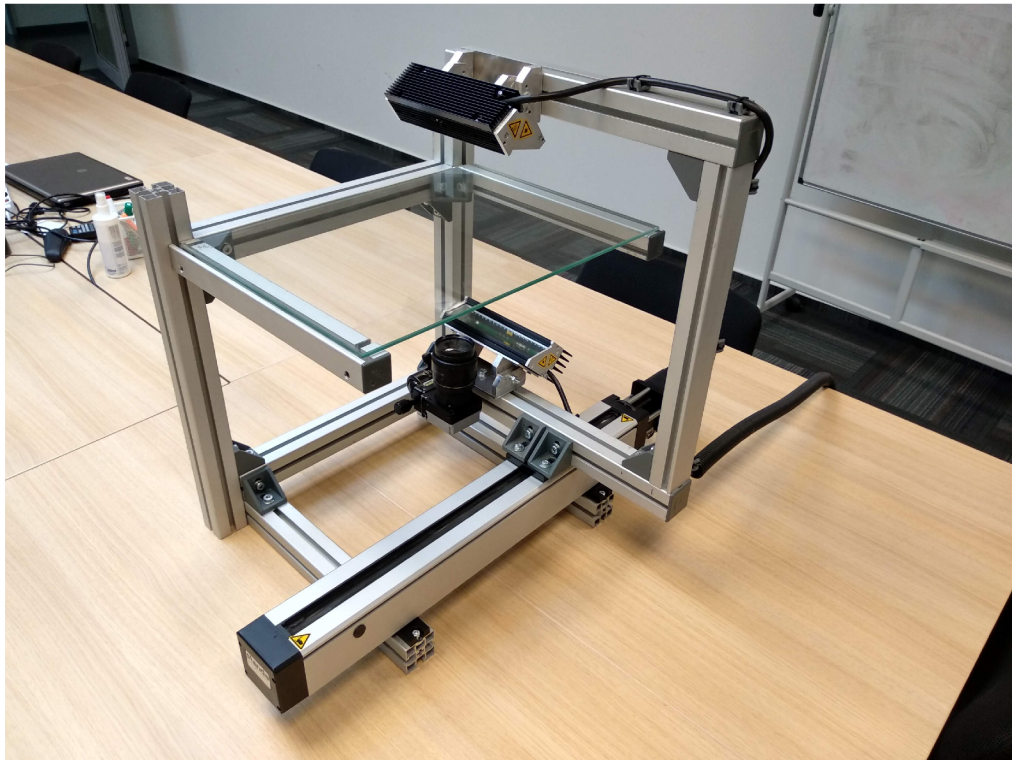


Figure 2.2: Second prototype of the scanner. The front legs were removed to allow free movement of the “C” beam.

2.2 Linear Motor

The linear actuator is made by Lithuanian company STANDA[®]. The motor is controlled by two-axis driver that is powered by 12 – 36 VDC and communicates with computer using simple RS-232 protocol. There’s been discussion about using second actuator perpendicular to the first one that will allow scanning of wider strips of material in multiple passes. This version however currently uses only one.

This motor unit has very simple interface with all parameters available for setting. With poorly written manual though, it was difficult to find proper setup. Many combinations of parameters resulted in jerky movement, unacceptably loud noise or motor not moving entirely. After the most quiet setting was found, it was set as default and informed other decisions in design, as describer further in chapter 3.

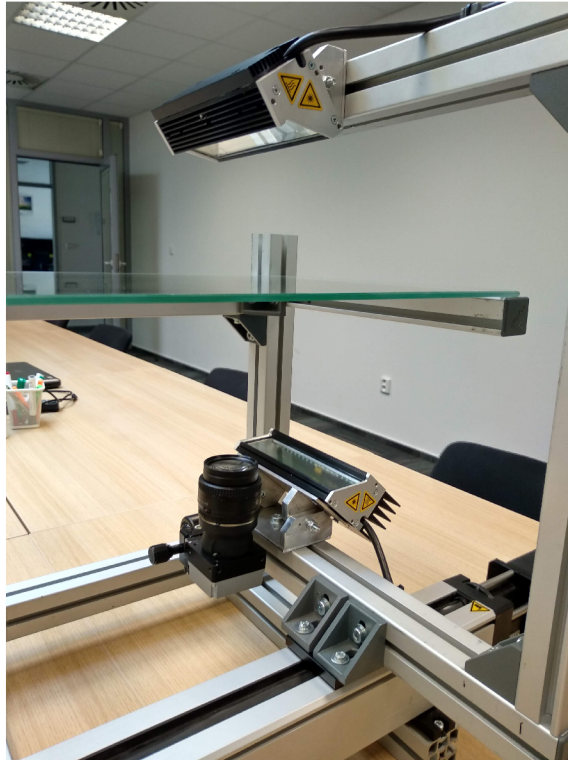


Figure 3.1: Close-up to camera, and both transmission and fluorescence lights.

3.1 Linescan camera

This type of camera reads one line at a time. This makes it suitable to scan long strips of material, where one picture from conventional area scan cameras doesn't cover the whole object. It's especially advantageous for moving objects, for example endless webs of material that move underneath the camera.

The other possibility is to move the camera, which is the case in most scanners, this one being no exception.

Single Basler Racer linescan camera was used. It has resolution of 6144 *kpx* and maximal refreshing rate 17 *kLPS* (thousands of lines per second).

The camera is situated under the working plane, looking up. Alternative position above the working plane on top of "C" arm was refused due to vibrations during movement.

3.1.1 Resolution of linescan camera

The resolution was computed using calibration sheet. This is usually piece of paper with print of exact and known dimensions, that is scanned with the camera. Apart from calibration of traditional matrix camera, speed of movement and shutter frequency is also an important factor in the linescan camera. The translation speed was set due to noise limitations of the motor, so the shutter frequency f_s was set according to following formula:

$$t_s = t_m. \quad (3.1)$$

This means that to achieve square pixels, the time to take one picture must be equal to the time the drive moves to the position of the next pixel.

$$f_s = \frac{1}{t_s}, \quad (3.2)$$

$$v = \frac{s}{t_m}. \quad (3.3)$$

Frequency of the shutter and speed of the drive are trivial.

$$f_s = \frac{v}{s}. \quad (3.4)$$

Units: t_s is time required to shoot one picture, one pixel high; [s],
 t_m is time required for the drive to move one pixel further; [s],
 f_s is frequency of the shutter; [Hz],
 s is length of one pixel; [m],
 v is speed of the drive; [$m \cdot s^{-1}$].

Frequency of the shutter is speed of the drive over desired length of one pixel. The aim is to achieve the same length as height, as pixels in computer are square and the image would seem to be compressed or elongated otherwise.

The other approach is experimental; camera scans the calibration sheet several times with different shutter speeds and the value is found experimentally. Due to insufficient documentation on the drive and technical complications tied to exact measurements of speed and submillimeter lengths, the second approach was chosen.

When the final image of the calibration sheet is stitched together, the final resolution may be computed as follows:

$$r = \frac{l}{N}. \quad (3.5)$$

Units: r is computed resolution; [m],
 l is measured length; [m],
 N is number of pixels in the measured length; [-].

This setup was able to achieve resolution of $12 \mu m$ per pixel. Compared to standard office copier that scans at 300 dpi which is $84 \mu m$ per pixel. That is

seven times better performance, which allows for very detailed images. However, this setup has several drawbacks that will be explained further.

3.2 Lens

The ordinary 50 *mm* Nikon lens with variable aperture was used for prototyping. There is also close-up lens added to the front to lower the distance between camera and the observed plane. This also distorts the image on the edges, so for a trade-off between lower distortion and distance, +4 diopters close-up lens was chosen.

The 50 *mm* lens has angle of view equal to 47° that creates 15 *cm* field of view, together with the close-up lens. Given that sides of the picture are distorted and generally out of focus, the resulting working area is even narrower and the system will need either multiple cameras scanning overlapping strips, or second linear drive moving the camera perpendicularly to achieve same effect in multiple passes. To combat out of focus borders, it's advisable to increase the aperture. But this allows less light onto the chip, thus requiring stronger lighting and/or increasing exposure time. The exposure is however limited by the shutter speed. The camera parameters can be seen in tables 3.1 and 3.2.

parameter	value
shutter speed	500 <i>LPS</i>
exposure	1.99 <i>ms</i>

Table 3.1: Fixed settings of the camera.

mode	parameter	value
fluorescence	gain	2047
	aperture	5.6
transmission	gain	1000
	aperture	22

Table 3.2: Variable settings based on the type of lighting.

The gain value is arbitrary 2^9 number set by API.

Because the aperture increases the depth of field, the main goal was to increase it as much as possible. That means increasing exposure to maximum achievable value and adjusting brightness by gain which should remain as little as possible to reduce noise. (Exposure times shutter speed is not equal exactly one second because of the camera's computation overhead.)

The transmission mode gets more light in the camera so the aperture can be closed all the way while gain can be on the half of maximum value. Fluorescence carries much less light so the aperture must be partially open, while gain is at maximum.

3.3 Lighths

The scanner features two Corona II LED linescan lights, made by german company Chromasens. They are generally used for linescan cameras as transmission or reflect lights. While this scanner uses lights with infinite focal length, there are different focal lengths available.

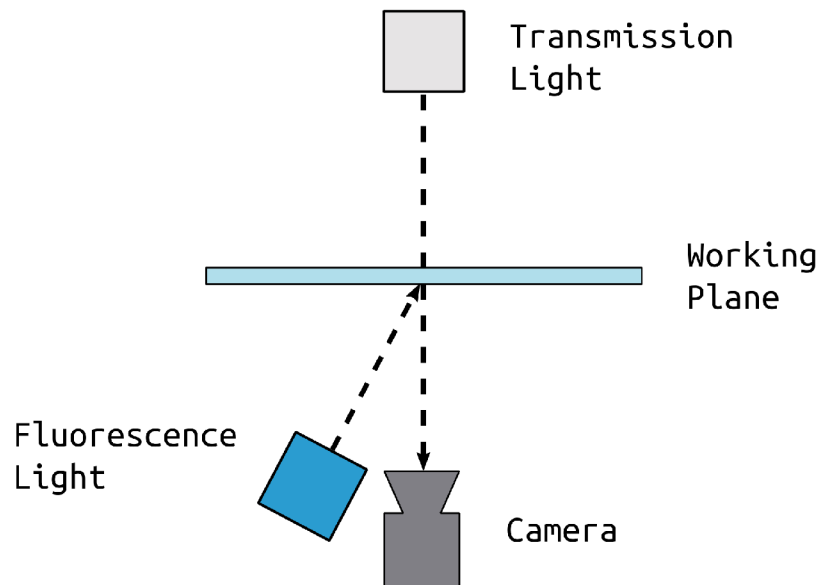


Figure 3.2: Placement of Transmission and Fluorescence lights. Side view.

3.3.1 Fluorescence

When analyzing the fibre sample, lab assistants use UV light when searching for defects dyed in Rhodamine. Because Rhodamine has much higher response to green light, it's more convenient to use green LED light with narrow spectral characteristics to excite the dye and then use filter on the lens to let in only fluoresced light. The wavelength of green LED diodes in Corona light is 520 nm . When compared to figure 1.6, the highest peak in absorption spectrum of Rhodamine is right around this value.

As can be seen on figure 3.2, the green light shines straight on the plane that is in camera view. Apart from reflection camera, the fluorescence occurs in every direction and the reflected light from glass plane is avoided.

3.3.2 Transmission

This light sits above the camera, on the other side of moving "C" profile from camera. It illuminates the working plane and shines straight into the lens. There

is also diffusor cover between the light and the working plane to ensure uniform distribution of light. That creates bright image in the camera, except for light blocking objects. They appear as black shadows on bright surface. This is especially helpful for detection of bundles of threads and thick standalone threads. The light is white in color so the filter lets at least part of it come through.

3.4 Filter

Among available filters, three were selected and tested as fluorescence filter. Important properties were amount of fluoresced light gone through (the more the better) and amount of background light from other sources (the less the better) such as exciting light, sun, etc.

Comparison of all three filters can be seen in figures 3.3 through 3.5. First and second filter are single frequency filters with FWHM (full width at half maximum) of 50 nm . The third filter has two bands, one at 470 nm , the other one at 645 nm . Second filter has very little light going through, as the filter's wavelength is far from fluoresced one. The third filter has lots of ambient light shining through the sample. The first filter has the best definition and crisp edges, so this filter was used for the rest of the experiments.

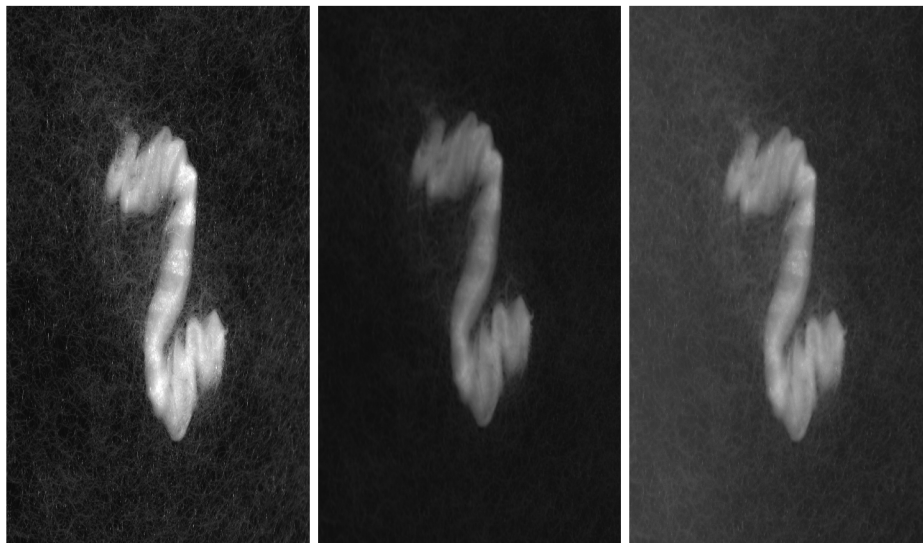


Figure 3.3: 605 nm .

Figure 3.4: 635 nm .

Figure 3.5: 470 nm , 645 nm .

In recent years there has been enormous boom in neural networks with emergence of convolutional neural networks. That allows many possibilities in image processing; previously too complex tasks are now feasible with use of machine learning.

4.1 Neural Networks

Artificial Neural Networks (ANN) are an attempt to model vast web of connections in human brain. First model of neuron – perceptron – has been proposed in 1958 [14]. After criticism of this mathematical model by Minsky and Papert [13] whole research field of machine learning stagnated. They proposed two key issues: current neural networks weren't able to reproduce simple XOR logic gate, the second being the computing power required by networks was beyond what was possible at the time.

New interest in machine learning and ANNs was sparked by Werbos' backpropagation algorithm in 1975 [16]. That allowed learning of multilayered networks, resulting in overcoming the XOR problem. Also, computing power has been steadily growing, allowing more complex applications and whole field of study became more accessible for wider audience.

4.2 Convolutional Neural Networks

Second (and recent) advance has come with creating deep convolutional neural network (CNN) in 2011 [12]. It also models function of animal brain, more specifically, function of visual cortex, part of brain that processes visual information. The network uses little to no standard preprocessing methods and instead is emulating them via complex of interconnected convolutions. The weights of every convolutional kernel are subject to adjustment by the backpropagation algorithm. That is useful for applications where preprocessing using standard (non-learning) methods are very complex or even impossible to manufacture.

The convolutional network follows traditional concept of layers. Instead of using fully connected layers though, that get rid of majority of spatial information, convolutional layers are main tool.

4.2.1 Convolutional Layers

They use mathematical operation called convolution, that is usually used in image processing as first step of edge detection, blurring and other applications.

Convolution uses kernel, that is usually hand-picked for specific task. In case of neural network, each value of this kernel becomes parameter that is affected by learning and participates in final error of the network for backpropagation algorithm. This way the previous, deliberately picked kernels, are replaced by organically evolved substitutes.

It's possible to visualize the results as seen in image 4.1. It's much harder to visualize the kernels in deeper layers. The input image has only three channels (RGB), but deeper representations can have even hundreds of layers, which is impossible to show easily in three-dimensional color space.

The convolutional layer also uses strides to determine size of output image. If the stride is one in each direction, resulting image will be the same as original. Stride two in any direction means, that every other pixel will be left out as center, but will participate as border in others. In stride three, for 3×3 kernels, some pixels will be left out completely.

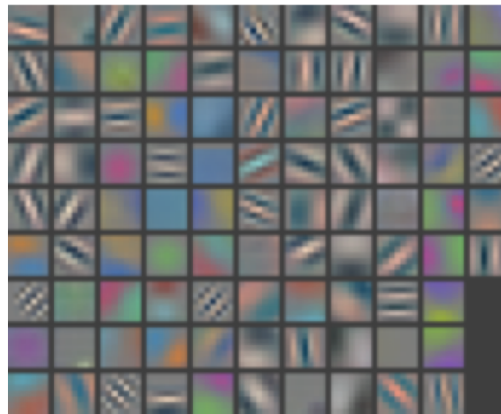


Figure 4.1: Example of kernels in first layer of CNN. [17]

4.2.2 Pooling Layers

Every neural network works with enormous amount of data, and with CNNs even more so. The pooling layers have helped to handle the amount of parameters in the network. Most common and generally most effective method is called *max pooling*. It takes highest value from area of the picture, say a square 2×2 . This done on whole picture effectively downsamples the picture to quarter its size, lowering number of parameters required in the network.

This also means that some information is inevitably lost by this process. Therefore it's important to be aware of this when designing the network.

Another type of pooling is for example *average pooling*.

4.2.3 Batch Normalization

Batch normalization is process that speeds up learning and allows each layer to learn independently on the previous layers. It normalizes (subtracts the mean and divides by standard deviation) currently processed batch. That means, that

every layer will work with normalized and therefore similar data, so the weights won't drift so far from starting location.

4.2.4 Fully Connected Layers

Last part of convolutional network is usually some sort of classification. One of the ways is to use fully connected standard neural network.

This is however by no means necessary. What the convolutional part of the network creates in fact is a feature vector, that can be used in numerous ways, only one of which is neural network. One could as well put any other classifier at the end, for example SVM (support vector machine) decision trees, etc. On the picture 4.2 you can see four layers of fully connected network, ending in 1000 categories.

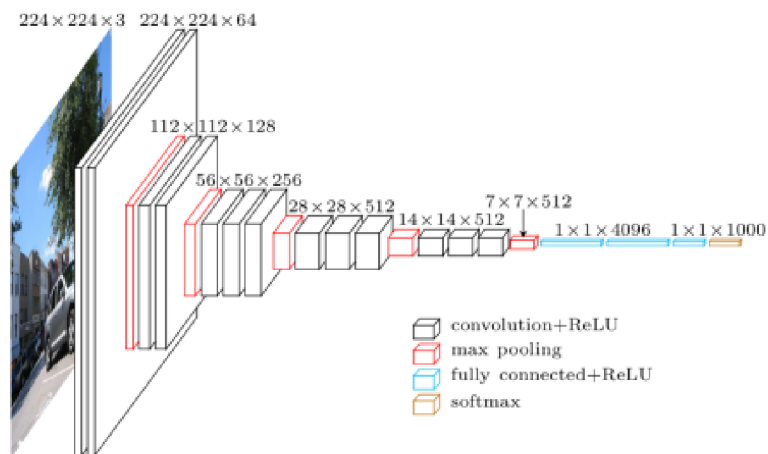


Figure 4.2: Example of CNN architecture (VGG16). [4]

4.3 Fully Convolutional Networks

Usual CNNs create feature vector that can be then classified into single or multiple classes. But where's the need for image segmentation instead of classification, the standard CNN falls short. That's where *fully connected network* (or FCN) comes.

Instead of having picture with classes associated to it forming the training dataset, there is now set of two pictures. First one is the original, the second one is a map of present objects. That allows for finding borders of distinct objects in the picture, but is also very difficult to get dataset large enough to train this type of network.

This network consists of convolutional layers, pooling layers and new type of layer, that is (according to Stanford lectures [5]) called transpose convolution. It's sometimes in the literature incorrectly called "deconvolution".

The transpose convolution uses its filter to project single value to larger field as explained in figure 4.4. Other types of upsampling include various versions

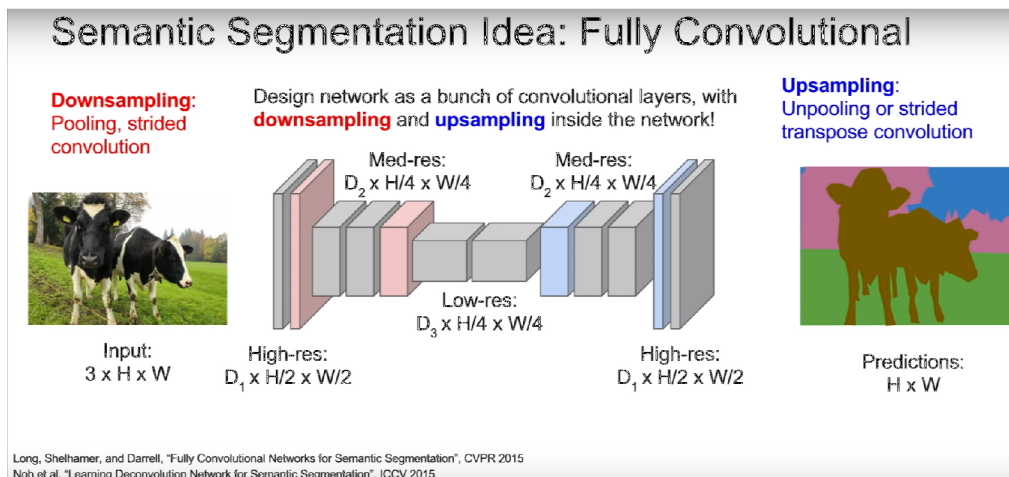


Figure 4.3: Example of fully connected network from Stanford lectures [5].

of unpooling, also discussed in the lecture. It can be copying single value to 2×2 fields, or placing a single value in one of the corners of this 2×2 field and filling the rest with zeros (so-called "bed of nails").

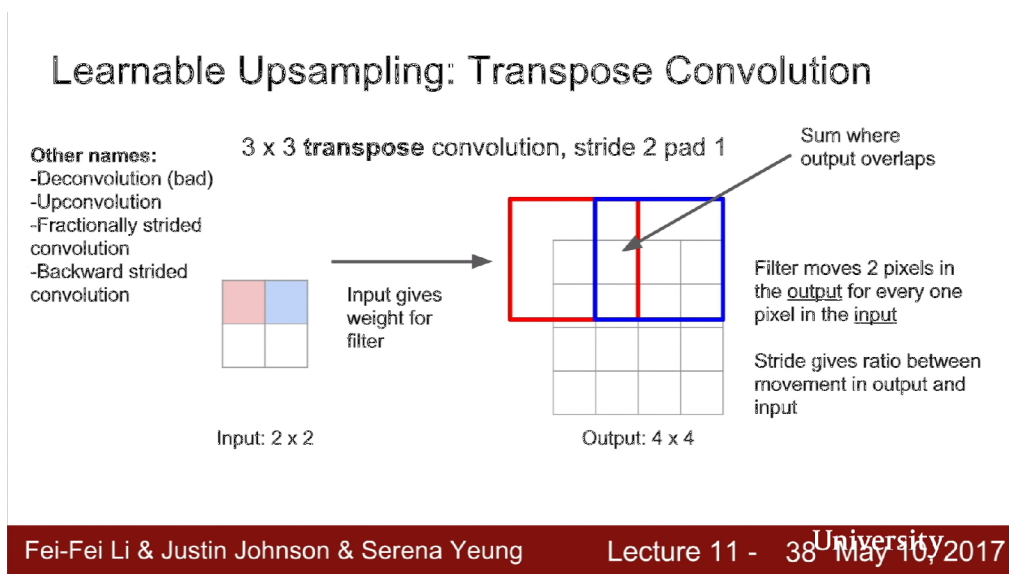


Figure 4.4: Description of transposed convolution from Stanford lectures [5].

SOFTWARE IMPLEMENTATION

5

CHAPTER

During the lifetime of the project, several programs were written to control various parts of scanner. While one unified interface is still not created, majority of the scanning process is automated and controlled by custom software. The motor came with only a basic library for C, so the effort was made to write more complete API to operate the drive. Both motor and camera controls are written in C#, while the neural network was completely written in Python. The decision and advantages are discussed further. The image preprocessing script was written in MATLAB as it was meant to use one time only.

5.1 Motor Control

Company Standa included simple library written for C with only three commands: Open connection, Send command and Close connection. For purposes of this project, simple library in C# was written to accommodate basic required tasks, such as move set number of steps, go to either end, set acceleration ramp and so on. List of commands is presented as an appendix. Because the provided library is so simple and the communication is a simple protocol across RS-232, it would've been arguably easier to write own library from the ground up.

With the library came also simple demo program allowing to control the motor via GUI or simple custom terminal. Because the applications were so poorly created though, they were used only in the initial tests. Custom GUI handling basic commands had been written in C#.

5.2 Image Aquisition

Basler cameras come with advanced proprietary software, ready to set up any camera and use it in various modes. It also comes with full library in C# ready to work with every camera, however custom class specific to Basler Racer had to be written. It used event system generated by camera to handle every incoming image and either show it on the monitor, or save it on the disc.

Apart from that, custom GUI, handling both motor control and image aquisition had been created. It only served test purposes so it had only basic functionality, but it allowed to start the motor run and the image aquisition in one button click.

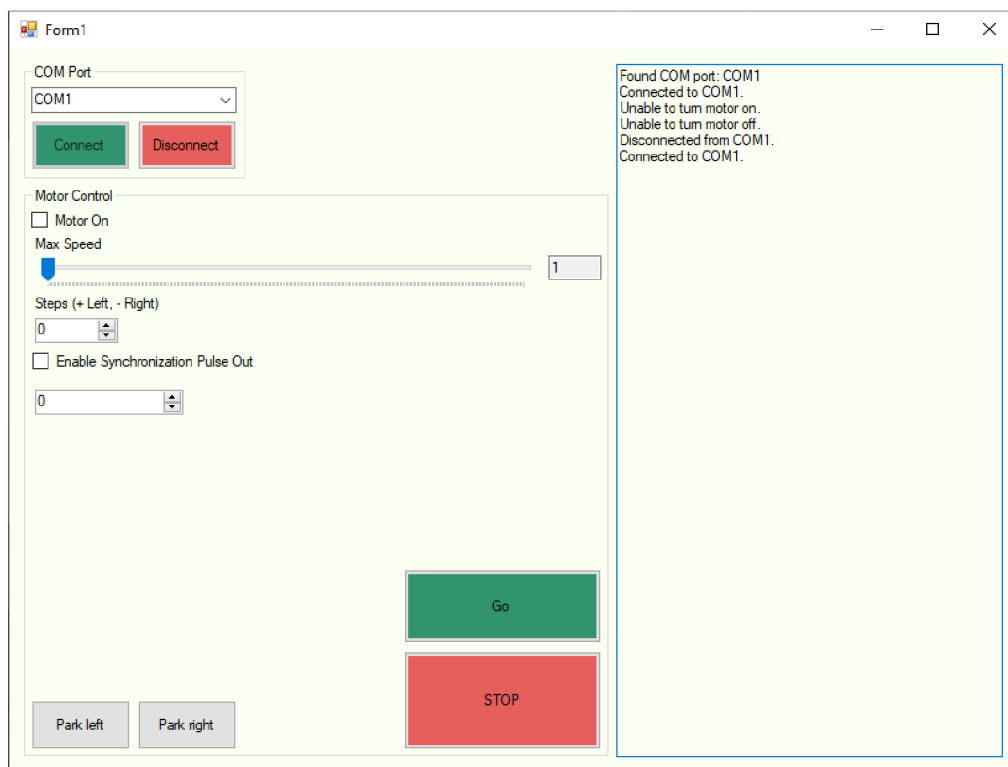


Figure 5.1: Motor control GUI.

5.3 Image Preprocessing

Resulting images have resolutions around $6000 \times 20000 \text{ px}$, which is unsuitable for convolutional neural network that accepts input usually in hundreds of pixels across. That means the input image had to be split into manageable chunks, in this case $512 \times 512 \text{ px}$ large. The images were composed of layer of transmission image and fluorescence image, separately, but in same manner, true values were kept. They were created by tracing each defect by hand, making map of defects in large image. To create training and testing dataset, the resulting images were rotated and flipped in every direction, effectively creating eight times larger datasets. The algorithm 1 illustrates the process.

```

initialization;
while images left in folder do
    read fluorescence image;
    read transmission image;
    create two-channel image;
    read true image;
    rotate and flip images;
    append images to dataset;
end
save dataset;

```

Algorithm 1: Image preprocessing.

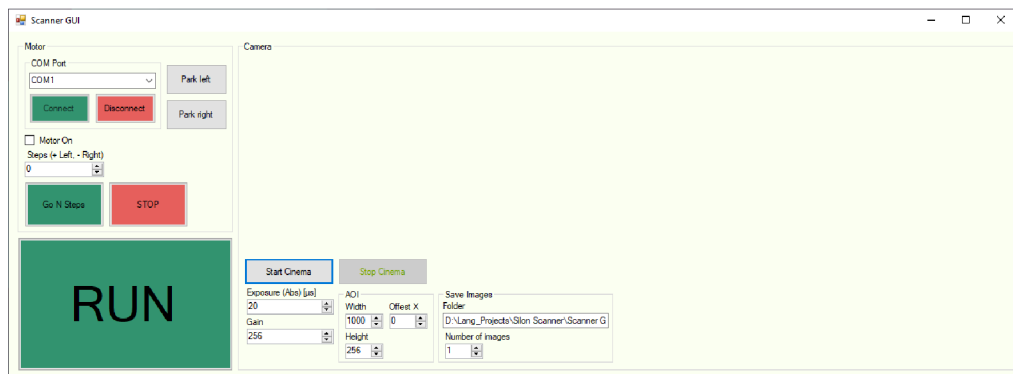


Figure 5.2: GUI controlling both camera and linear drive.

5.4 Neural Network

For the convolutional neural network, the Python programming language has been chosen. It currently has the largest community of data scientists and programmers interested in machine learning, provides most backend and frontend frameworks and has largest amount of reference material available.

While MATLAB might have been good tool for fast prototyping, Python, with its data manipulation and data science package, and its availability for free, has gained ever growing popularity and in field of neural networks with support of GPUs, is downright superior.

5.4.1 Backend

There are many many choices of backend for CNNs, such as Tensorflow [6], Theano [7], MXNet [8], CNTK by Microsoft [9], PlaidML [10], etc.

Tensorflow is an open source library for tensor computation. It is developed by Google and is available for both CPU and GPU computing. GPU is using CUDA[®], parallel computing platform developed by NVIDIA, running on their graphics cards. Tensorflow was chosen for familiarity with the framework and for its popularity.

5.4.2 Frontend

As a Python 3 package, Tensorflow provides frontend suite named Keras. Keras is a high-level API, written in Python. It runs on top of Tensorflow, Theano or CNTK. It serves as unified interface, inspired by scikit-learn – another machine learning Python package – allowing fast prototyping.

5.4.3 Architecture

For the application, the fully convolutional network (FCN) has been chosen because the output should be in form of an image, highlighting defects. Several designs has been tested with the final one presented in table 5.1.

layer type	output channels	strides	note
Conv2D	10	1	
BatchNormalization			
Conv2D	10	2	
BatchNormalization			
Conv2D	15	2	
BatchNormalization			
Conv2D	20	1	
BatchNormalization			
Conv2D	40	1	
BatchNormalization			
Conv2D	40	1	
BatchNormalization			
Conv2DTranspose	20	2	
BatchNormalization			
Conv2DTranspose	20	2	
BatchNormalization			
Conv2D	2	1	softmax activation

Table 5.1: Architecture of the FCN.

The network has shape of an hourglass. The stride two halves the resolution of the image, while the increasing number of channels preserves the amount of information. In the transposed convolution, the stride two means basically stride $1/2$, which results in doubling the resolution. Every layer uses the ReLU activation function (nonlinearity), except for last one, which takes the result with maximum value for each pixel. The padding of each layer has been set so the original dimensions would be preserved.

5.4.4 Dataset

Due to a time consuming process of defect highlighting in graphical software, the working dataset was fairly small, which harmed the performance of the network. The original dataset consisted of thirty unique defect images. After enhancing dataset by rotating and flipping the images, the resulting dataset had 240 images. Example of the dataset can be seen in figure A.10. Red color is reserved for true labels, green and blue are for fluorescence and transmission respectively.

5.4.5 Learning

Thanks to Tensorflow and Keras, training the model is matter of calling one function. The model is generated and initialized with random values. After that, the training process presents every training sample in batches and continues to do so for number of epochs. Due to memory constraints, the batch size has

been quite small – only seven samples. The training process was going for 200 epochs, which even with high-end GPU took about 15 minutes to complete. After each training session it was possible to save trained model and resumed training anew. The simple version is shown in the algorithm 2.

```
initialization;  
load dataset;  
start learning session;  
initialize model;  
for  $i = 0$  to  $epochs$  do  
  | fit model;  
end  
clear session;  
save model;
```

Algorithm 2: Model fitting.

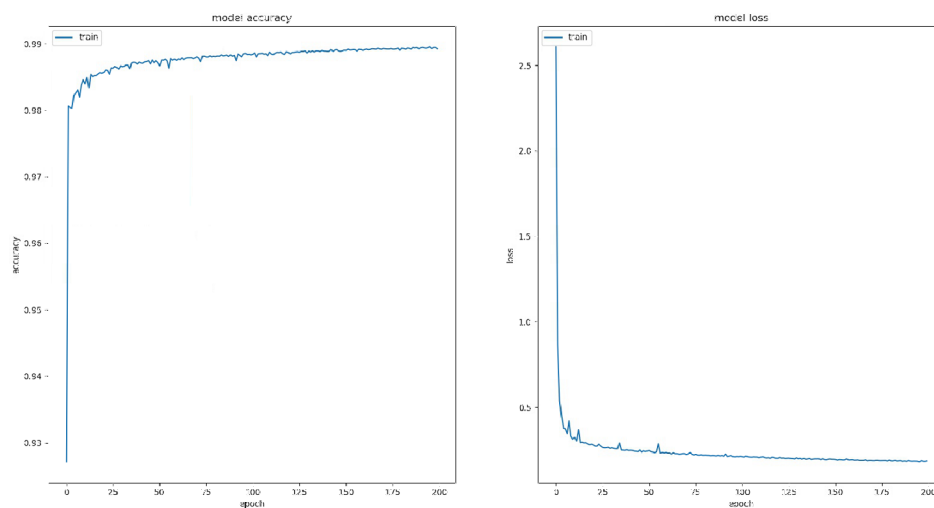


Figure 5.3: Model accuracy and model loss.

TESTS & PROJECT FUTURE

CHAPTER

6

First tests were made on the first scanner prototype to determine the best filter to use. The results are presented in figures 3.3 through 3.5 and discussed in section 3.4. The figure 6.1 shows the sample that was tested. As was stated before, the combination of 520 *nm* light and 605 *nm* filter was chosen as most suitable.

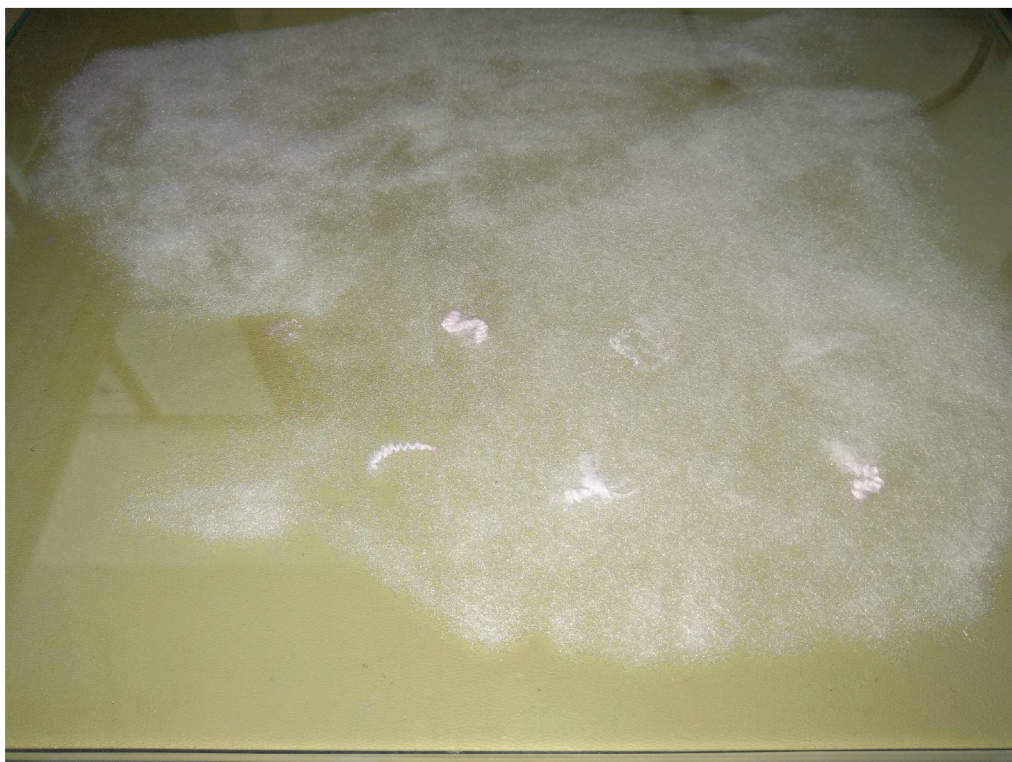


Figure 6.1: Layer of rhodamine colored fibres between two glass sheets.

Comparing transmission and fluorescence images in figures 6.2, 6.3 and 6.4 show the importance of both modes of lighting, as some defects get completely lost without the fluorescence light.

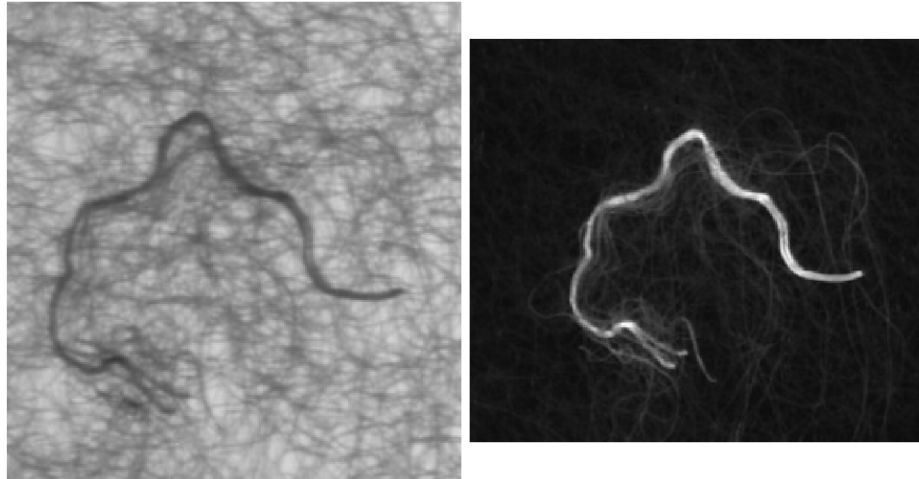


Figure 6.2: Image under transmission light(left) and fluorescence light (right).

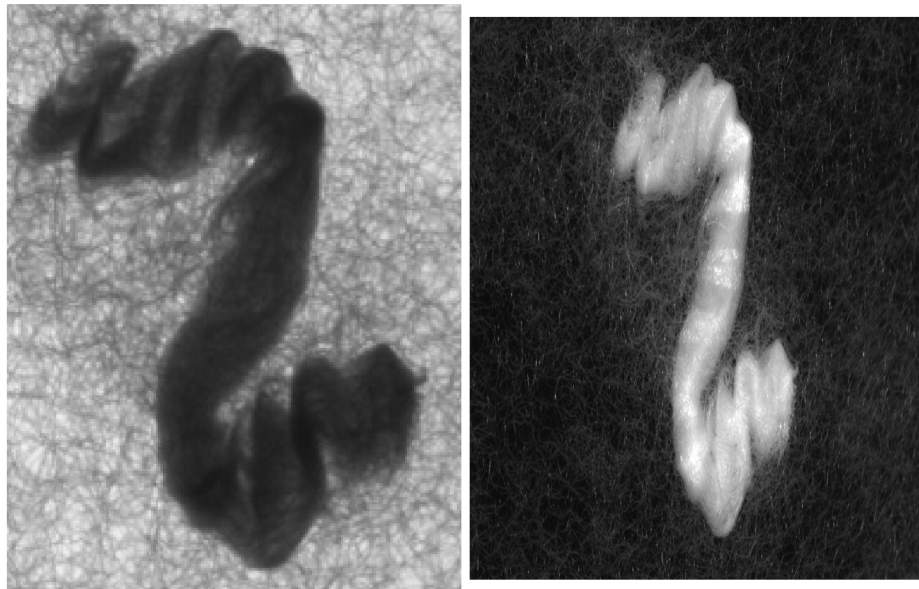


Figure 6.3: Image under transmission light(left) and fluorescence light (right).

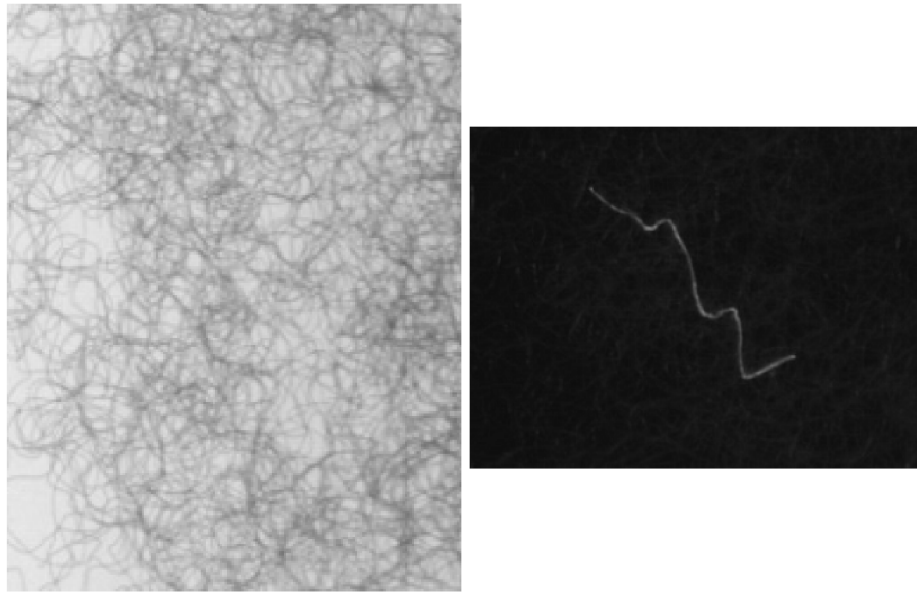


Figure 6.4: Image under transmission light(left) and fluorescence light (right). Here would be the thick fiber completely undetectable without rhodamine coloring.

The final figure 6.5 shows the performance of the network. The first and second column show both channels of the input image. The third and fourth images are the same, but inverted, because the discriminator is binary. They are the predicted data, from which the fourth column is predicted defect. Fifth column shows the ground truth.

It can be seen that it's fairly good at recognising well defined structures, either in transmission, or fluorescence. But it still has troubles in partially obscured areas. Further development and fine-tuning of the network will be done before presenting the system to the customer.

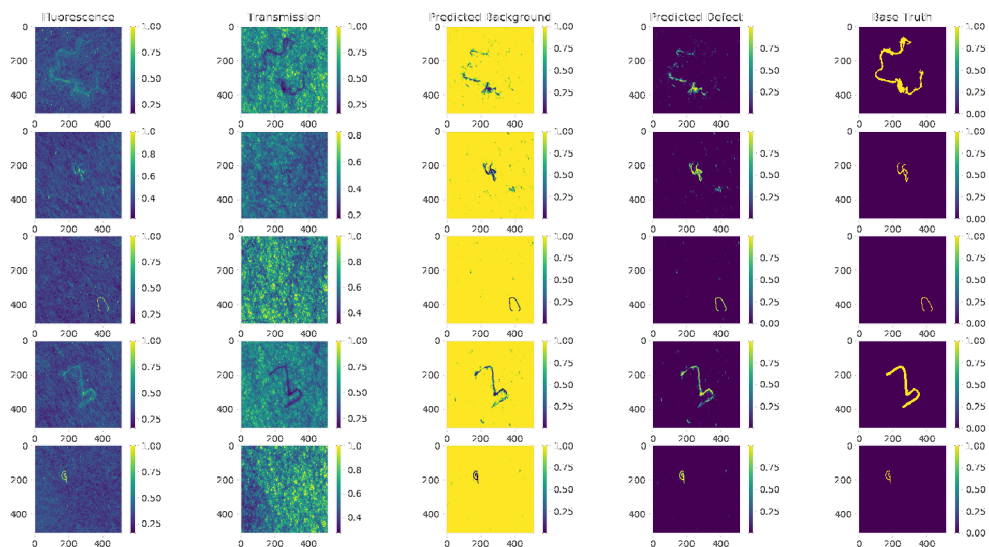


Figure 6.5: Prediction of the network. Larger version available as figure A.11.

CONCLUSION

The goal of this thesis was to create prototype of scanning bench for defect detection in SILON's quality check process of fibre material, to at least partially automate so far manual process.

Two scanners were built, providing large enough resolution for such task. Software tools were developed to control the scanner and to acquire images from it. It's possible to send commands to the drive and continuously scan the area using linescan camera.

Several tests were done to choose best lighting and filters to get the best image possible.

The fully convolutional network was chosen to automatically detect any defects in the fibre, such as bundles of thread or thick, brittle threads. The FCN was implemented in Python, using Tensorflow and Keras packages.

The functionality of the network was demonstrated on custom made dataset of fibre defects, proving at least partial ability to find defects and mark them.

Bibliography

- [1] <http://www.silon.eu/products/tesil/>. [2018-12-29].
- [2] <https://www.jobs.cz/fp/silon-s-r-o-12623876/>. [2019-05-01].
- [3] <http://www.silon.eu/silon-company/quality/laboratory-fibers/>. [2019-05-01].
- [4] <https://www.quora.com/How-does-VGG-network-architecture-work>. [2019-01-03].
- [5] <https://www.youtube.com/watch?v=nDPWywWRIRo>. [2019-01-03].
- [6] <https://www.tensorflow.org/>. [2019-01-01].
- [7] <https://github.com/Theano/Theano>. [2019-01-01].
- [8] <https://mxnet.apache.org/>. [2019-01-01].
- [9] <https://www.microsoft.com/en-us/cognitive-toolkit/>. [2019-01-01].
- [10] <https://github.com/plaidml>. [2019-01-01].
- [11] Eastman Chemical Company and Eastman Kodak Company. *Eastman Laboratory Chemicals: Catalog*. Number no. 55. Eastman Kodak Company, 1993.
- [12] Hector P. Martinez, Yoshua Bengio, and Georgios N. Yannakakis. Learning deep physiological models of affect. *IEEE Computational Intelligence Magazine*, 8(2):20–33, may 2013.
- [13] M.L. Minsky and S. Papert. *Perceptrons: An Introduction to Computational Geometry*. Mit Press, 1972.
- [14] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, pages 65–386, 1958.
- [15] Etan J. Tal. <https://commons.wikimedia.org/w/index.php?curid=51696280>. CC BY-SA 4.0. [2019-05-01].
- [16] P.J. Werbos. *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences*. Harvard University, 1975.

- [17] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

Appendices

Contents of included CD

-

Hardware

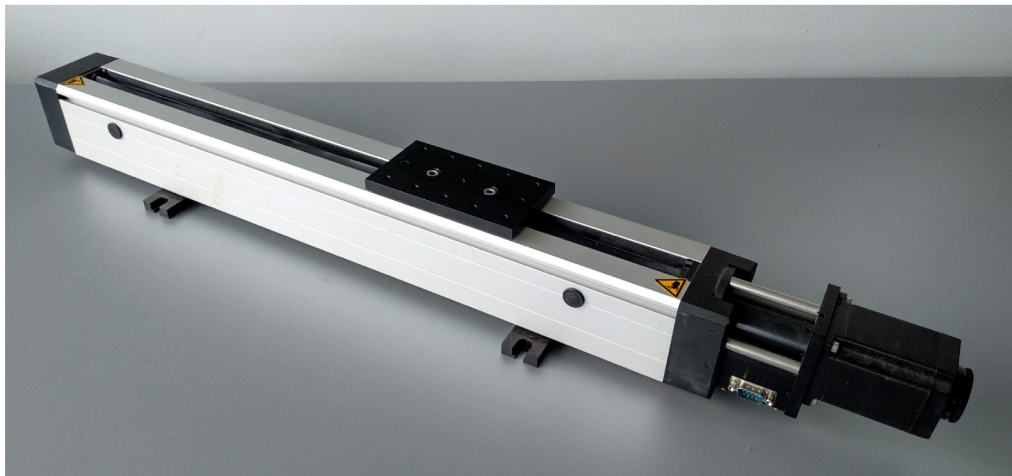


Figure A.1: Linear actuator by STANDA®.

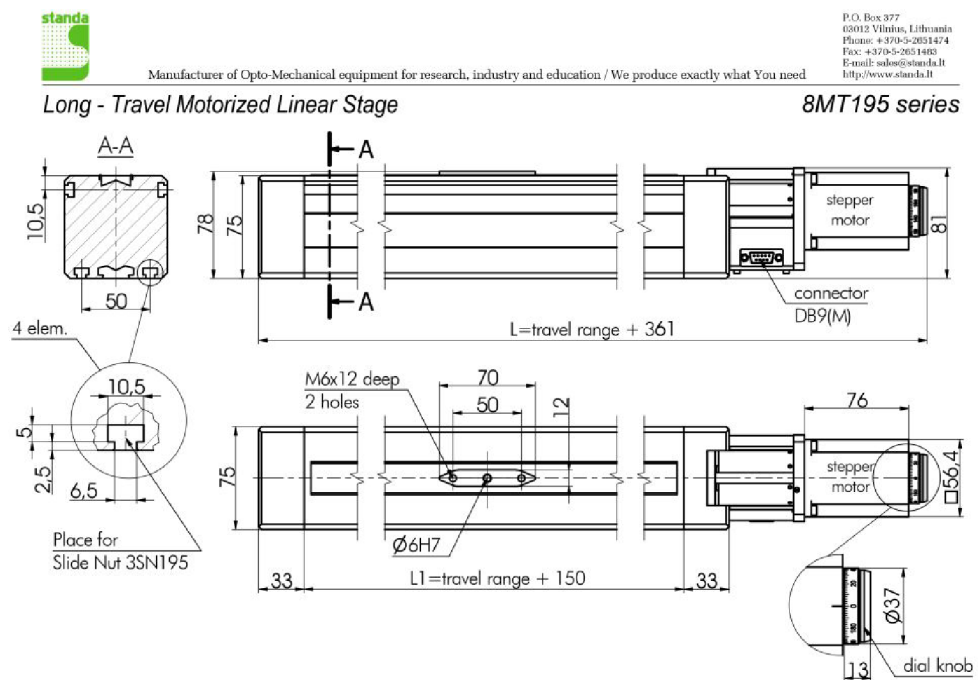


Figure A.2: Standa linear actuator schematics.

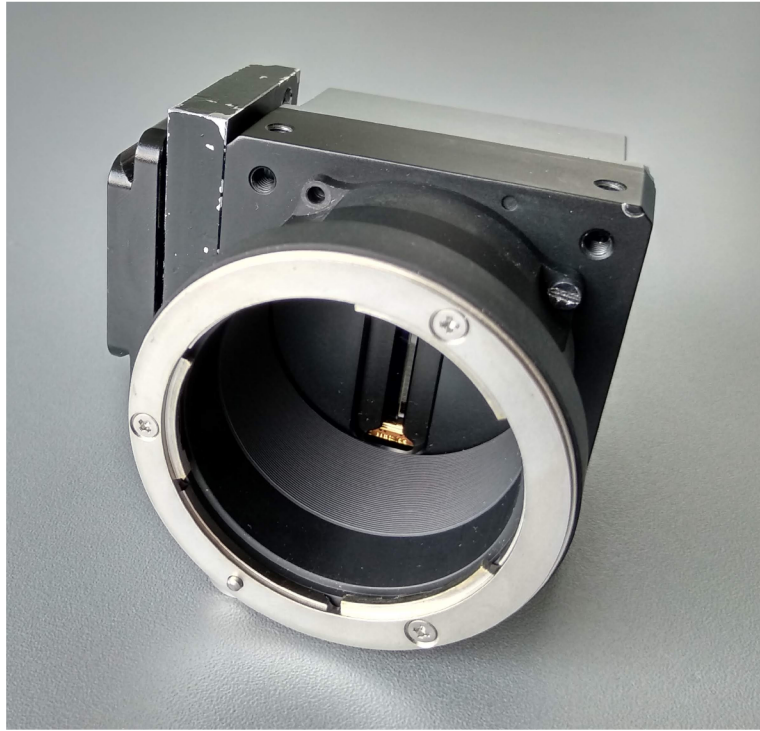


Figure A.3: Linescan camera – front.



Figure A.4: Linescan camera – back.



Figure A.5: Nikon lens 50 mm.



Figure A.6: Set of close-up lenses.



Figure A.7: Light filter temporarily fixed to the back side of a lens. Note professional attachment.

Software

List of linear drive commands

- Open(device)
- Close(device)
- Step(steps)
- Goto(left/right)
- Start_speed(speed)
- Max_speed(speed)

Motor control GUI by Standa

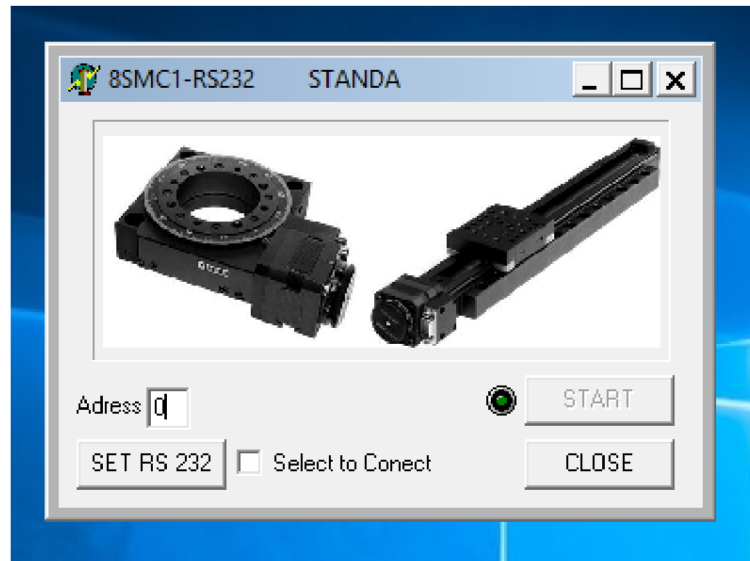


Figure A.8: COM port selection screen.

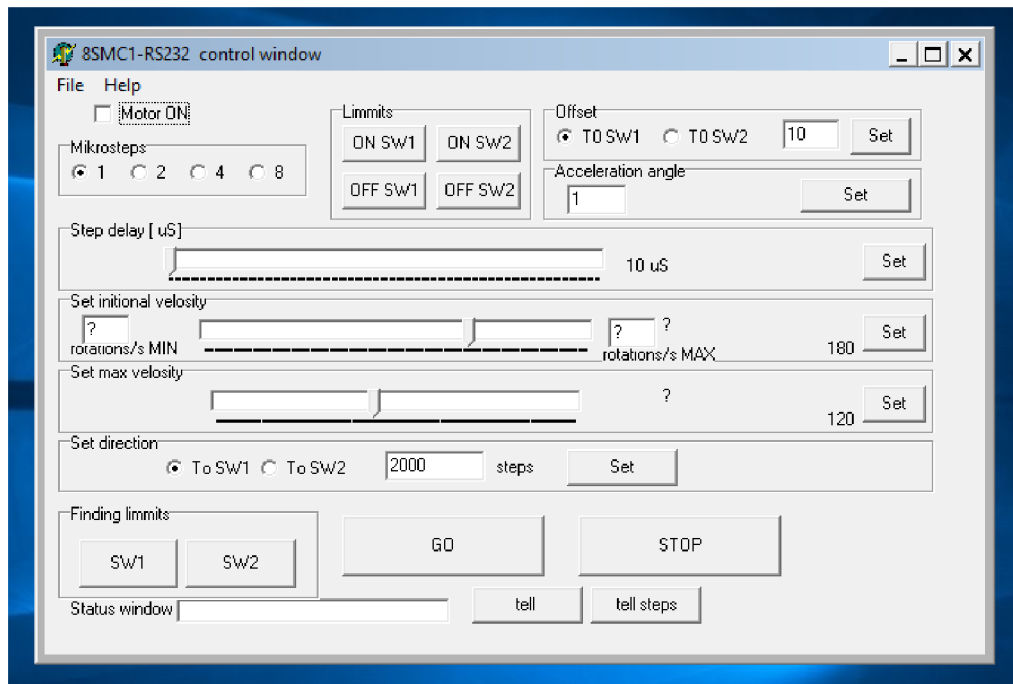


Figure A.9: Operating software of STANDA[®] motor.

Dataset

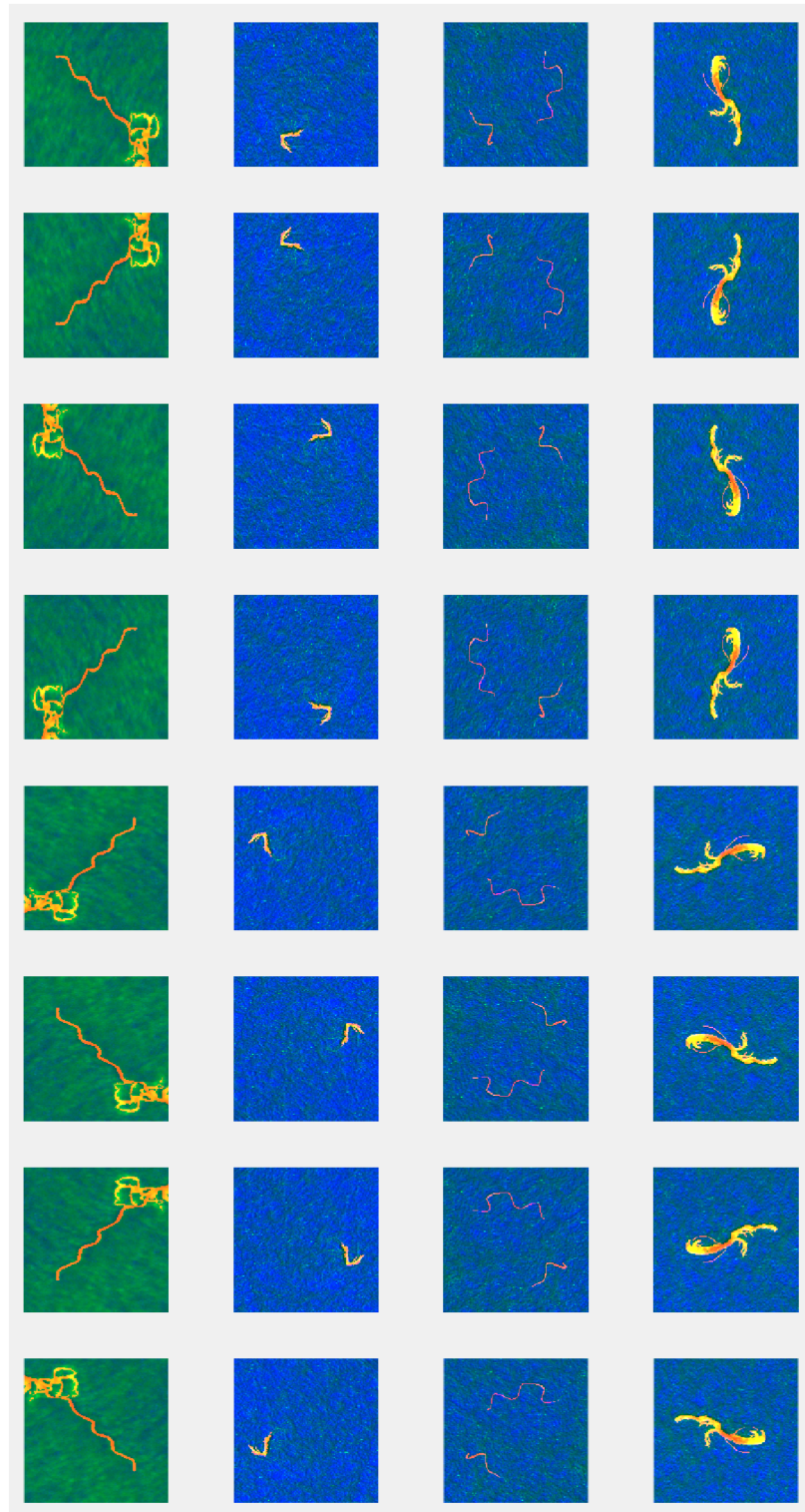


Figure A.10: Example of the training dataset. Red color is for true labels, green and blue are for fluorescence and transmission respectively.

Prediction

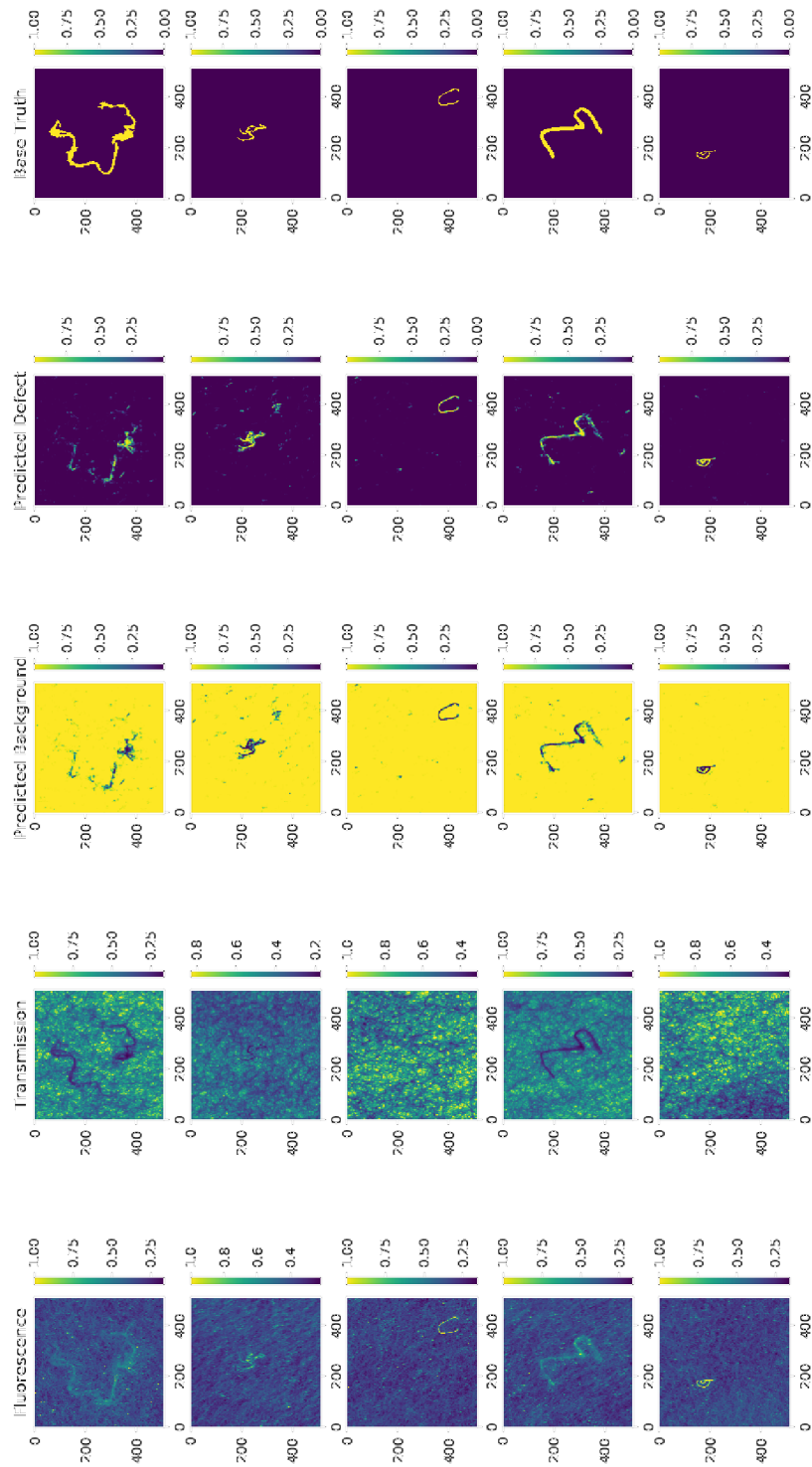


Figure A.11: Prediction of the network.

List of appendices

Contents of included CD	39
Hardware	39
Software	42