

JYU DISSERTATIONS 580

Anna-Maria Raita-Hakola

From sensors to machine vision systems

Exploring machine vision, computer vision
and machine learning with hyperspectral
imaging applications



UNIVERSITY OF JYVÄSKYLÄ
FACULTY OF INFORMATION
TECHNOLOGY

JYU DISSERTATIONS 580

Anna-Maria Raita-Hakola

From sensors to machine vision systems

**Exploring machine vision, computer vision and machine
learning with hyperspectral imaging applications**

Esitetään Jyväskylän yliopiston informaatioteknologian tiedekunnan suostumuksella
julkisesti tarkastettavaksi Agora-rakennuksen Lea Pulkkinen salissa (Ag B431.1)
joulukuun 2. päivänä 2022 kello 10.

Academic dissertation to be publicly discussed, by permission of
the Faculty of Information Technology of the University of Jyväskylä,
in building Agora, Lea Pulkkinen Hall (Ag B431.1) on December 2, 2022 at 10 a.m.



JYVÄSKYLÄN YLIOPISTO
UNIVERSITY OF JYVÄSKYLÄ

JYVÄSKYLÄ 2022

Editors

Marja-Leena Rantalainen

Faculty of Information Technology, University of Jyväskylä

Timo Hautala

Open Science Centre, University of Jyväskylä

Copyright © 2022, by University of Jyväskylä

ISBN 978-951-39-9240-8 (PDF)

URN:ISBN:978-951-39-9240-8

ISSN 2489-9003

Permanent link to the online version of this publication: <http://urn.fi/URN:ISBN:978-951-39-9240-8>

ABSTRACT

Raita-Hakola, Anna-Maria

From sensors to machine vision systems: Exploring machine vision, computer vision and machine learning with hyperspectral imaging applications

Jyväskylä: University of Jyväskylä, 2022, 132 p. (+included articles)

(JYU Dissertations

ISSN 2489-9003; 580)

ISBN 978-951-39-9240-8 (PDF)

The hypothesis of this study is “The machine vision systems should be designed, built and evaluated through the machine vision fundamental phases.” The dissertation defines the fundamentals inspired by the literature and shows how to use them and design a machine vision system from the sensor level to the analysis. The study covers the hardware and software designing phases, continuing to the data pre-processing, transformation and analysis phases.

The conducted research consists of six published articles. Three have a data-analytical point of view concentrating on developing and testing new versions of a distance-based machine learning method Minimal Learning Machine (MLM). In contrast, the rest of the articles introduces a novel concept of 3D hyperspectral imaging and convolutional neural networks for detecting, classifying and delineating skin cancer on complex skin surfaces. The skin cancer imager concept includes devices, user interfaces, pre-processing, transformation and analysis.

As the main results, we introduce the machine vision fundamentals with three effective variations from the MLM, which is suitable for hyperspectral imaging anomaly detection and classification tasks in real-time applications. Other outcomes were a novel hyperspectral imaging concept that can reach complex skin surfaces, opening the road for future optical biopsy. As skin cancers are the world’s third most common type of cancer, an optical biopsy can reduce diagnosis and treatment costs and save lives through early, accurate detection.

The results confirm that by developing machine vision systems according to the application and paying attention to the machine vision fundamentals, it is possible, for example, to influence the system’s computational complexity and improve the system’s results. By understanding the multi-directional relations between the fundamental phases of a machine vision system, we can affect the overall performance. For instance, by designing a machine vision system that collects only the necessary data and uses optimised fast computational methods, we can affect the system’s efficiency, energy need and operating costs.

Keywords: Machine vision fundamentals, machine vision, computer vision, machine learning, machine vision sensors, optical components, imaging, Minimal Learning Machine, convolutional neural network, hyperspectral imaging, hyperspectral imager, computational data-analysis, skin cancer research, optical biopsy, anomaly detection, classification

TIIVISTELMÄ (ABSTRACT IN FINNISH)

Raita-Hakola, Anna-Maria

Sensoreista konenäköjärjestelmiin: konenäön, tietokonenäön ja koneoppimisen tarkastelua hyperspektrikuvantamisen sovelluksissa

Jyväskylä: University of Jyväskylä, 2022, 132 s. (+artikkelitekijät)

(JYU Dissertations

ISSN 2489-9003; 580)

ISBN 978-951-39-9240-8 (PDF)

"Konenäköjärjestelmät tulisi suunnitella, rakentaa ja arvioida hyödyntäen konenäön perusteita." Väitöskirjassa määritellään konenäön perusteet, kattaa kaikki konenäköjärjestelmien vaiheet sensoritasolta kuvantamiseen, optisen kuvan esikäsittelyyn, kuvadataan muuntamiseen, analysointiin ja tuloksiin.

Tutkimus koostuu kuudesta julkaisusta artikkelistä. Kolmessa on laskennallisen data-analytiikan näkökulma, joka keskittyy kehittämään ja testaamaan uusia versioita etäisyyssperustaiseen koneoppimismenetelmään (Minimal Learning Machine (MLM)). Loput kolme artikkelia esittelevät ja testaavat uutta 3D hyperspektrikuvantamiseen ja konvoluutioneuroverkkoihin perustuvaa konseptia ihosyöpäkasvaimien tunnistamiseen, luokittelemiseen ja rajaamiseen, erityisesti haastavilla ihmipinnoilla kuvaten. Ihosyövän kuvantamiskonsepti esitellään yksityiskohtaisin tiedoin laitteesta, laiteohjauksesta, kuvantamislogiikasta, käytöliittymästä, kuvadataan esikäsittelystä, muuntamisesta ja analyysista tuloksineen.

Päätuloksenä esitellään konenäön perusteiden rinnalla kolme tehokasta muunnelmaa MLM:stä ja optiseen biopsiaan tähtäävä ihosyöpäkuvantamisen konsepti. MLM muunnelmat soveltuват esimerkiksi poikkeamien havaitsemiseen hyperspektrikuvista ja kuvien luokittelemiseen. Menetelmät ovat yksinkertaisia soveltuuen esim. reaalialaisiin sovelluskohteisiin. Ihosyöpäkuvantamiskonseptin tulosten kautta tasoitetaan tietä optiselle biopsialle. Ihosyöpien ollessa maailman kolmanneksi yleisimpiä syöpätyyppejä, optisen biopsian avulla voidaan tulevaisuudessa vähentää diagnoosi- ja hoitokustannuksia, sekä aikaisen hoidon myötä jopa säädää ihmishenkia.

Tulokset vahvistavat, että kehittämällä konenäköjärjestelmiä sovelluskohdaisesti konenäön perusteiden avulla räätälöiden, voidaan esimerkiksi vaikuttaa järjestelmän laskennalliseen monimutkaisuuteen ja parantaa järjestelmän kokonaistuloksia. Ymmärtämällä miten aiemmat vaiheet perusteissa liittyvät seuraavan vaiheen tuloksiin huomataan, että esimerkiksi konenäköjärjestelmä, joka kerää vain tarvittavat tiedot ja käyttää optimoituja nopeita menetelmiä, on tehokas laskennaltaan, energiantarpeeltaan ja lopulta myös käyttökustannuksiltaan.

Avainsanat: Konenäköjärjestelmien perusta, konenäkö, tietokonenäkö, koneoppiminen, konenäkösensori, kuvantaminen, minimaalinen oppimiskone, konvoluutioneuroverkko, hyperspektrikuvantaminen, laskennallinen data-analyysi, ihosyöpätutkimus, optinen biopsia

Author	M.Sc Anna-Maria Raita-Hakola Faculty of Information Technology University of Jyväskylä Finland
Supervisors	Associate prof. Dr. Ilkka Pölönen Faculty of Information Technology University of Jyväskylä Finland
	Prof. Dr. Timo Hämäläinen Faculty of Information Technology University of Jyväskylä Finland
Reviewers	Prof. Dr. Wen Li School of Computer science and engineering University of Electronic Science and Technology of China China
	Prof. Dr. (Tech) Lasse Lensu Computational Engineering LUT School of Engineering Science Finland
Opponent	Associate prof. Dr. Arto Klami Department of Computer Science University of Helsinki Finland

ACKNOWLEDGEMENTS

"If I have seen further it is by standing on the shoulders of Giants."

-Isaac Newton-

Newton could not have been more right with the thoughts of standing on the shoulders of Giants. Without the Giants I am lucky to have in my life, you would not be reading this dissertation. I would like to thank my colleagues (the Giants), current and past, who have been working at the Spectral imaging laboratory - or in our office. There are so many members at the spectral family, so just to name some of us, thank you Pauliina, Kimmo, Leevi I, Leevi II, Vilho, Anri, Jarno, Sampsaa, Samuli, Jaana, Severi, Salli and Tero. I probably forgot to mention someone. If that happened, please, fill in your name:

I am sorry, it was not intentional (I probably owe you a cup of coffee).

Paavo - thank you for introducing me to Ilkka - without one elevator conversation, I would not be writing this. Ilkka, thank you for believing I was good enough to be hired as a software developer a few years ago. I tried to show my thin programmer's biceps and said to you: "*I am not sure if I am a skilled enough developer.*" Nevertheless, you have believed in my skills and pushed me forward by teaching and encouraging me. Over the years, it has been quite a few days when I have felt like a bird in a sea filled with beautiful fish while trying to learn from my skilled colleagues; mathematicians, physicists, software professionals and biologists. Despite the self-doubts, Ilkka, you have explained maths and remembered to say that I am qualified and smart enough to do my job (and that sometimes, the fishes need friends with wings for flying). My second supervisor Timo, thank you for encouraging me and for all your feedback. You have been very prompt in answering my questions, and it has been nice to work with you.

This dissertation was mainly funded by working on interesting projects; thus, I would like to thank the Academy of Finland. Thank you, Emil Aaltonen Foundation for my young researcher's grant, IT Faculty for a device grant, and, latest, Nokia Foundation for the scholarship that allows me to publish the MLM methods.

Kiitos perheelleni, erityisesti Helmileille ja Tuomille, jotka tietävät mitä tarkoittaa hajamielinäinen äiti, joka ajattelee ja kirjoittaa. Kiitos äidilleni, isälleni ja veljellemi sekä ystävilleni.

Jyväskylässä 16.11.2022

Anna-Maria

LIST OF ACRONYMS

ANN	Artificial neural network
CCD	Semiconductor charge-coupled-device sensor
CFA	Color filter array
CMOS	Metal-oxide-semiconductor sensor
CNN	Convolutional neural network
CV	Computer vision
DN	Digital number
FPI	Fabry-Pérot interferometer
FWHM	Full width half maximum
HUS	Helsinki University Hospital
HS	Hyperspectral
HSI	Hyperspectral imager
LED	Light-emitting diode
MFPI	MOEMS based Fabry-Pérot interferometer
MLM	Minimal Learning Machine
ML	Machine learning
MOEMS	Microoptoelectromechanical system
MV	Machine vision
NN	Neural network
PCA	Principal component analysis
RF	Random forest
RGB	Red-green-blue (colour model, colour space)
VTT	Technical Research Centre of Finland Ltd

LIST OF FIGURES

FIGURE 1	Monochromatic, RGB, spectroscopy, multispectral and HS features.	20
FIGURE 2	Preview of an HS image.....	21
FIGURE 3	Deep and overlapping terminological relationships	23
FIGURE 4	Fundamentals of machine vision	24
FIGURE 5	Fundamentals of computer vision.....	25
FIGURE 6	The fundamental processes of machine vision	30
FIGURE 7	Image capturing processes and related process phases	31
FIGURE 8	A simplified physical model of a pixel and imaging sensor.	34
FIGURE 9	Full well capacity.....	35
FIGURE 10	The Bayer BGGR pattern filter.....	37
FIGURE 11	Bilinear convolution.	38
FIGURE 12	Bilinear interpolation.....	38
FIGURE 13	Forming an optical image.	41
FIGURE 14	Image vignetting and shading.....	42
FIGURE 15	Long and short focal length.....	43
FIGURE 16	Field of view	44
FIGURE 17	An example of illumination in spectral imaging.....	47
FIGURE 18	HSI types and scanning techniques.....	48
FIGURE 19	Operation principles of the Fabry-Pérot interferometer.	50
FIGURE 20	Fundamentals of computer vision.....	52
FIGURE 21	Lambertian surface and hemispherical reflectance.....	54
FIGURE 22	Bi-directional reflectance and transmittance imaging.....	55
FIGURE 23	Examples of a hemispherical reflectance.....	56
FIGURE 24	Examples of transmittance and absorbance spectral data.	57
FIGURE 25	Effect of centring and projecting data.....	61
FIGURE 26	A simple tree structure formed with a branching algorithm.	64
FIGURE 27	SVM and kernel SVM variations.....	65
FIGURE 28	Max Pooling	68
FIGURE 29	Perceptron, a single computation unit of a neural network.	68
FIGURE 30	An example of a multilayer perceptron (ANN).	69
FIGURE 31	The convolution operation.....	70
FIGURE 32	3D CNN.....	71
FIGURE 33	Minimal learning Machine.....	73
FIGURE 34	Classifying data points with minimal learning machine.....	73
FIGURE 35	SICSURFIS instrumentation.	75
FIGURE 36	SICSURFIS imager modules.....	76
FIGURE 37	SICSURFIS imager's LED module and imaging setup parameters.....	76
FIGURE 38	Opening the software.	79
FIGURE 39	The patient information.....	80
FIGURE 40	Capturing the dark reference.....	81

FIGURE 41	Capturing the white reference.....	82
FIGURE 42	White reference's hidden device-level actions.....	82
FIGURE 43	Capturing the spectral data.....	83
FIGURE 44	Hidden device level actions	83
FIGURE 45	User action "save"	84
FIGURE 46	User action; dismiss	84
FIGURE 47	Pavia dataset's geometry	91
FIGURE 48	The piecewise MLM re-trains	94
FIGURE 49	Joint anomaly detection maps	96
FIGURE 50	Comparison of the ROC-curves.....	96
FIGURE 51	New data points	100
FIGURE 52	RLS update.....	100
FIGURE 53	Classification rules for the new classes.....	101
FIGURE 54	Classification and error maps of Experiment B.....	102
FIGURE 55	Visualisation of the self-learning MLM's	102
FIGURE 56	Pre-processing phases of the SICSRFIS 3D data.....	105
FIGURE 57	Transformation phases of the SICSRFIS 3D data.....	106
FIGURE 58	The pixel selection.....	106
FIGURE 59	SICSRFIS CNN.....	107
FIGURE 60	An example prediction map.....	108
FIGURE 61	An example of the acquisition-related challenge.....	108
FIGURE 62	Pixel selection improvement.....	109
FIGURE 63	Prediction map of a pigmented nevus.....	110
FIGURE 64	CubeView software	115
FIGURE 65	The relationship between the articles and machine vision fundamentals.....	120
FIGURE 66	Future work	122

LIST OF TABLES

TABLE 1	Commonly used colour definitions.....	21
TABLE 2	Spectral sub-divisions	22
TABLE 3	Research sub-questions and related articles	28
TABLE 4	The GenICam standard modules.....	33
TABLE 5	Optics terms and definitions	40
TABLE 6	Research questions, related articles and sections with topics	86
TABLE 7	Review process details	87
TABLE 8	Comparison of the computation times for the ColorChecker dataset	95
TABLE 9	Classification report of the pixel classification, slice-half model.	107
TABLE 10	Leave-one-out (L-O-O) and slice half pixel classification accuracies	111

LIST OF ALGORITHMS

1	PCA with HS data	60
2	MLM training	72
3	MLM prediction	72
4	The PC-MLM algorithm	90
5	The piecewise MLM training	95
6	Anomaly detection using the piecewise MLM model	95
7	Recursive Least Squares (Romberg 2016)	100

CONTENTS

ABSTRACT

TIIVISTELMÄ (ABSTRACT IN FINNISH)

ACKNOWLEDGEMENTS

LIST OF ACRONYMS

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

CONTENTS

LIST OF INCLUDED ARTICLES

1	INTRODUCTION	17
1.1	The short history of machine vision.....	18
1.2	Hyperspectral imaging	20
1.3	Machine vision terminology	23
1.4	Motivation.....	24
1.5	Hypothesis and research questions.....	27
1.6	Author's other works.....	29
1.7	Dissertation structure and common acronyms	29
2	MACHINE VISION FUNDAMENTALS	30
2.1	Image capturing	31
2.1.1	Standards.....	31
2.1.2	Machine vision sensors	33
2.1.3	Optical and optomechanical components.....	39
2.1.4	Illumination	45
2.1.5	Controlling devices with Python	45
2.1.6	Imaging setups.....	46
2.1.7	Spectral imagers	47
2.1.8	Piezo-actuated metallic mirror Fabry-Pérot interferometer.	50
2.1.9	Summary: Image capturing process.....	51
2.2	Computer vision: Image pre-processing	52
2.2.1	Spectral imaging and image processing	52
2.2.2	From raw image to spectral data	53
2.2.3	Spectral data examples.....	56
2.3	Computer vision: Machine learning	58
2.3.1	Exploring the structure and internal organisation of the data.....	58
2.3.2	Principal component analysis	59
2.3.3	Computer vision's main tasks and machine learning terminology	61
2.3.4	Random forest.....	63
2.3.5	Support vector machine	64
2.3.6	Artificial neural networks.....	65

2.3.7	Convolutional neural network	69
2.3.8	Minimal learning machine.....	72
2.4	Machine vision: Imaging systems.....	74
2.4.1	Design principles, environmental issues and user needs....	74
2.4.2	Instrumentation.....	74
2.4.3	Imaging processes and HSI's device modules	75
2.4.4	Device workflow and user interface	77
2.4.5	Summary: machine vision fundamentals	85
3	EXPERIMENTS AND RESULTS	86
3.1	Review processes.....	86
3.2	Developing methods for computer vision analysis.....	88
3.2.1	Effective classification method for hyperspectral imaging..	88
3.2.2	Minimalistic approach to hyperspectral image anomaly detection	93
3.2.3	Self-learning model for classification	99
3.3	From sensors to analysis.....	103
3.3.1	3D Spectral imaging system for complex skin surfaces - toward optical biopsy	104
3.3.2	Combining new imaging technology with user-friendly imaging and analysis software	113
4	DISCUSSION AND CONCLUSION.....	116
4.1	Application domains, inspiration and importance	116
4.2	Machine vision fundamentals and research questions	117
4.3	Conclusions	120
4.4	Future work	122
	YHTEENVETO (SUMMARY IN FINNISH)	123
	REFERENCES.....	124
	INCLUDED ARTICLES	

LIST OF INCLUDED ARTICLES

- PI Hakola Anna-Maria and Pölönen Ilkka . Minimal Learning Machine in Hyperspectral imaging classification. *SPIE - Society of Photo-Optical Instrumentation Engineers, Proceedings Volume 11533, Image and Signal Processing for Remote Sensing XXVI; 115330R*, <https://doi.org/10.1117/12.2573578>, 2020.
- PII Raita-Hakola Anna-Maria and Pölönen Ilkka . Piecewise Anomaly Detection Using Minimal Learning Machine for Hyperspectral Images. *ISPRS - Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, <https://doi.org/10.5194/isprs-annals-V-3-2021-89-2021>, 2021.
- PIII Raita-Hakola Anna-Maria and Pölönen Ilkka . Updating Strategies for Distance Based Classification Model with Recursive Least Squares. *ISPRS - Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, <https://doi.org/10.5194/isprs-annals-V-3-2022-163-2022>, 2022.
- PIV Raita-Hakola Anna-Maria, Annala Leevi, Lindholm Vivian, Trops Roberts, Näsilä Antti, Saari Heikki, Ranki Annamari and Pölönen Ilkka. FPI Based Hyperspectral Imager for Complex Skin Surfaces - Calibration, illumination and applications. *MDPI Sensors, Volume 22, Number 9, Pages 3420*, <https://doi.org/10.3390/s22093420>, 2022.
- PV Lindholm Vivian, Raita-Hakola Anna-Maria, Salmivuori Mari, Jeskanen Leila, Saari Heikki, Koskenmies Sari, Pitkänen Sari, Pölönen Ilkka, Isoherranen Kirsi and Ranki Annamari. Differanting Malignant from Bening Pigmented or Non-pigmented Skin Tumors - a Pilot Study on 3D Hyperspectral Imaging and convolutional neural networks. *MDPI - Journal of Clinical Medicine, Volume 11, Number 7, Pages 1914*, <https://doi.org/10.3390/jcm11071914>, 2022.
- PVI Trops Roberts, Hakola Anna-Maria, Jääskeläinen Severi, Näsilä Antti, Leevi Annala, Eskelinen Matti A., Saari Heikki, Pölönen Ilkka and Rissanen Anna. Miniature MOEMS Hyperspectral Imager with versatile analysis tools. *SPIE - Society of Photo-Optical Instrumentation Engineers, Proceedings Volume 10931, MOEMS and Miniaturized Systems XVIII; 109310W*, <https://doi.org/10.1117/12.2506366>, 2019.

Author's contribution

Articles PI, PII and PIII:

The author developed the methods together with the second author. She conducted the research from the Python code to article writing and conference presentations. The second author taught, supervised and guided her during the phases of the research process.

Articles PIV and PV:

The author designed and developed the device control flows and imaging logic, resulting in a CubeView hospital user interface. She participated in designing the stray light protection cones and the imaging trolley where the system was mounted. Her colleague did the image pre-processing from raw data to albedo and 3D cubes. She continued the work from the data transformation phases into analysis producing the results with Python code. She was the main writer at article PIV and had equal first authorship on PV (technical aspects).

Article PVI:

The author designed, developed and implemented the CubeView user interface with Python and wrote the according subsection into the article.

1 INTRODUCTION

Just add light, a machine vision sensor and some optical components. Capture a non-invasive signal, process it and visualise the phenomenon. You might catch something vital from a point of view that is hidden from your naked eyes. This something can hide beneath the stroke of an artist's brush, suddenly multiplying the value of a painting. It could be something malignant or benign growing under your skin; it could be a lesion, a tumour or even a problem with your teeth. Expand the view outside, let the view grow from the cellular level to space, exploring water, food, fruits, trees or asteroids. These all are examples of machine vision applications introduced in recent research, which combine hyperspectral imaging with machine learning models.

The aforementioned applications and beyond, despite the level of complexity, technical speciality or challenges, can be implemented using the fundamentals of machine vision. The question to wonder is what are the fundamentals of machine vision systems and how those fundamental phases can be utilised, together or separately while designing and implementing a simple research application or challenging, complex applications. In the following subsections, we will travel through a short historical overview and the basis of hyperspectral imaging. After defining the terminology, we will discuss the motivation and aims of the study with the hypothesis and research questions (Section 1.5). We will end the introduction with the author's other works and a description of the dissertation's structure.

This dissertation is about the journey of a beam of light, captured with a sensor, to the final application.

1.1 The short history of machine vision

For someone, an image tells more than a thousand words. For a scientist or machine vision system developer, an image, besides the words, is a two-dimensional array, a function $f(x, y)$, where the spatial coordinates (dimensions) are x and y . Each data point (a pixel) is a pair of coordinates (x, y) , which represent the intensity of the grey level at that specific data point. When x, y and the intensity values of f are finite, discrete quantities, we have a *digital image* (Gonzalez and Woods 2007).

The origin of digital image processing can be tracked back to the 1920s (Ren et al. 2021). One of the first applications involved newspaper images sent via submarine cables across the Atlantic between London and New York. However, the technology was not computer-based. In the context of computer vision, we must travel the time until John von Neumann introduced the concepts behind central processing units (CPUs) in the 1940s. After that, a series of innovations, from transistors to programming languages and microprocessors, were introduced, which formed the basis for computers. Finally, in the 1960s, when the first computers powerful enough for image processing were invented, we saw the early days of computer vision (Gonzalez and Woods 2007; Smith et al. 2021).

In the 1970s, a Japanese *intelligent robot* that was able to follow instructions and build shapes from building blocks. It was equipped with two cameras: one camera captured the drawing plan, while the other looked at the blocks' instead. The images were analysed with a computer, and the analysis was used to control a robotic arm (Ejiri 2007). Despite the significant problems with the robot's weak computational capacity, which was seen in an extremely slow image processing (Ejiri 2007), the potential of computer vision was noticed. Since the early days, data capturing and image analysis methods have continuously been under major development (Smith et al. 2021). However, within the first two decades, the development was relatively slow due to the intrinsic difficulties and the limitations of computing power (Ejiri 2007).

The early applications (e.g., intelligent robot (Ejiri et al. 1972) utilised vidicon imaging tubes (Takemura 2019). Vidicons were replaced after the semiconductor charge-coupled-device (CCD) and, subsequently, the metal-oxide-semiconductor (CMOS) sensors were improved; the first portable snapshot CCD imager was introduced in the 1970s (Takemura 2019). The switch from vidicon tubes to CCDs in the middle of the 1980s enabled remarkably improved performance, providing a promising technological foundation for the today's high-quality imaging devices (Takemura 2019). Starting from the 1980s, the increasing computational capacity along with improved camera and lighting technologies enabled industrial-grade cameras, light-emitting diode (LED) lighting and microprocessors, which led to the beginning of the era of automated industrial inspection (Smith et al. 2021).

Currently, there is a wide coverage over automated industrial tasks, among target size and position inspections, sorting, surface safety and defect, contami-

nation, colour and texture control. These tasks may require strict environmental control (Smith et al. 2021; Ren et al. 2021), which needs designing and decisions regarding the choice of devices and components to the image processing and analysis to be performed (Ren et al. 2021). The difficulty level of industrial activities' environmental control varies from easy to impossible, but the structuring is usually relatively easy to accomplish. For example, external ambient lighting can be controlled, and the vision system can be integrated into industrial robots or the production line (Smith et al. 2021). In the 2020s, the field of classic machine vision, combined with fast GPU-equipped computing, has reached a point where previously extremely challenging or even impossible industrial tasks can be automated (Smith et al. 2021). Therefore, the current machine vision technologies provide a promising foundation for intelligent manufacturing; by improving detection efficiency and accuracy, degree of automation and reducing the workforce, the methods can be especially beneficial for large-scale repetitive industrial production processes (Ren et al. 2021).

Since the early days, the goal of machine vision has been to replace human eyes for measurement and judgement. Today, in many cases, machine vision systems are outperforming human vision. For example, a machine vision system can cover a wider range of spectral responses than a human eye or be able to work long hours even in harsh environments (Ren et al. 2021). Side-by-side with device-level development, the field of computer vision has been growing and improving. The history seems to be weighted with industrial applications. Still, in the current state, machine vision systems are widely applied and studied in wider fields, including, e.g. mobile devices, self-driving cars, robots, medical applications, and aerial and military equipment.

The recent development of machine learning, which can be seen as the core of computer vision, has revolutionised the discipline (Smith et al. 2021). Especially artificial intelligence has transferred machine vision capabilities closer to human vision and, in certain applications, such as hyperspectral imaging, beyond it.

1.2 Hyperspectral imaging

Hyperspectral (HS) imaging is a non-invasive imaging technology that can reveal phenomena invisible to human vision capability. Spectral analysis aims to analyse the substances based on how their properties absorb or reflect light. Many substances or materials can be identified by their unique spectral signature, which is determined by how the substances reflect differently different wavebands of light. HS data is typically highly dimensional, which provides high accuracy and robustness for characterisation and identification tasks (Camps-Valls and Bruzzone 2005; Bioucas-Dias et al. 2013).

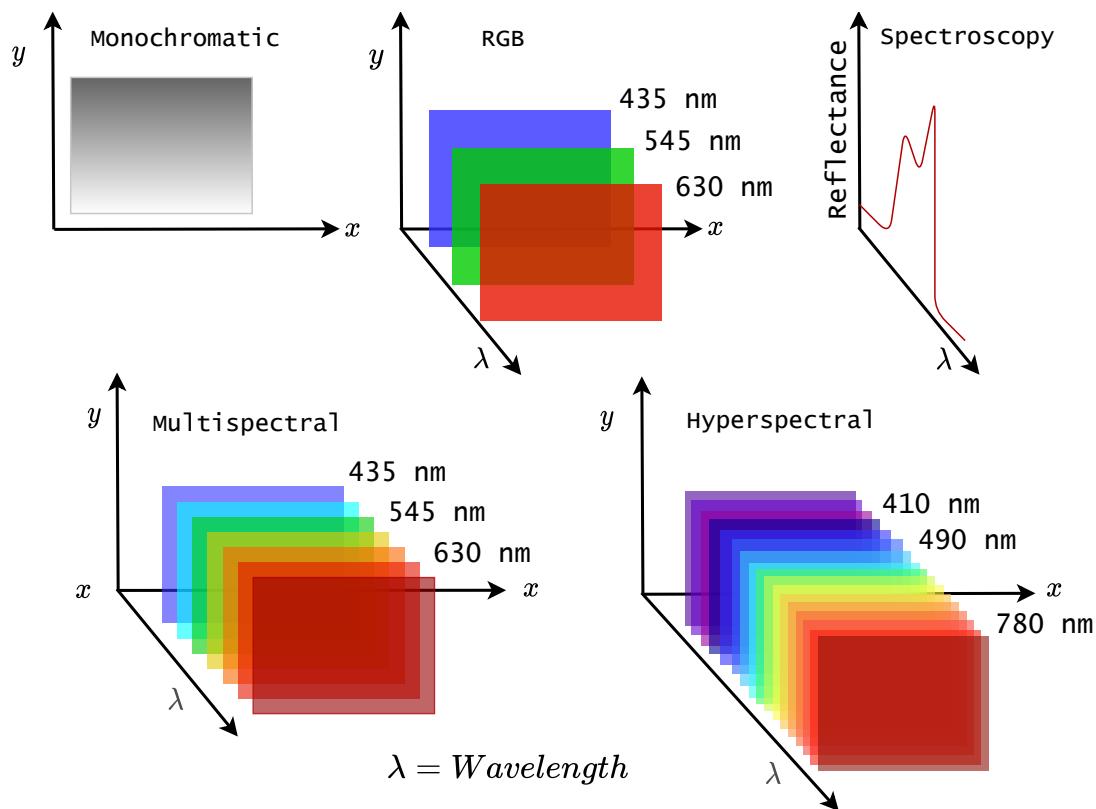


FIGURE 1 Monochromatic, RGB, spectroscopy, multispectral and HS features. Monochromatic images have one channel with spatial x and y dimensions. RGB images are constructed from three colour channels (spectral dimension λ) and spatial x and y dimensions. Spectroscopy measures dozens to hundreds of spectral channels, but the spatial dimensions are limited to one pixel. Multispectral and HS images have spatial and spectral dimensions (x , y and λ). Multispectral image typically has three to ten frames, and the spectral channel is wider than with HS images. HS images provide an almost continuous spectrum, which is constructed from hundreds or even thousands of narrow spectral channels.

HS imaging extends traditional photography by measuring electromagnetic radiation typically from visible (VIS) and near-infrared (NIR) light (below, Tables 1 and 2). In contrast with to RGB images' red, green and blue frames, an HS image can be considered a data cube of tens, hundreds or thousands of frames (spectral

bands or channels), each representing the intensity of a different wavelength of light, as seen in above, Figure 1 (Li et al. 2013).

TABLE 1 Commonly used colour definitions. The visible spectral range for the human eye is approximately from 380 nm to 740 nm.

Colour	Spectral range
Violet	380-430 nm
Blue	430-500 nm
Cyan	500-520 nm
Green	520-565 nm
Yellow	565-590 nm
Orange	590-625 nm
Red	625-740 nm

The power of spectral imaging lies in the detailed pixel-wise information. Since substances emit electromagnetic radiation, the intensity distribution at different wavelengths forms the radiation spectrum. The spectrum is continuous if no clear boundaries exist between the wavelength ranges. For example, the spectrum of the light from the sun or an incandescent lamp is continuous, while the light from a fluorescent lamp is a discontinuous line spectrum. Because each substance has its characteristic line spectrum, it is identifiable. By looking at the spectrum in Figure 2 (below), higher peaks can be observed, and their position on the wavelength axis and intensity can be compared with the known spectra. For example, the real plant in Figure 2 can be distinguished from artificial plants based on its spectrum.

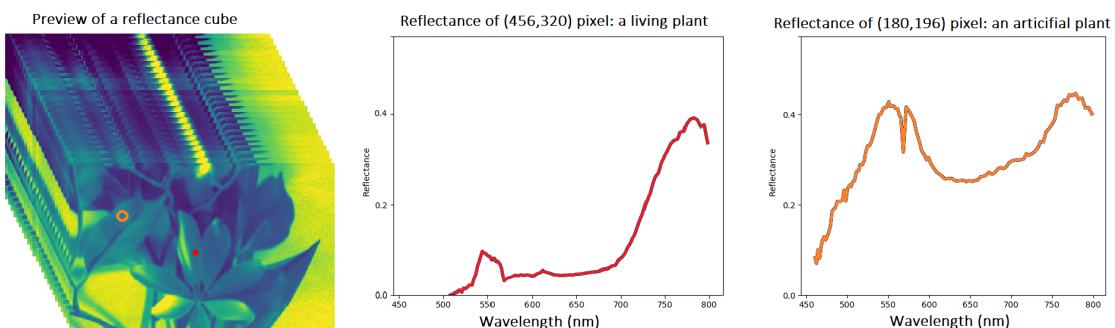


FIGURE 2 Preview of an HS image. Red dot and spectrum: a living plant; orange circle and spectrum: an artificial plant. Left: the preview of an HS image, middle: reference spectra of a living plant, right: an example spectrum of an artificial plant.

Figure 2 is a preview of an HS image. The red spot in the figure represents a selected spatial pixel, and the spectrum is visualised on the right. Each pixel has its own spectrum, so the HS image contains spatial and spectral domains, which enable, for example, accurate pixel-wise classification (Ahmad et al. 2022). The common definitions and spectral sub divisions and range can be seen in following Table 2.

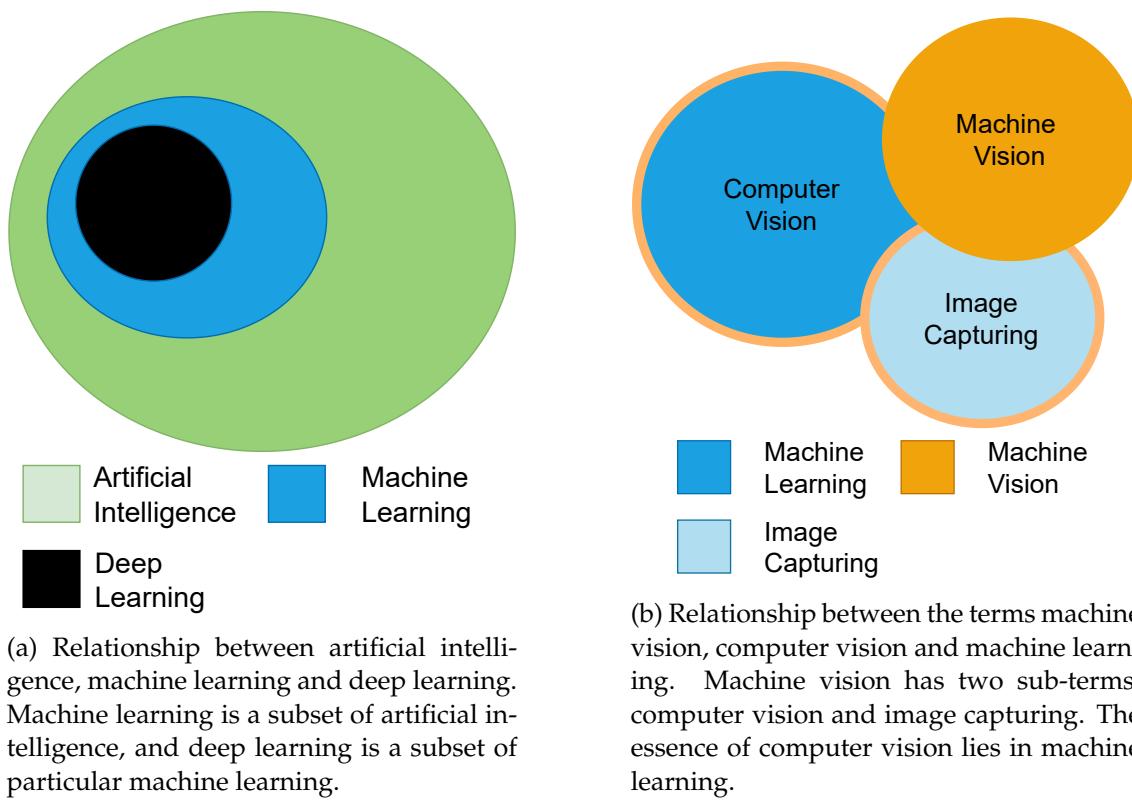
TABLE 2 **Spectral sub-divisions** with abbreviations and spectral wavelength range.

Name	Spectral range	Abbr.
Ultraviolet C	200-280 nm	UV-C
Ultraviolet B	280-315 nm	UV-B
Ultraviolet A	315-400 nm	UV-A
Visible light	380-740 nm	VIS
Near-infrared	740 -1400 nm	NIR
Short-wavelength infrared	1400-3000 nm	SWIR
Mid-wavelength infrared	3000-8000 nm	MWIR
Long-wavelength infrared "Thermal imaging"	8000-15000 nm	LWIR

The history of HS imaging consists of earth observation and remote sensing applications. After developments in sensor technology led to a reduction in imagers' physical sizes and made them more affordable (especially in VIS and VNIR range), this non-invasive method gained interest and yielded promising results in many other application fields. Some examples current applications are related to the domains of agriculture (Thorp et al. 2017), forestry (Adão et al. 2017), medicine (Fei 2020), mining (Krupnik and Khan 2019), biology (Salmi et al. 2022), food industry (Pathmanaban et al. 2019), space (Lind et al. 2021) and defence (Makki et al. 2017).

1.3 Machine vision terminology

The first high-level term in this field was *computer vision*. Since the core of the 1970s *intelligent robot* was vision, the research area was named after it (Ejiri 2007). In terms of research content, however, the case and name refers more to the current interpreter of machine vision. According to Smith et al. (2021), the current de facto interpretation of machine vision is "*computer vision techniques to help solve practical industrial problems that involve a significant visual component*". The state-of-the-art interpretation combines machine vision and deep learning methods and considers machine vision as one of the core technologies of artificial intelligence (Smith et al. 2021).



(a) Relationship between artificial intelligence, machine learning and deep learning. Machine learning is a subset of artificial intelligence, and deep learning is a subset of particular machine learning.

(b) Relationship between the terms machine vision, computer vision and machine learning. Machine vision has two sub-terms; computer vision and image capturing. The essence of computer vision lies in machine learning.

FIGURE 3 Deep and overlapping terminological relationships between artificial intelligence, machine learning, deep learning, machine vision and computer vision. (Original image source: Smith et al. (2021)).

Above, Figure 3 illustrates the relationships and overlapping terminology of the research fields, which form the basis of machine vision. Figure 3a shows how deep learning is a part of machine learning, and machine learning can be seen as a subset of artificial intelligence. Term machine vision includes computer vision based on machine learning (above, Figure 3b). The role of machine vision as a core technology of artificial intelligence has undergone revolutionary developments in recent years (Smith et al. 2021), which may partly explain why the term *machine vision* can be misused or confused with *computer vision*. In the era

of big data, the Internet provides plenty of databases and collections of different types of imaging data. Thus, machine learning models for image analysis can be developed and examined without the data capturing phases. As can be seen in examples (Geirhos et al. 2021; Jia et al. 2013), the term machine vision can be found even in scientific contexts (or in everyday language) used as a term for image data processing, which is the essence of computer vision. The definition of machine vision involves image capturing and image processing phases as seen below in the machine vision fundamentals (Figure 4).

At this dissertation, we will define the terms according to (Uma and Sukanya 2021) and (Smith et al. 2021). *Computer vision* covers image processing phases gaining a deep understanding of the material provided by machine vision's image acquisition phase. In contrast *machine vision* is a higher level term that describes all the fundamental phases of image capturing and computer vision. The fundamentals are visualised in Figure 4, which was inspired by the literature (e.g., Uma and Sukanya (2021); Smith et al. (2021) and Usama et al. (1996)). Figure 4 summarises the terminology and visualises the foundation of the machine vision fundamentals, which is the meta-theme of this dissertation. A machine vision based imaging application is introduced in Section 2.4. The imaging systems in articles PVI, PIV and PV are defined as *machine vision systems* and the articles PI, PII and PIII are related to computer vision.

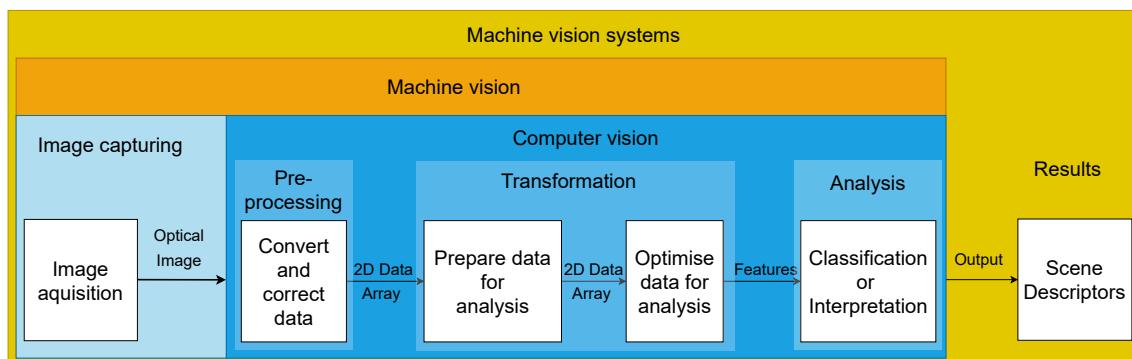


FIGURE 4 Fundamentals of machine vision

1.4 Motivation

A wide range of pre-processing, transformation and analysis methods exist for image data (Uma and Sukanya 2021; Iqbal et al. 2018). Thus, while the methods might vary, the process remains similar. The scene descriptors (e.g., classification results, output, above in Figure 4) are obtained using a multi-phase computer vision process, which is visualised below in Figure 5. The acquired optical images will be subjected to the major phases of pre-processing, transformation and analysis (Iqbal et al. 2018; Tian et al. 2021). For example, if the aim is to detect citrus lesions, the scale of detection methods varies among edge detection, watershed,

clustering, saliency, active contour, and thresholding (Iqbal et al. 2018).

Depending on the application's level of autonomy, the machine vision system may deliver information to the user or execute decisions. For example, the system can provide medical analysis results (Esteva et al. 2021), or the machine vision system can be adapted to an agricultural robot or an autonomous vehicle, that executes navigation decisions based on its machine vision system's outputs (Tian et al. 2021). Regardless of the use case, the machine vision systems should be developed accurately to serve the required purpose.

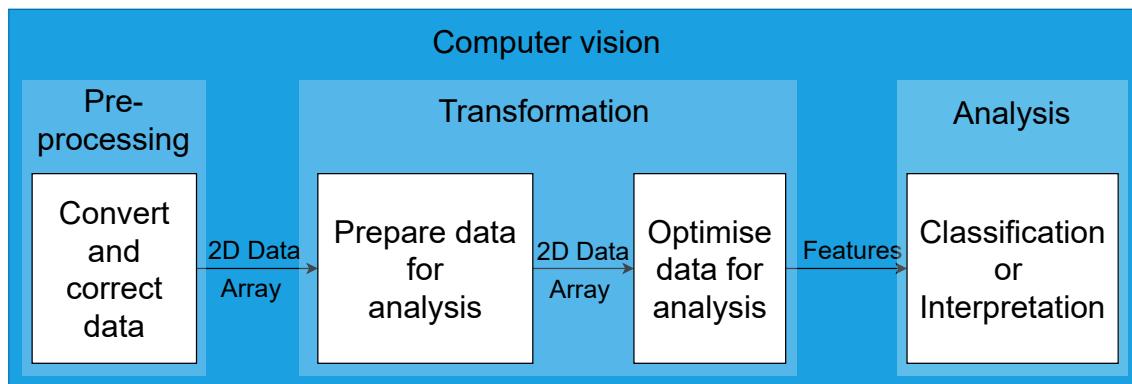


FIGURE 5 Fundamentals of computer vision. Before the results are obtained, the image data is typically processed in several phases.

The machine vision system's performance should be inspected throughout all the fundamental phases. Every phase, from image acquisition to the outputs, affects the other. According to Ren et al. (2021), the main evaluation indexes of a machine vision system are its accuracy (correct, true positives and true negatives), efficiency and robustness. Iqbal et al. (2018) extends the performance measures to cover sensitivity (true positive rate), precision rate, the area under the curve (AUC) and time, which is one of the crucial factors in real-time applications.

A pre-processing phase might include several computational methods, from converting an optical image to Red-Green-Blue (RGB) or some other colour space, algorithms for interpolation, colour enhancing, contrast correction, denoising, etc. The goal is to pre-process the image data into the appropriate form for the second phase, which is transformation (Iqbal et al. 2018). Correctly pre-processed image data increases the accuracy of the second phase. For example, a poor background might decrease the segmentation accuracy, which affects the final classification. The typical pre-processing challenges to outcome are noisy background, contrast between the target and background or low hue between colours. Other examples include variation and illumination changes, low-intensity values, and correlated redundancy (Iqbal et al. 2018; Ren et al. 2021; Tian et al. 2021). As the number of computational operations increases while each pre-processing phase is computed, the accuracy might increase but at the cost of time consumption and computational efficiency. According to Ren et al. (2021), the real-time processing speed is typically the bottleneck of a machine vision system.

The transformation challenges are generally related to the pre-processing

and image acquiring phases (Tian et al. 2021; Iqbal et al. 2018). Segmentation and threshold methods can be used to select the area of interest to emphasise certain target features (Ren et al. 2021; Tian et al. 2021) or decrease the computational intensity. According to Meribout et al. (2022), if the following analysis method utilises deep learning, the region of interest can be used for a partial image instead of computing those algorithms through all of the captured data. This approach enables deep learning and other computationally challenging methods to be applied closer to the image source (i.e., edge computing) instead of using cloud computing.

Besides the pre-processing, transformation and analysis challenges, the first phase of machine vision fundamentals extensively impacts the system's overall performance. As an illustration of the image acquisition phase, agricultural targets, such as plant factories, are good examples of challenging illumination. According to Tian et al. (2021), plants usually grow in an illumination of 660 nm and 460 nm, which causes difficulties for segmentation methods. Compared to images captured in an illumination containing more wavelengths of light, the plants and canopy are difficult to extract from the white background. The quality of the acquired optical images should be high, and, in applications that utilise deep learning, the resolution needs to be high as well (Mirbod et al. 2021; O'Mahony et al. 2019). Other challenges include the lack of available labelled training data, the need for inexpensive imaging hardware designed for special conditions and the need for mathematical models for utilising the machine vision interpretations in order to extend the system to decision making. According to Tian et al. (2021), Esteva et al. (2021) and Ahmad et al. (2022), the aforementioned challenges are common to many machine vision application fields.

As an example of challenges and their proposed solutions, we can discuss labelling, denoted as time-consuming and expensive (Ahmad et al. 2022). Esteva et al. (2021) propose using transfer learning with models that are trained with standard data-sets as a solution for medical research; but Paullada et al. (2021) noticed that there are several challenges related to standard data, including hidden artefacts and inconsistent labelling, which can bias the models' performance. Special imaging applications and technologies, such as HS imaging, have a known need for training materials for deep learning analysis (Ahmad et al. 2022), which may lead to methodological developments using standard data. It would be interesting to know how much the image analysis methods are eventually influenced, for example, by the features of the Pavia University or Salinas data (HS data, provided by Graña et al. (2022)).

As we now understand, besides the possible bias, standard data and environmental issues might cause challenges in machine vision systems. A pilot system may perform exceptionally well in a controlled environment but fail when implemented in real-life. The standard data might not be equivalent to the information obtained during the machine vision fundamentals' image acquisition phase; the environmental variables among system configurations affect the next fundamental phases (Iqbal et al. 2018; Ren et al. 2021; Mirbod et al. 2021; Paullada et al. 2021). On the other hand, the explainability of deep neural networks is low,

and networks' decisions may be affected by underlying artefacts in the training data (Paullada et al. 2021). Thus, it is important to conduct more machine vision research; we can benefit the machine vision science by utilising machine vision research with novel imagers and data. The results obtained using variable or unique data from different application areas can be utilised in many ways. Besides enabling the easy testing of new application areas, the generalisability of the existing results obtained using specific data and the computer vision approach can be evaluated and validated from a new point of view. Last, it may take 17 years to progress from scientist's trials to real-life applications (Marx and Fuegi 2021), the thresholds between new technologies and real-life applications should be actively pushed with practical scientific approaches and applications.

We can now conclude this section about the motivation and understanding behind the fundamentals of machine vision. We know that each phase affects the next phase and the whole system's performance. This is the primary motivation for exploring and developing machine vision systems as entities and allowing the fundamental phases of machine vision to lead us through hypotheses and research questions to deepen theory and conducted research.

1.5 Hypothesis and research questions

For science, it is necessary to develop methods for imaging and produce research using materials acquired with different types of sensors and imaging setups, which is, literally, the *foundation of machine vision*. In a wider scope, for those designing and developing real-life systems in the industry, it offers more reliable or predictable building blocks based on versatile methods and approaches developed by scientists.

During the conducted research, the fundamental phases of machine vision have been repeated several times with different entry angles. As a result, the research discusses the doubts and solutions raised by the hypothesis, which is "*The machine vision systems should be designed, built and evaluated through the machine vision fundamental phases*". Since the machine vision fundamentals have clear technical and data-analytical domains, both aspects are approached as follows:

1. The aim of the data-analytical computer vision aspect is to show how computationally effective machine learning methods can be optimised, for example, to edge computing in real-time applications. Since sensor development has enabled affordable and lightweight solutions, it is important to develop research methods that are suitable for edge computing, which is a paradigm aiming to bring computation and data storage closer to the data sources (Meribout et al. 2022).
2. The technical aspect is presented with research examples from HS imaging applications and practical examples within Articles PVI, PIV and PV. The aim is to present how a target-wise machine vision system can be designed and built from the sensor and component level to the user interfaces and

analysis tools, including all fundamental phases of machine vision.

The following research questions are raised from the articles included in Chapter 3. Below, Table 3 clarifies the relationship between the sub-questions and articles. Questions from 1a to 1d examine Minimal Learning Machine (MLM), which is a distance-based supervised machine learning method. Questions from 2a and 2b cover the machine vision fundamentals with customised HS imaging applications.

1. How to optimise a computationally effective distance-based machine learning method for real-time applications?
 - (a) How a distance-based machine learning model performs in hyperspectral imaging classification?
 - (b) Can we reduce the method's computational complexity by intentionally selecting the reference training points and maintain a high accuracy?
 - (c) How a distance-based machine learning model for classification can be adapted to local and global anomaly detection, and how to improve the resilience with respect to noise?
 - (d) How to create a self-learning distance-based machine learning model which uses an extremely small amount of labelled training data?
2. How to design and build machine vision systems target-wisely?
 - (a) How to develop and clinically pre-test a hyperspectral imaging system for complex skin surfaces in skin cancer research? How does the target-wisely designed system perform in technical, user-related and classification tasks?
 - (b) How to design and implement a versatile user interface with analysis tools for hyperspectral imaging and provide the experience of easy and correct imaging for any imaging scenes and scientists with no or little background in spectral imaging?

TABLE 3 Research sub-questions and related articles

Sub-questions	Related articles
1a1b	PI
1c	PII
1d	PIII
2a	PIV and PV
2b	PVI

The outcomes of the author's doctoral research are introduced in this dissertation; she worked on a project that enabled the associated research to cover the whole domain, from sensors and programming to analysis and methodological development. Therefore, we will examine the fundamentals of machine vision systems

and combine machine vision, computer vision and machine learning. The aim is to bring not only the latest research results of conducted six articles but also to write and visualise the theory for the reader (scientists and students); to inspire and decrease the threshold of machine vision system implementations.

1.6 Author's other works

Besides the included six articles, which were chosen based on the contribution, the author has published three articles, which are related to the included articles. Article 1. belongs to series of two articles with Article PI, Article 2. is a continuation study to Articles PIV and PV, of where the methods are further improved based on previous studies. Article 3: the optics selection for the sensor fusion was conducted by the author, utilising optical computational methods which are later presented in this dissertation. The Articles 1 to 3 are listed below.

1. "*Minimal learning machine in anomaly detection from hyperspectral images*", by (Pölönen et al. 2020).
2. "*Discriminating Basal Cell Carcinoma and Bowen's Disease from Benign Lesions with a 3D Hyperspectral Imaging System and Convolutional Neural Networks*" by (Lindholm et al. 2022)
3. "*Reflectance Measurement Method Based on Sensor Fusion of Frame-based Hyperspectral Imager and Time-of-flight Depth Camera*" by (Rahkonen et al. 2022)

1.7 Dissertation structure and common acronyms

The beginning of the introduction called for combining light,a machine vision sensor and optical components to capture a signal, process it, and visualize the phenomenon. This dissertation continues by showing how to do it, in theory and in practice throughout the articles.

The dissertation is organised as follows: Chapter 2 introduces the fundamentals of machine vision, being divided into four main sections which are image-capturing (Section 2.1), computer vision (Section 2.2), machine learning (Section 2.3) and imaging systems (Section 2.4). The associated experiments and results are introduced in Chapter 3. The dissertation is concluded in Chapter 4.

About the abbreviations; since a continuous use of several acronyms sometimes creates hard-to-read content, the author intentionally breaks the grammatical rules and avoids using acronyms. However, the most common abbreviations, mostly used in Chapters 3 and 2 and within the articles are: Fabry-Pérot interferometer (FPI), hyperspectral (HS), hyperspectral imager (HSI), Artificial intelligence (AI), artificial neural network (ANN), convolutional neural network (CNN) and Minimal Learning Machine (MLM).

2 MACHINE VISION FUNDAMENTALS

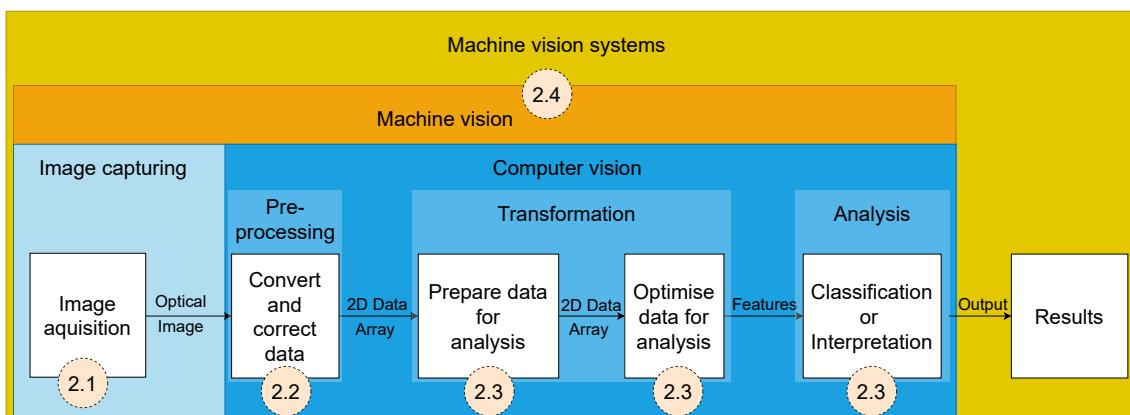


FIGURE 6 The fundamental processes of machine vision with examples of process phases with related subsections.

Machine vision covers image capturing and computer vision processes. This chapter walks thru the fundamental phases of machine vision, using spectral imaging as a domain. Above, Figure 6 shows how each phase has related processes. Those are explored the in subsections mentioned within the dotted circles.

The image capturing processes are introduced in Section 2.1. Section 2.2 discusses the phases of computer vision, offering computation principles to illustrate how a raw data from an HSI can be pre-processed and transformed into an HS data cube. Machine learning and related data transformation examples are introduced in Section 2.3. The big picture of the machine vision systems and all previous phases are presented together as imaging systems with user interfaces in Section 2.4. Since the theory of machine vision systems have a wide coverage and for avoiding the lede to be buried with details, the methods are summarised: Section 2.1.9 is an image capturing summary and Section 2.4.5 concludes the machine vision fundamentals.

2.1 Image capturing

This section aims to point out which sensor and optical details should be considered while selecting components for a machine vision system.

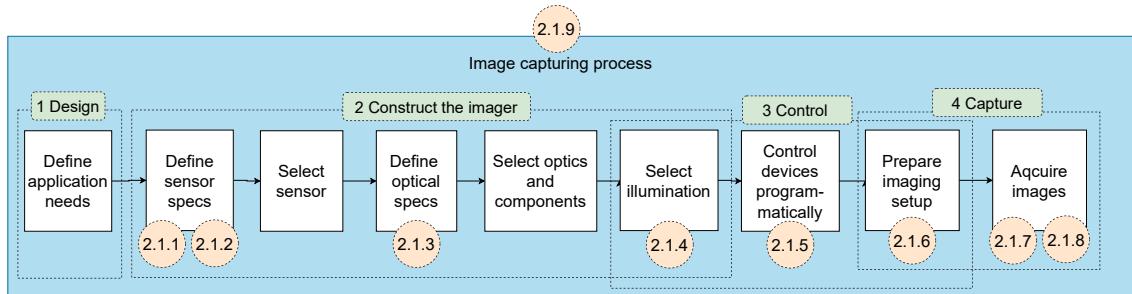


FIGURE 7 Image capturing processes and related process phases

The section's structure is divided into four image capturing topics and related process phases which are visualised above in Figure 7: design, construction of the imager, control and capture. In the scope of this dissertation, image capturing aims to produce a high-quality optical image (a detector array) for the computer vision processes.

This section covers all four topics and processes, with related phases. Similarly, the image capturing process is covered in the included articles; Articles (PIV; PV and PVI) ensembles the control and capture phases (3 and 4 in Figure 7). The illumination can be external or integrated, so the process phase can be included with the imager construction (Articles PIV and PV) or into the controlling process phases (Article PVI).

This section is organised as follows. At first; Section 2.1.1 gives a short introduction of two important standards with the aim of enhancing the reader's understanding of how sensors are compared and how a standard can make the device-controlling phase easier. Section 2.1.2 presents advanced information about machine vision sensors by introducing their physical model and describing how the optical image is formed from the captured photons. Optics and optomechanical components are explored in Section 2.1.3. After that, the topics of illumination (Section 2.1.4), device controlling methods (Section 2.1.5), spectral imagers (Section 2.1.7) and imaging setups (Section 2.1.6) are examined. Finally, the summary (Section 2.1.9) concludes the main points to remember.

2.1.1 Standards

Two important standards are briefly introduced in this section, hosted by the European machine vision associate (EMVA), a non-profit organisation founded in 2003. The EMVA aims to connect organisations in machine vision, computer vision, embedded vision and imaging technologies, which includes manufacturers, system and machine builders, integrators, distributors, consultancies, research

organisations and academia (EMVA 2022). The reason for introducing these standards is that a standard might be valuable for comparing different sensors and it may enable more versatile programming interface than a common programming library.

The first standard is EMVA 1288, which can be used for comparing sensors; the second standard is GenICam, which provides a generic programming interface for sensors and other devices. The GenICam standard is the foundation of the Python library Camazing (Jääskeläinen et al. 2019a), which is used with all HS imagers introduced in this study. Camazing was developed at the Spectral Imaging Laboratory at the University of Jyväskylä (JYU 2022) with a MIT license. No standard has been set for the field of HS imaging or HS imagers. Some ongoing projects can be found via online searches, but, unfortunately, there are no current standards, for instance the associated devices or spectral data features.

2.1.1.1 EMVA 1288

Sensor selection is the first phase, and one of the most important ones in the image capturing process. Manufacturers offer a large selection of different sensors, which vary in their price. Some of the properties introduced in the Section 2.1.2 are definably dependent on the requirements of the target; such as, the resolution, the choice between monochromatic or colour sensor, the shutter type and the camera interface. The challenge lies in the measurable features; such as, the sensor's sensitivity, temporal noise and spectral response.

EMVA 1288 is a standard for the specification and measurement of machine vision sensors and cameras (EMVA 2022). Its main purpose is to create transparency by defining reliable and exact measurement procedures and data presentation guidelines to make it easier to compare cameras and image sensors. Manufacturers that do not hold EMVA licences might measure and present the sensor performance and feature measurements differently and may even leave some important details to the buyers to test, which might be expensive and difficult.

Manufacturers that adhere to the EMVA 1288 standard measure the sensors and cameras according to the standard's exact measurement procedures, making the presented data sheets of different sensors and sensor manufacturers comparable. The "EMVA standard 1288 compliant" logo ensures that products are licensed, measured and presented transparently and reliably. The complete list of measurement groups and their mandatorily can be found from the standard specification (EMVA 2016). The standard's documentation includes mathematical formulas, descriptions of the measurement setups and guidelines for presenting the results in a consistent format. The current release cited in this dissertation is 3.1.

2.1.1.2 GenICam standard

Generic Interface for Cameras (GenICamTM) provides a generic programming interface for various sensors, cameras and other devices, regardless of the inter-

face technology (GigE Vision, USB3 Vision, CoaXPress, Camera Link HS, Camera Link, etc.) they use or what the features they implement.

TABLE 4 The GenICam standard modules. Source: (GenICam 2022).

Module	Main tasks
GenApi	Application programming interface (API) for configuring a camera
SFNC	Standard Feature Naming Convention
GenTL	Programming interface for transport layer (TL) that allows grabbing images
CLProtocol	PI for interfacing Camera Link camera to GenAPI
GenCP	Generic Control Protocol

The GenICam standard comprises five modules, which are introduced above in Table 4. The role of GenApi is to define the mechanism used to provide the generic application defining interface (API) via a self-describing eXtensible Markup Language (XML) file in the device. The XML file format is defined in Schema, which is a part of GenApi. Underneath GenApi is the standard features naming convention (SFNC), which standardises the name, type, meaning and use of device features, resulting in standard functionalities between manufacturers. The features are typically introduced in a tree view, being controllable via an application. The related standard for the consistent naming of pixel formats is pixel format naming convention (PFNC).

GenTL is the standard for the transport layer programming interface. As a low-level API, it provides standard device interfaces and allow to enumerate devices, access device registers, streaming data and delivering asynchronous events. Data Container (GenDC) module allows devices to send any form of data (e.g. 1D, 2D, 3D, multi-spectral, metadata) in a Transport Layer Protocol (TLP) independent format and permits to share of a common data container format for all the TLP standards. The control protocol (GenCP) is a low-level standard to define the packet format for device control, and can be used as a control protocol for each new standard.

The main purpose and benefit of GenICam is to provide an API, which is identical regardless of interface technology used in standardised devices (GenICam 2022). The Camazing Python backend (Jääskeläinen et al. 2019a), which is used to control all three spectral imagers of this dissertation, is based on GenICam.

2.1.2 Machine vision sensors

Machine vision sensors should be selected according to the application they are to be used for. The two scanning types of sensors are line and area scanners, where the former is more demanding. The "area" refers to the sensor shape inside the camera. The line-scanning sensor provides a one-dimensional line at a time, whereas the area-scanning sensor produces two-dimensional data. Therefore, the area scanner is a typical choice for general applications.

The two main types of sensors are charge-coupled devices (CCD) and complementary metal oxide semiconductors (CMOS). The main difference between

the sensor types is the pixel-level conversion from charge to voltage. CCD uses sequential methods, whereas the CMOS sensor reads the pixels parallel. Currently, the performance of CMOS sensors has exceeded CDDs, and the CMOS sensors are widely used in machine vision applications. With CMOS sensors, each pixel is addressable on a row and column basis. The voltages are read parallel, enabling high frame rates and users to define the regions of interest (ROIs). Depending on the number of transistors per pixel, the sensor might have a global shutter or higher signal-to-noise ratio. The sensor's pixel-level operating principles and basic characteristics are described in the following subsections.

2.1.2.1 From photons to digital numbers

The pixel's function is to convert the incoming photons into electrons. The electrons are converted into a voltage, which can be measured. Each sensor has a full-well capacity, which describes the maximum number of electrons stored in a pixel. Below, Figure 8 illustrates a pixel well and a physical model of an imaging sensor that converts the photons into digital numbers. As can be seen in Figure 8, the full-well area is proportional to the pixel's light-sensitive front area.

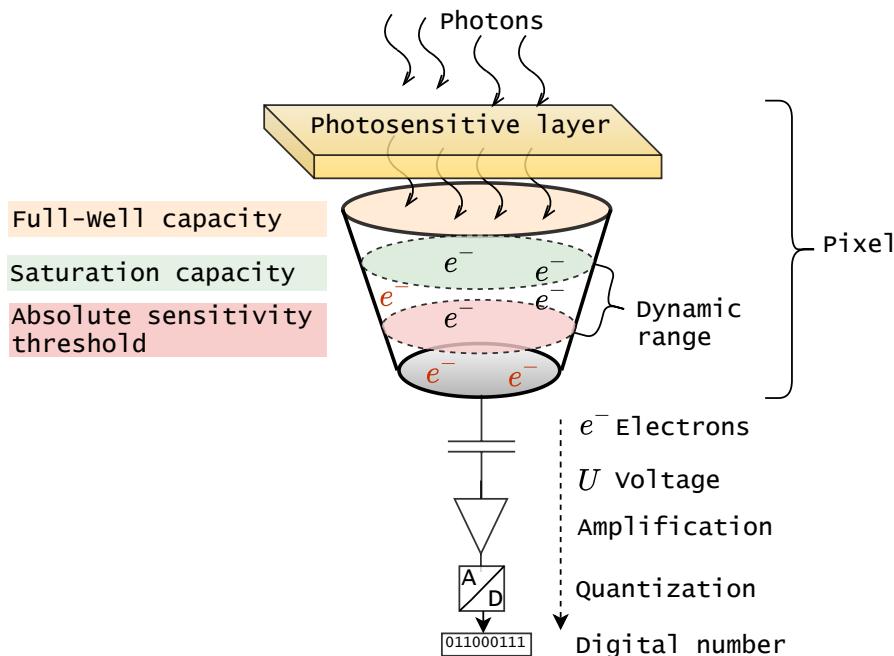


FIGURE 8 A simplified physical model of a pixel and imaging sensor. A number of photons hit the photosensitive pixel area during the exposure time. The photons are converted to photo-electrons. The charge formed by the electrons e^- is then converted by a capacitor to a voltage, before being amplified and digitised, resulting in the digital grey values. The red e^- denotes temporal noise. The pixel-depth dependent physical properties, from the full-well capacity to the dynamic range, are visualised in the pixel well.

Dynamic range (DR, Figure 8) is the ratio between the smallest and largest

amplitude of the signal a sensor can produce (Baumer 2022). It describes the imager's ability to simultaneously provide detailed information from bright and dark areas. The parameter is important when the illumination conditions rapidly change or the targets have a strong contrast.

The area that can capture light is called the fill factor. Since each pixel in a CMOS sensor has its own readout, charge conversion and digitalisation structures (visualised as the substrate, below in Figure 9), the interline transfer might have a 20-50% fill factor from the pixel size (Stemmer 2022). This affects the sensors' overall photosensitivity, which can be improved, for instance, by using microlenses in front of the pixels (Figure 9).

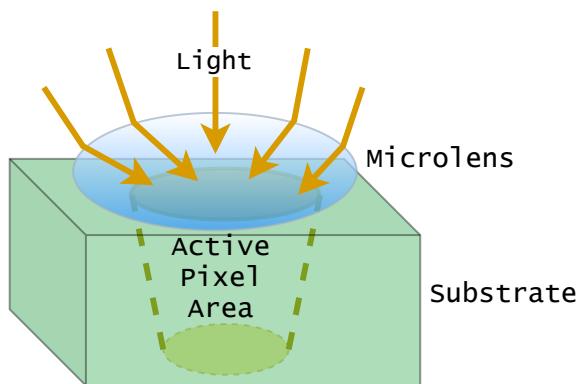


FIGURE 9 Full well capacity i.e., maximum electrical charge possible, is limited by the physical pixel depth. One way to improve the pixel's photosensitivity is to use microlenses above the pixel well. Therefore, some of the light heading into the sensor's substrate areas can be directed to the pixel.

Photons are converted into electrons, and the conversion ratio is called quantum efficiency (QE). The quantum efficiency depends on the wavelength. The sensor's light sensitivity depends on the number of photons converted to electrons. The more photons are converted, the greater is the quantum efficiency, increasing the level of information provided by the sensor. If the sensor is used with filters, the measured quantum efficiency of the system might differ from the sensor level quantum efficiency (EMVA 2016).

2.1.2.2 Sensor noise and signal-to-noise ratio

Sensors have different sources of noise. Dark current noise occurs when the electrons emerge through thermal processes in the pixel. The level is related to temperature and exposure time by increasing with them. Photon noise is caused by light, as the photon flux striking the sensor is Poisson-distributed (EMVA 2016). This limits the maximum signal-to-noise ratio (SNR). Readout noise occurs when the electrons are converted into voltages. The quantisation noise is caused when the voltages with continuous values are converted to digital values with discrete values (A/D-conversion). Temporal noise is a combination of all the aforementioned sources of noise. It exists even when the pixels are not illuminated. The

exposure time and temperature generate electrons without light. The level of a dark signal varies.

The signal-to-noise ratio is the ratio between the maximum signal and the noise floor. It describes a real signal after the A/D conversion. When the signal-to-noise ratio is 1, the maximum signal and noise floor level are equal (EMVA (2016) and Stemmer (2022)).

2.1.2.3 Sensor sensitivity and spectral response

The quantum efficiency and background noise influence the sensitivity of the sensor. The sensitivity is high when the quantum efficiency is high and the background noise level is low. The background noise level is measured with the lens covered. Each pixel has an absolute sensitivity threshold (AST), which describes the lowest possible number of photons with which the sensor can produce a useful image. The sensitivity of a sensor increases when the threshold decreases. Absolute sensitivity threshold is a significant variable in low-light applications. Absolute sensitivity threshold combines the quantum efficiency, dark noise and shot noise values and it is determined when the signal-to-noise ratio ratio level is 1 (EMVA (2016); Stemmer (2022) and Baumer (2022)).

Spectral response describes the wavelength range that a sensor can capture. Typically, CMOS sensor's range is from 350 nm to 1100 nm (EMVA 2016). Some CMOS sensors might have enhanced sensitivities for VNIR imaging at the range of 700 - 1000nm. The spectral imagers used for IR applications can use InGaAs sensors that have range of 900 to 1700 nm or SWIR sensors. The UV sensors' spectral sensitivity ranges upwards from 200 nm (EMVA 2016).

2.1.2.4 Spatial resolution and shutter types

The number of active pixels in a sensor is called spatial resolution. The optimum resolution should be calculated target-wisely for robust inspection systems. The key is to use the measures of the smallest feature in the field of view, which defines the minimum resolution. As a practical example, suppose the feature size is 1×1 , the object size is 100×100 , and the analysis method requires 3×3 pixels per one feature. The minimum resolution is addressed by multiplying the object size with the requirement: 300×300 pixels (Stemmer 2022). The mentioned formula is for monochromatic sensors. The minimum resolution for a colour sensor with a Bayer pattern should be doubled.

The sensor shutter type describes how the sensor exposes the pixels. A rolling shutter starts and stops the exposure sequentially, row-by-row or pixel-wise. The delay of all pixels to be exposed can be up to $1/\text{framerate}$, which can be an issue when the target moves (Stemmer 2022). A rolling shutter sensor might have a global shutter start, which allows all pixels to be activated for exposure simultaneously but turned off sequentially. This mode might cause some blurring to the bottom areas of the image, and its use demands a proper illumination design. The global shutter activates and deactivates its pixels at once, being the

most practical choice in most applications dealing with moving targets.

2.1.2.5 Monochromatic and color sensors

Sensors can be divided into monochromatic and colour sensors, and the selection should be made depending on the target. The main difference between monochromatic and colour sensors is the colour filter placed in front of the pixels. A sensor with a pixel-wise filter provides colour filter arrays (CFAs) that can be computed to colour images using demosaic algorithms. In contrast, the monochromatic sensor provides data that can be processed as an image directly.

The CFA limits the sensitivity of each receiving pixel well into a single part of the visible spectrum. Therefore, each red-green-blue (RGB) channel's CFA pixel has a limited spectral range, representing either red, green or blue spectral responses (Alleysson et al. 2003). The CFA filter is part of the imaging system's spectral sensitivity, which is determined as a combination of the sensor sensitivity and the transmittance of the CFA filter, such as Bayer blue-green-green-red (BGGR) (Sadeghipoor et al. 2012).

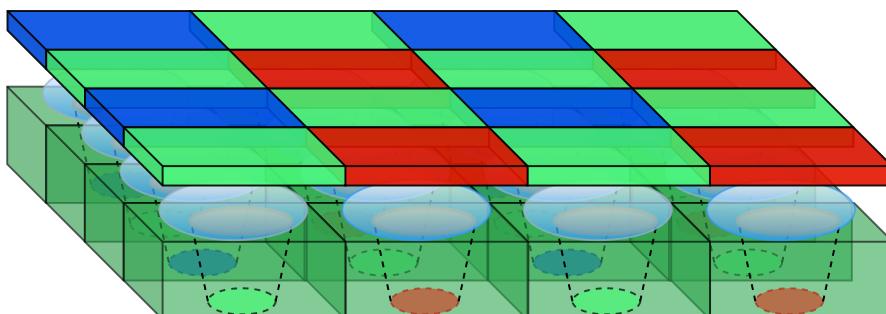


FIGURE 10 The Bayer BGGR pattern filter is placed in front of the sensor's photosensitive pixel wells. The pixel wells measure the intensity of light, providing information to green, blue or red pixels. The full three-channel RGB image is produced with interpolation for the missing pixel values.

Above, Figure 10 visualises the Bayer pattern BGGR filter placed in front of the sensor. Bayer BGGR filter is a common RGB filter, repeating patterns of 2×2 pixels: one blue, two green and one red. Demosaic algorithms perform an interpolation that estimates the three-pixel values of the RGB image Eskelinan (2019).

Below, Figure 11 shows that the value of each missing pixel can be calculated, for example, with a bilinear interpolation, where each of the 2D plane's pixel values is an average of the neighbouring pixels. The red, green and blue planes can be calculated separately with convolutional operations by placing zeros to missing values in R_0 , G_0 and B_0 , and sliding a weighted 3×3 kernel over the red, green and blue colour planes, filling the middle pixel with the result Eskelinan (2019).

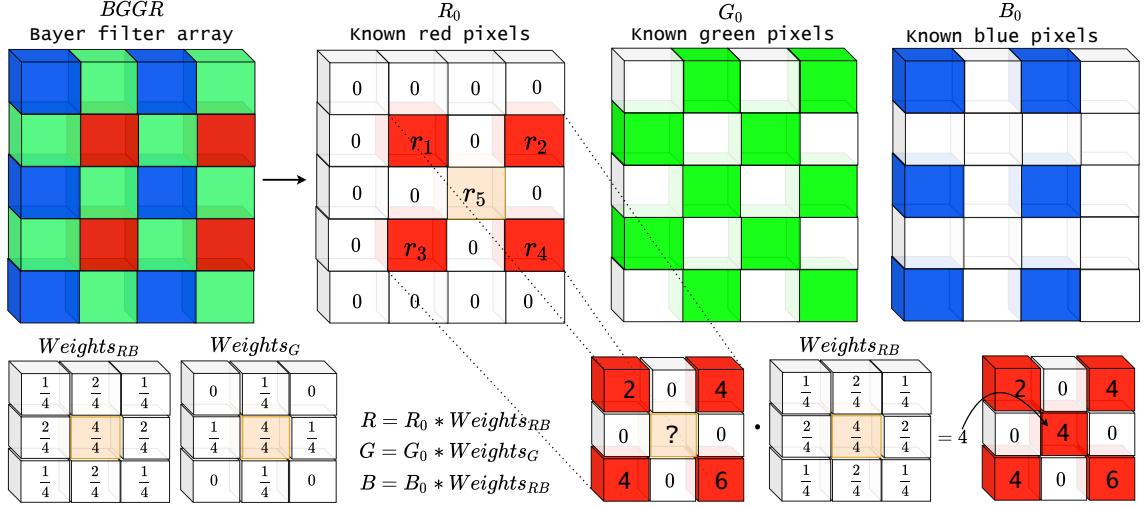


FIGURE 11 Bilinear convolution. The Bayer BGGR color filter array is divided into three color planes R_0 , G_0 and B_0 . Each of the missing values is set to zero (0). A weighted kernel ($Weights_{RG}$ or $Weights_G$) slides over each pixel of the colour planes (y and x directions). The value of the middle pixel is the weighted average of the neighbouring pixels.

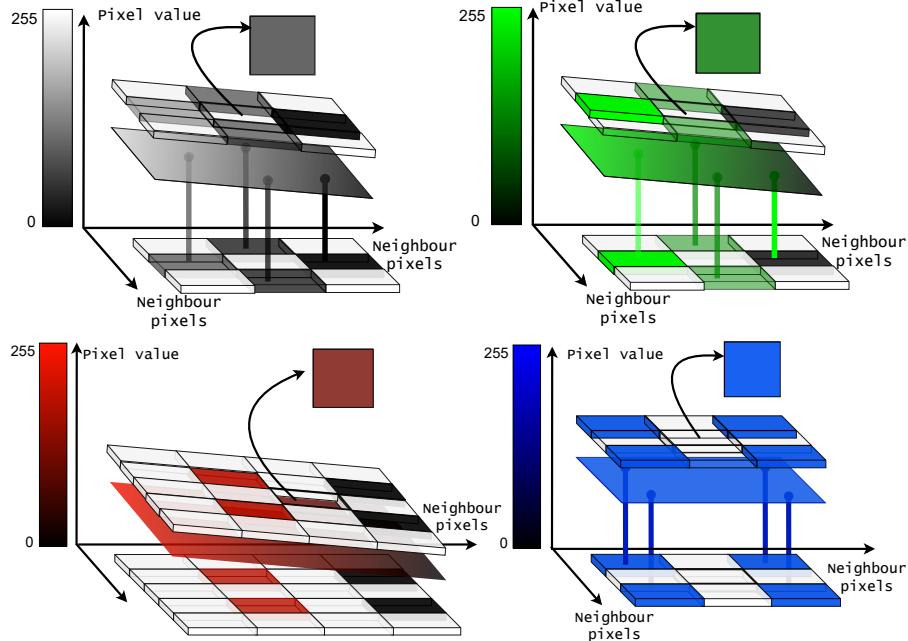


FIGURE 12 Bilinear interpolation. Intuitively, the bilinear interpolation can be seen linear towards the image's x and y directions, where the mean of the neighbouring pixels forms a surface, of where the middle pixel gets its pixel colour value. The colour bar represents the RGB colours' minimum and maximum values (0 to 255). A pixel captured with a high number of photons reaches higher values, resulting in a "bright colour". A pixel with a low number of photons gets values close to zero, and the pixel is seen as black.

Above, Figure 12 illustrates how the behaviour of the pixel values in the RGB colourspace. The minimum RGB value of one channel is zero (0), which is

visualised as black. The highest value 255, is white in a grey scale image, but in a colour channel, it is visualised as the most intensive red, green or blue tone.

Intuitively the interpolation can be seen as a pixel value surface formed from the mean of neighbouring pixels. The pixel value surface is drawn above the original Bayer pattern in the illustration (Figure 12). After interpolation, the top pattern with the individual middle pixel represents the computed pixel value. The green and red examples are fading colours, so the surfaces are inclined. The blue represents an equally intensive pixel neighbourhood, where the pixel value surface is drawn horizontally straight.

Sources for the technical details

The detailed mathematical formulae for computing most of the above sensor characteristics can be found in the EMVA 1288 standard (EMVA 2016). Section 2.1.2 is based on the general information and EMVA standard (EMVA 2016) that are freely available online. The selected technical details, with no explicit scientific citations, were written by the author. Besides the standard, the details were confirmed at the time of writing from the technical materials of leading sensor manufacturers and machine vision system developers: (Baumer 2022; Basler 2022; Stemmer 2022).

2.1.3 Optical and optomechanical components

Machine vision cameras require optical designing. When the sensor is mounted with optical elements, such as lenses and other optomechanical components, we can call it *machine vision camera* or *machine vision imager*. Commercial lenses can be used with optomechanical components and machine vision sensors. Further, prototype imagers can be built from scratch using, for example, extension tubes, individual lenses and filters. Combining a sensor and lens requires only a basic understanding of an imager's design, whereas the more complex systems require a mathematical understanding over the geometrical optics. Both approaches can be valuable in science. Prototypes pave the way for new applications, and the optics, if needed, can be improved within the continuation studies. This subsection introduces the basic terminology for optics, which would help to select and understand the optical parameters for designing a machine vision imager.

2.1.3.1 Terminology for optics

The simplest example of a machine vision imager is a combination of sensor and a commercial lens. The lens selection parameters should be defined based on the target features and imaging setup. The lens features such as mount, field of view (FOV), focal length (FL), depth of view (DOF), resolution, possible polarisation and IR-cut are important (Greivenkamp (2004) and Stemmer (2022)). Short definitions of the commonly used terms are introduced below in Table 5.

Each lens parameter should be evaluated from the sensor and imaging setup

point of view since the decisions directly affect the imaging quality and the prototype's ability to perform its expectations. A carefully designed system with suitable optics can reduce image-quality-related issues, decreasing the need for computational corrections (Greivenkamp 2004).

TABLE 5 Optics terms and definitions , collected from (Greivenkamp 2004) and (Stemmer 2022).

Term	Abbr.	Description
Lens mount	-	Describes the type of the mount that attaches the lens and sensor
Focal point	-	An optical centre at lens where the light is focused to infinity
Focal length	FL	The distance (mm) between the focal point of the lens and the sensor plane
Back focal length	BFL	The distance between the last surface of an optical lens to the sensor plane.
Flange focal distance	-	Distance from the sensor mount (flange) to the sensor plane.
Magnification	-	Ratio between target size and image size, defined by FL
Field of view	FOV	The area captured with sensor, defined by sensor size and FL
Depth of view	DOF	Distance between the nearest and the farthest objects that are in acceptably sharp focus
Working distance	WD	The distance between the leading edge of the lens and target
Minimum object distance	MOD	Minimum distance between the lens and target where the image quality remains at an acceptable level
Aperture	-	An opening of an optical system through which the light travels
Adjustable iris	-	Mechanically adjustable aperture, each standard increment decreases the amount of light 50%
F-number	F	The amount of light passed to the lenses; FL divided by aperture diameter
Image circle	IC	A circular area in a sensor plane which the optics can uniformly expose without remarkable vignetting on the margin
Sensor size	-	Physical dimensions of the sensor
Maximum IC diameter	-	The image circle and sensor size needs to match. The maximum circle diameter is the sensors' diagonal
Optical resolution	-	Lens resolution defines the detailed structure of the image. The resolution of the lens should be the same or better than the sensor's resolution.

2.1.3.2 Forming an optical image

Figure 13 (below) visualises the basic setup, in which the lens reconstructs the scattered light into an image, which happens in a light-sensitive area between the lens and the sensor. The parameters that impact the image are the lens radii, the distances between the lenses, the working distance and the distance between the lens and the sensor (Greivenkamp 2004).

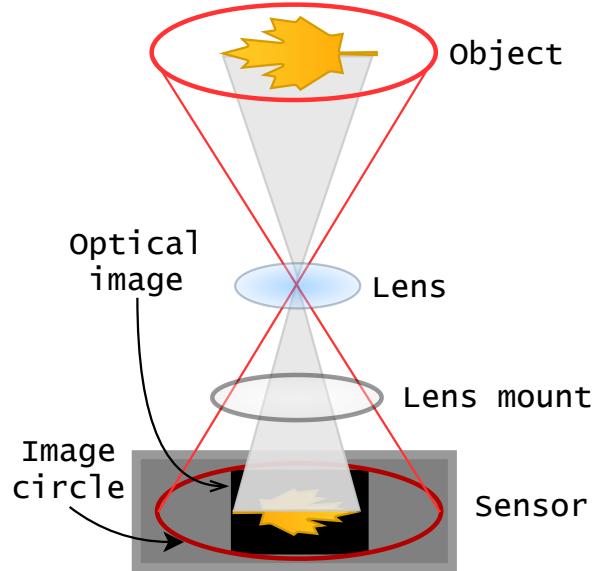


FIGURE 13 Forming an optical image. The lens reconstructs the scattered light into an image captured with a sensor. The sensor's photosensitive pixel area captures an optical image formed through a lens. The object size, lens properties such as focal length, lens radii and the distance between the object, lens and sensor are examples of optical parameters that affect the captured image. The image circle is formed when a light strikes a perpendicular target, i.e., the sensor, which forms a circle of light.

2.1.3.3 Lens mount

The lens must be attached steadily to the sensor. Two common lens mounts in machine vision systems are S and C. S-mount lenses are small and inexpensive, being often used with board-level sensors (Stemmer 2022). A typical S-mount lens has a fixed focus and minimal adjustment possibilities, whereas C-mount lenses are the most common lenses in machine vision applications. C-mount optics have a wide range of compatible components, enabling them to be used in prototype imagers constructed with different optical and optomechanical components. Typically, these lenses have adjustable iris and focus. The price range resonates with the optics' quality and resolution. Several other mounts exist, but C- and S-mounts are considered as the most suitable for basic research purposes due to their satisfactory quality, available accessories and affordability (Baumer (2022), Stemmer (2022)).

2.1.3.4 Lens image circle and sensor diagonal

A carefully selected lens provides high-quality images from the objects. The image circle is formed when a light strikes a perpendicular target, i.e., the sensor, which forms a circle of light. Below, Figure 14 shows the relation between the lens image circle and sensor diagonal. As a lens-related property, the image quality typically deteriorates towards border areas, and the images might suffer from shading or vignetting. To avoid mechanical vignetting, choosing the right size

optics with the sensor is necessary. If the lens image circle or lens mount is too small, the image will be heavily vignetted (lower-middle image).

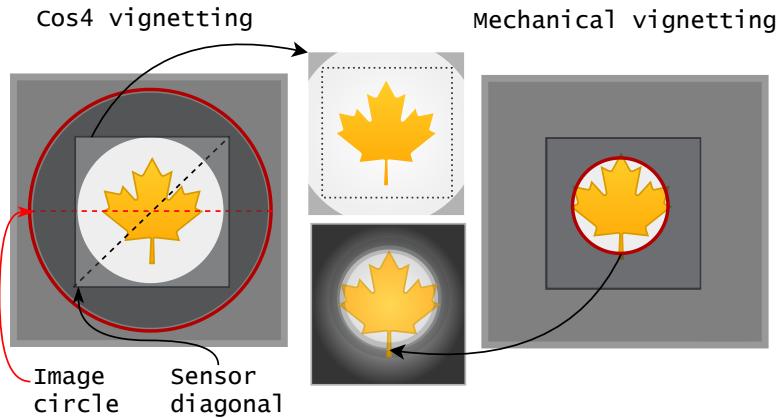


FIGURE 14 Image vignetting and shading. In the illustration on the left, the image circle is larger than the sensor diagonal, and the shade is caused by Cos^4 vignetting. In contrast, in the illustration on the right, the image circle is smaller than the sensor diagonal, causing mechanical vignetting. Both situations can cause vignetting or shades. If the object is in the middle of the non-shaded area, the image can be cropped, as shown with a dotted line in the upper-middle image. The right side example produces images that are not usable (lower-middle image).

Another source of vignetting, Cos^4 , is seen in the upper-middle image (above, Figure 14). If the light travels to the edges of the image from a further distance and reaches the sensor at an angle, it affects image quality; the light falloff is determined by the $\cos^4(\theta)$ function, where θ is the angle of incoming the light with respect to the image space's optical axis. The drop in intensity is more significant for wide incidence angles, causing the image to appear brighter at the centre and darker at the edges (Stemmer 2022; Greivenkamp 2004).

Since it is sometimes difficult to find inexpensive optics of the right size for the desired sensor (other parameters that depend on the target might limit the selection), the lens image circle can be oversized. In such cases, the images can be cropped and used, or the effect can be controlled by decreasing the aperture size.

2.1.3.5 Focal length, magnification and working distance

Focal length (FL) affects the field of view and magnification, as it is one of the most important parameters when choosing optics. Focal length refers to the distance between the main plane of the lens and a particular point where the light is focused from infinity. Below, Figure 15 visualises how the focal length, marked with f and f' , affects to the image height. A long focal length provides greater magnification than a short focal length (Greivenkamp 2004).

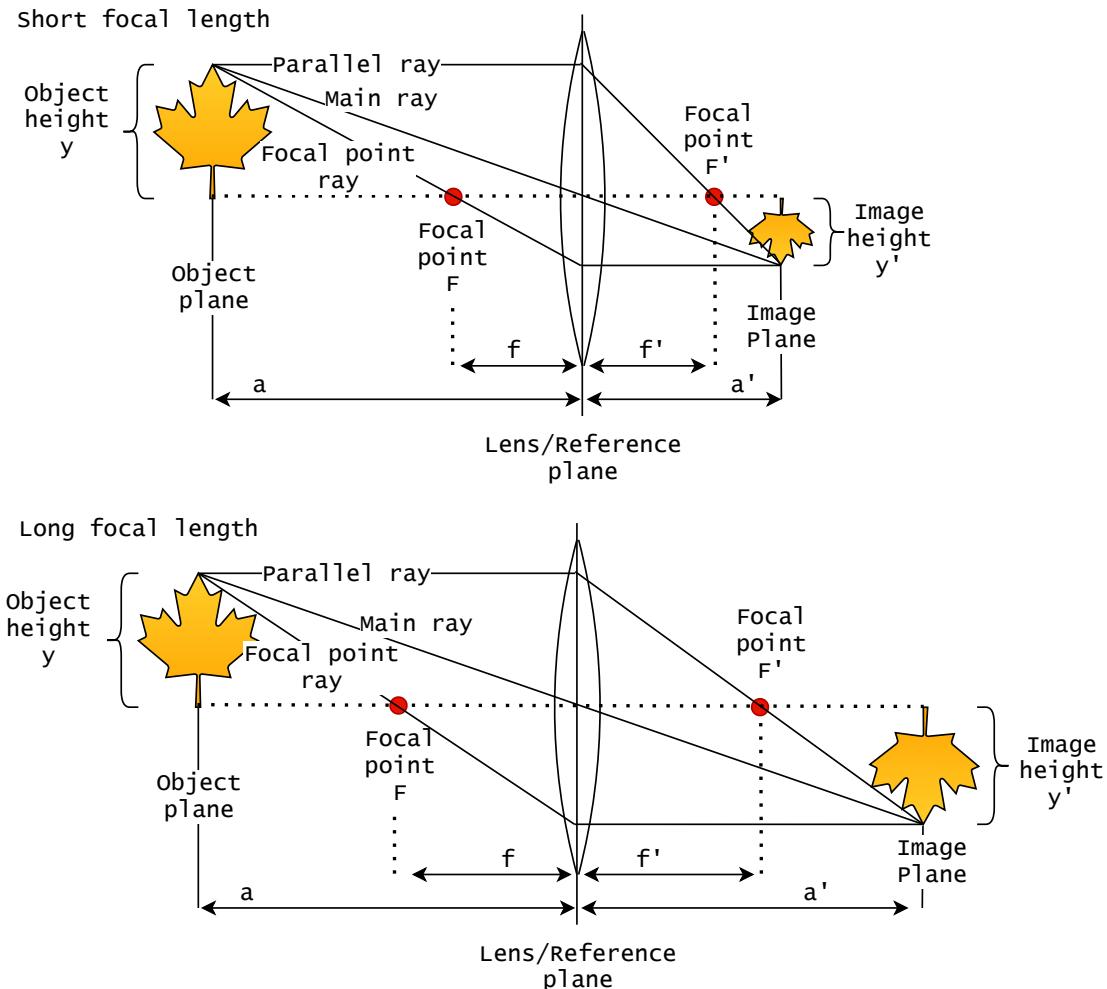


FIGURE 15 Long and short focal length. Figure parameters: a : working distance, a' : image distance, f : object-side focal length, f' : image-side focal length, F : object side focal point, F' : image side focal point, y : object height and y' image height. Increased focal length magnifies the object size on image plane (y'), while the working distance can remain unchanged.

The working distance, object size and focal length affect the magnification. From above, Figure 15, we can see that a longer focal length increases the magnification without extending the working distance a . Magnification β can be approximated for a non-complex optical setups as follows (Greivenkamp 2004; Stemmer 2022):

$$\beta = \frac{a'}{a} = \frac{y'}{y}. \quad (1)$$

For thin lenses,

$$\beta = \frac{1}{f'} = \frac{1}{a} + \frac{1}{a'}. \quad (2)$$

Below, Figure 16 explains the lens opening angles and their relations with the horizontal, vertical and diagonal field of view (FOV). By looking at the angles and rays of the image, we can see that the relation between the field of

view and working distance correlates; the image field of view decreases while the working distance shortens and vice versa. The relation between the opening angle and focal length is the opposite; the larger the focal length, the narrower the opening angle. Most of the lens parameters described in Table 5 (Section 2.1.3.1) can be calculated either with pen and paper or using services like Vision Doctor (Doctor 2022) or sensor manufacturers' web tools, which are provided for designing imaging systems. While there are many parameters to consider, the infrared (IR) cut and possible colour corrections are the least worth mentioning features. Suppose an imaging system is for spectral imaging and the interesting wavelength range is in IR. It might be good to exclude lenses designed to block IR light or lenses that have some other undesirable colour corrections by default (Greivenkamp 2004; Stemmer 2022).

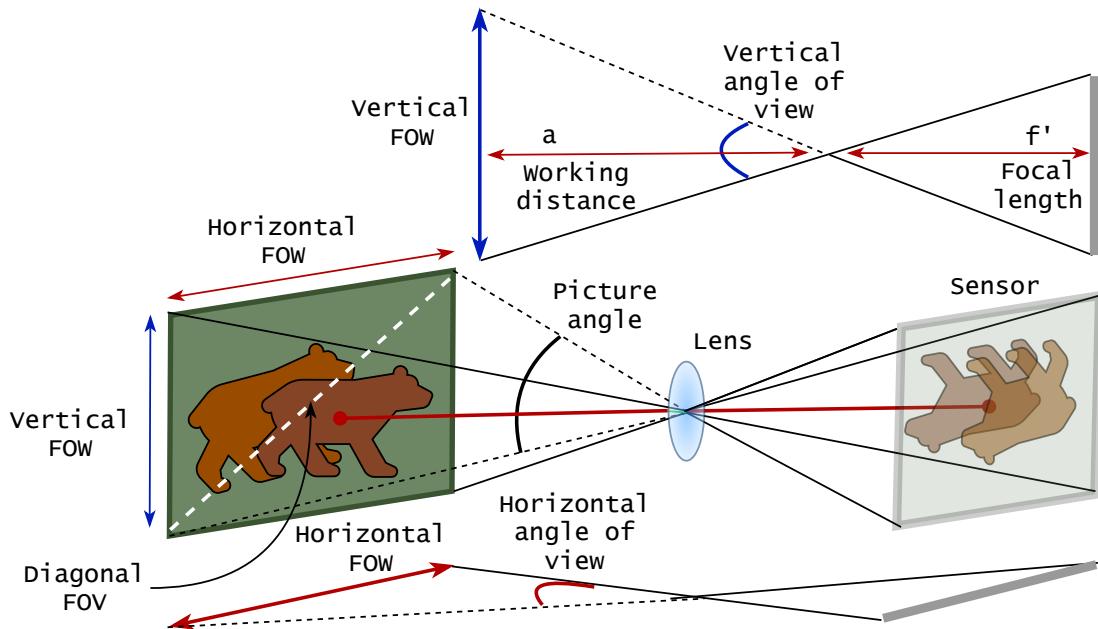


FIGURE 16 **Field of view** is affected by the lens focal length, opening angles and working distance. These parameters can be evaluated, and the lens should be selected according to the working distance and target measures.

2.1.3.6 Optomechanical components

Extension tubes are useful for creating more complex prototype imagers than a basic sensor and a lens. With extension tubes, the optomechanical components can be mounted together. Different band-pass filters, beam splitters and lenses can be used to produce, for instance, a simple two-channel spectral imager (introduced as future works in Section 4.4). While designing such a system, it is valuable to understand that the system's parameters will change depending on where the extension is added. For example, if the extension is placed between the lens and the sensor, the image-side focal length increases, decreasing the minimal working distance and field of view, magnifying the object (Greivenkamp 2004).

2.1.4 Illumination

The illumination of a machine vision system can be understood in two ways. It can be used as an internal light source that is integrated into the imager. Another interpretation is the environmental light, designed to emphasise the target phenomenon and fade out non-essential features. Designing the most effective type of lighting is one of the most significant tasks in machine vision applications (Golnabi and Asadpour 2007). A low illumination level can cause a low signal-to-noise ratio, decreasing the image quality and increasing the need for computational data corrections. Images captured with too bright light can cause over-exposure pixels, which does not include feature information for the analysis.

The illumination should be selected target-wisely. The light source can be, for example, laser, tungsten lamp, fluorescent, halogen or light-emitting diode (LED) (Golnabi and Asadpour 2007). A typical light source in HS imaging is a halogen since it emphasises the features by providing a wide spectrum of wavelengths of light. Besides the wide spectrum, the approach can be designed with a few selected LEDs, providing certain wavelengths of light. The benefit of LED lighting is the energy consumption and balanced temperature. A halogen light might warm up the imaging environment, which might be an issue in some applications.

A stray light, uncontrollable backlight or sample containers can be problematic. A stray light protection cone (shown in Article PIV) can protect the machine vision system's integrated illumination. Reflections caused, for example, by sample containers (glass containers), shiny skin or some other attributes of the targets can be controlled with polarising optics (Uma and Sukanya 2021).

The properties of the specific light source and the illumination configuration should be designed according to the target. The illumination can be, for example, a backlight, spotlight, ring light or diffuse light. Section 2.1.6 will show how to configure different imaging setups with designed illumination for different HS image analysis needs.

2.1.5 Controlling devices with Python

When choosing a sensor, one important parameter is the device interface. Many of the available machine vision sensors can be controlled using Python programming. Typically, sensor data sheets offer alternatives for programming, and manufacturers might provide software that enables the sensors to be used without programming.

The benefit of using Python wrapper packages is the associated ease; if Python is combined with GenICam transport layer, as it is when using Camazing device backend (Jääskeläinen et al. 2019a), the programming is easy, and all of the sensor's features can be inspected and set in a versatile way. Compared to common libraries, such as open CV, the Camazing package offers possibilities to use commands for each sensor feature, and there are no backend-related limits. Hence the package adapts to the sensor's features and limitations.

For instance, by using Camazing, after the sensor is initialised, it is possible to ask the sensor to list all its features, feature explanations, restriction policy and minimum and maximum values. The restriction policy changes during the acquisition - some features can be set while the sensor is acquiring, while some features are locked during acquisition. Similarly, some of the feature dependencies can be revealed by inspecting the values after changes - for example, the frame rate and exposure time restrict each other, and sensor width and height affect enabling and disabling the offset settings. By using GenICam compatible backend, we can access features that might not be available via common libraries. For example, some sensors have settings for image autocorrection, choices for image sharpness algorithms, which to choose from, or the possibility to check the sensor temperature while capturing. For example, it is important to set white balance, and other colour correction features off with spectral imagers and for the long-term capturing series, it may be of value to monitor sensor temperature as it relates to noise.

Therefore, it is a good practice to inspect the sensor's individual features and feature relations first and then design the sensor command logic after the sensor's operating principles, logic and restrictions are known. In this dissertation, the Camazing Python backend was utilised to create the HSI software introduced in Articles PVI; PIV, and PV. Besides Camazing, which was developed at the University of Jyväskylä, Spectral Imaging Laboratory (JYU 2022), the spectral separator and LEDs were controlled with in-house Python libraries Spectracular (Jääskeläinen et al. 2019b) and unreleased LED backend.

2.1.6 Imaging setups

Illumination is an important factor in HS imaging. Below, Figure 17 shows how different imaging setups can be configured with designed illumination for different HS image analysis needs. Subfigure 17a is a basic setup for imaging reflectance and radiance HS cubes. The halogen light is diffused evenly from two sides at equal angles to the target. The imaging setup is symmetric, and the diffused light flux lands evenly on the object's surface.

Subfigure 17b represents the imaging setup for bidirectional reflectance. The setup has one diffused halogen light source, which is positioned towards the target at a known angle α drawn in Figure 22a in Section 2.2.2. The fourth image (17d) shows an example of proper illumination, emphasising the target features without unwanted reflections. Regardless of the imaging setup, the sensor's exposure time is set to be as high as possible to decrease the noise.

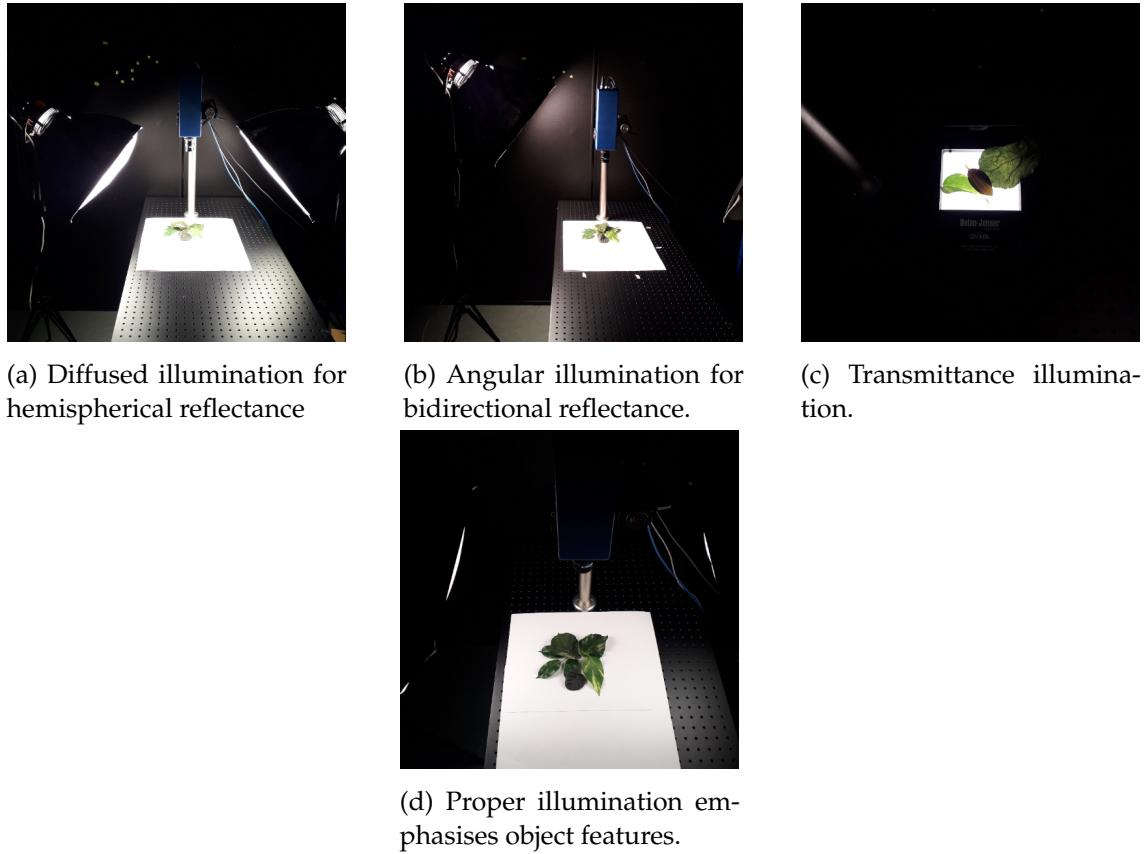


FIGURE 17 An example of illumination in spectral imaging. Here, (a) illustrates evenly diffused illumination from both sides of the target, while (b) is an example of an imaging setup for bidirectional reflectance. The setup has one light source, and the light travels at a known angle towards the target. The setup in (c) is for transmittance imaging. Finally, (d) is an example of good illumination that causes minimal unwanted reflections from the target and provides evenly illuminated conditions to highlight the target features for spectral imaging.

2.1.7 Spectral imagers

Spectral imagers can be divided into multispectral and hyperspectral imagers (HSI) depending on the number and width of the bands they capture. A typical multispectral imager captures three to ten wide spectral bands, whereas the HSI captures up to thousands of narrow 1-20 nm bands. The width of the captured spectral bands is called *spectral resolution*, defining the imagers' ability to record fine wavelength intervals. Term "Full Width Half Maximum" (FWHM) describes the spectral resolution. A high spectral resolution implies narrow spectral bands. Depending on the imager's construction, it can be a point or line scanner or a framing camera. The main difference is the scanning technique, which can be a benefit or disadvantage depending on the application.

2.1.7.1 Spectral separators and scanning techniques

A spectral separator is an optical or optomechanical component that is used to separate light wavelengths. A classic example of this is a glass prism reflecting different light wavelengths as a rainbow on a wall. This example can be used inside spectral cameras, as seen below in Figure 18. However, there exist more sophisticated spectral separators, such as related Piezo-actuated metallic mirror Fabry-Pérot interferometers (FPIs) which were developed and patented by the Technical Research Center of Finland (VTT). FPIs' operating principles are soon introduced in Section 2.1.8.

Point scanners (Figure 18a) are typically whiskbroom imagers. The HS image is constructed from spectra dispersed with a prism or grating. A whiskbroom point scanner operates by capturing a one-pixel spectrum at a time with a point-like aperture, and the spectrum is recorded with the one-dimensional linear sensor. The HS image is formed combining the recorded pixel spectra (Wang et al. 2017). A point scanner can produce a high spectral resolution. However, the process is relatively slow due to the spatial scanning over the x and y dimensions, which might limit the exposure time (Li et al. 2013).

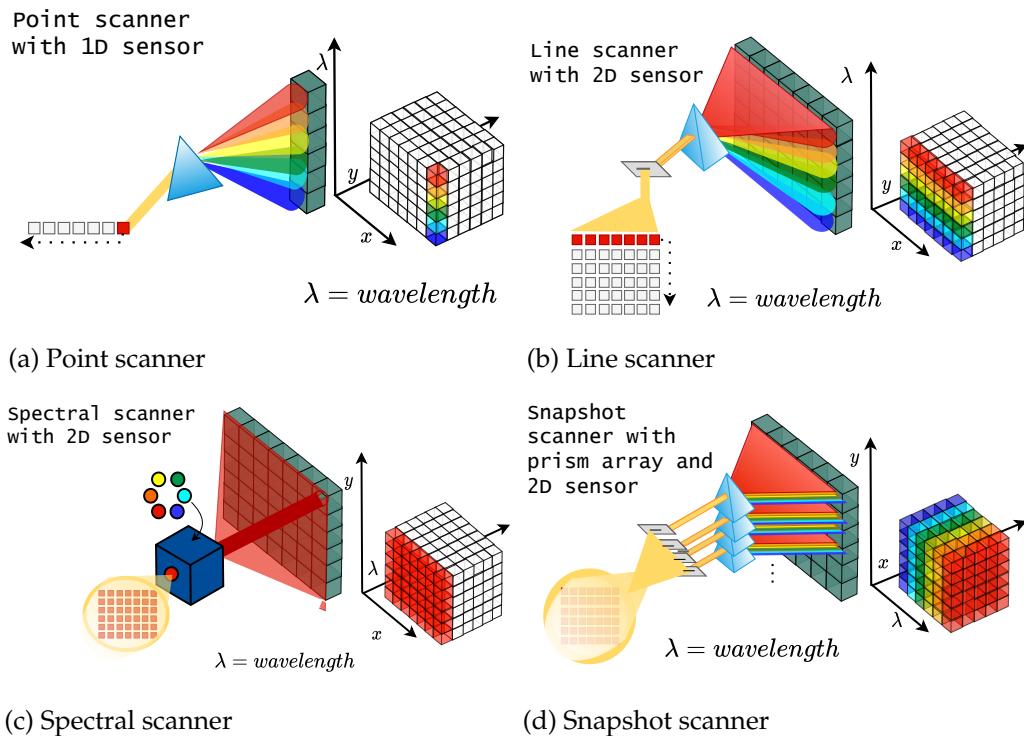


FIGURE 18 HSI types and scanning techniques for image acquiring. A whiskbroom scanner (a) captures one-pixel spectrum at a time with a point-like aperture, and the spectrum is recorded with the one-dimensional linear sensor. A pushbroom scanner (b) projects a strip of the scene onto a slit, which is dispersed using a prism or grating. The projected spectrum is captured with a two-dimensional sensor, and the HS image is formed one pixel spectrum row at a time. Staring ie. spectral scanner (c) captures HS images frame by frame, each frame representing different intensity of wavelength of light. A snapshot HSI (d) captures the whole HS data cube at once.

The line scanner (above, Figure 18b) is an example of a push broom scanner. For each pixel spectrum row, a line scanner projects a strip of the scene onto a slit, which is dispersed using a prism or grating. The projected spectrum is captured with a two-dimensional sensor (Li et al. 2013). The spatial dimensions of the line scanner's HS data cube is constructed one pixel row by row. Since one pixel row is recorded at once, the scanning process allows for longer exposure times than the point scanning, but for producing smooth HS images, the scanning over the target to x or y direction requires synchronisation between the scanning speed, exposure time and frame rate (Li et al. 2013). Line scanner's spectral resolution is high, and the system can operate at a medium speed. The typical application of a line scanner is in remote sensing or in conveyor belts (Wang et al. 2017). A small spectral imager presented in Riihiaho et al. (2021) is an example of a low-cost DIY imager based on spatial line scanning.

Spectral scanning (also known as staring, wavelength scanning and band sequential method (Li et al. 2013)), is presented above in Figure 18c. It utilises the 2D sensor's pixels frame by frame. Each sensor output is a spatial map of the scene, either monochromatic optical array or colour filter array. Typical wavelength scanners use filters for spectral separation; instead of prisms or grating, the spectrum is separated using fixed band-pass filter wheels, linear variable filters, variable interference filters, interferometers or tunable filters (Li et al. 2013). The operating principle is straightforward; the wavelengths are selected and filtered to the sensor, and frames with x and y spatial dimensions are recorded individually. The HS image is constructed by selecting the output wavelength with filters as a function of time (Li et al. 2013). The spectral resolution varies from low to high, and the speed ranges from medium up to high (Wang et al. 2017). The system allows the wavebands to be selected, and some devices allow the exposure time to be changed between the channels, which is a benefit compared to typical point and line scanners (Li et al. 2013). The downside of spectral scanning is that the approach is suitable for stationary scenes and platforms until all of the wavebands are captured. A slight or noticeable movement during the capturing process shifts the spatial coordinates of the pixel spectrum.

A snapshot (e.g., single-shot) scanner is an HSI that captures whole HS data cube at once (above, Figure 18d). The spectral dimensions are reconstructed from a perspective projection of the data cube. The HS image is achieved at once by simultaneously remapping and dispersing the scene onto a two-dimensional sensor (Li et al. 2013). Multiple configurations exist for the snapshot imagers, which are technically complex; the presented visualisation with prism arrays is heavily simplified.

2.1.7.2 Spectral imagers and user interfaces used in this dissertation

The HSIs used in Section 2.2.1 and Articles PVI; PIV and PV are examples of high-speed spectral scanners. The third VTTs HSI, which uses a microoptoelectromechanical system (MOEMS) FPI (MFPI), is excluded from the scope of the dissertation's methods, but its technical details can be found in Article PVI.

Section 2.2.1 describes how a raw image is processed into spectral data. The raw images are captured using VTTs prototype HSI (Saari et al. 2013), which utilises a Point Grey colour sensor with a Bayer pattern. The imager has a C-mount, and the spectral range varies according to the interchangeable low and high band-pass filters and calibration files: VIS from 600 to 650 nm, VNIR from 450 to 850 nm and Red-VNIR from 600 to 1000. The spectral resolution varies from 7 to 25 nm, and the maximum frame rate is 40 frames/s with a 5 ms exposure time. The sensor's resolution is 1920×1200 with 2.3 megapixels. The physical dimensions of the imager are $80 \text{ mm} \times 80 \text{ mm} \times 300 \text{ mm}$, and its weight is less than 1500 g. The example HS data in Section 2.2.3 (Figures 23 and 24) were captured with a laboratory version of a CubeView user interface, which first introduced in Article PVI.

Articles PIV, and PV introduce VTTs small hand-held prototype HSI, developed for skin cancer imaging on complex skin surfaces, enabling 3D skin surface maps. The imager-specific technical details are presented in Article PIV. The specialities of this imager are its LED module, which provides angular illumination, the imagers' small size and adaptive stray light protection cones. The operating principle of the VTTs imagers is explained in the following section.

2.1.8 Piezo-actuated metallic mirror Fabry-Pérot interferometer

The HS imagers used in this study have piezo-actuated metallic mirror Fabry-Pérot interferometers (FPIs) as their spectral separators. Below, Figure 19 visualises FPIs operating principle, which allows one to three transmission peaks into the sensor by changing the FPI air gap between the metallic mirrors.

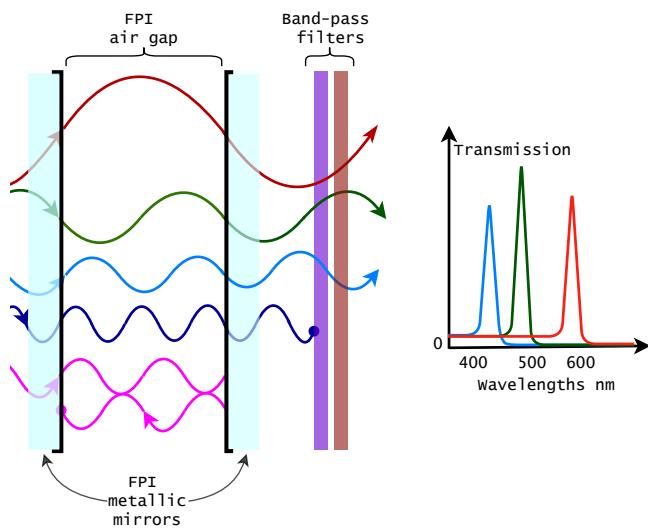


FIGURE 19 Operation principles of the Fabry-Pérot interferometer. The FPI controls the light's transmission to the RGB sensor. Each calibrated FPI position allows one to three transmission peaks, drawn in red, green and blue. Unwanted wavelength transmission is cut with separate long and short pass filters (illustrated by dark blue wave)..

Above, from Figure 19 we can see how the metallic half-mirrors are posi-

tioned parallel to each other. Light enters the FPI, interfering with itself while reflecting from the mirrors. The wavelengths of the light transmitted through the mirrors are integer multiples with constructive interference. These are represented by the red, green and light-blue lines. The unwanted transmission peaks are cut with band-pass filters (dark-blue wave in Figure 19). Opposing phases of light interfere destructively and do not pass through the interferometer (purple wave in Figure 19). The original operating principle with mathematical formulas can be obtained from the work of Perot and Fabry (1899).

2.1.9 Summary: Image capturing process

“Never memorise something that you can look up.” (Albert Einstein)

Although the subsections from 2.1.1 to 2.1.8 are filled with important detailed information on how to design and build a machine vision imager from scratch, it is wise to follow Einsteins’ words and summarise. The main outcome that ought to be memorised is knowing things that should be looked up to:

- EMVA standardised sensors can be easily compared; the vendors’ and manufacturers’ measures and data sheets follow the same mathematical formulas and principles for presenting the measured values.
- Consider selecting a GenICam compatible sensor. This allows all of the features of the sensor to be read, understood and utilised in a significantly more sophisticated manner than, for example, the open CV library. With XML description, the sensor communicates the limits of the features and shows the dependencies between them.
- Select a sensor that suits the application. Its basic features from colours, shutter, resolution etc., should be compliant with the scene.
- Choose optics that match the sensor and remember to re-check the optical system parameters if you use optomechanical components such as extension tubes.
- Write a piece of Python code and start working with the camera: test illumination, imaging setups and find the optimal sensor settings, using Camazing or other suitable backend.

2.2 Computer vision: Image pre-processing

The aim of this section is to briefly introduce the spectral image pre-processing phases that are utilised with VTTs HS imagers. We will learn how digital numbers (DNs) and colour filter arrays (CFAs) can be pre-processed from raw data to HS images.

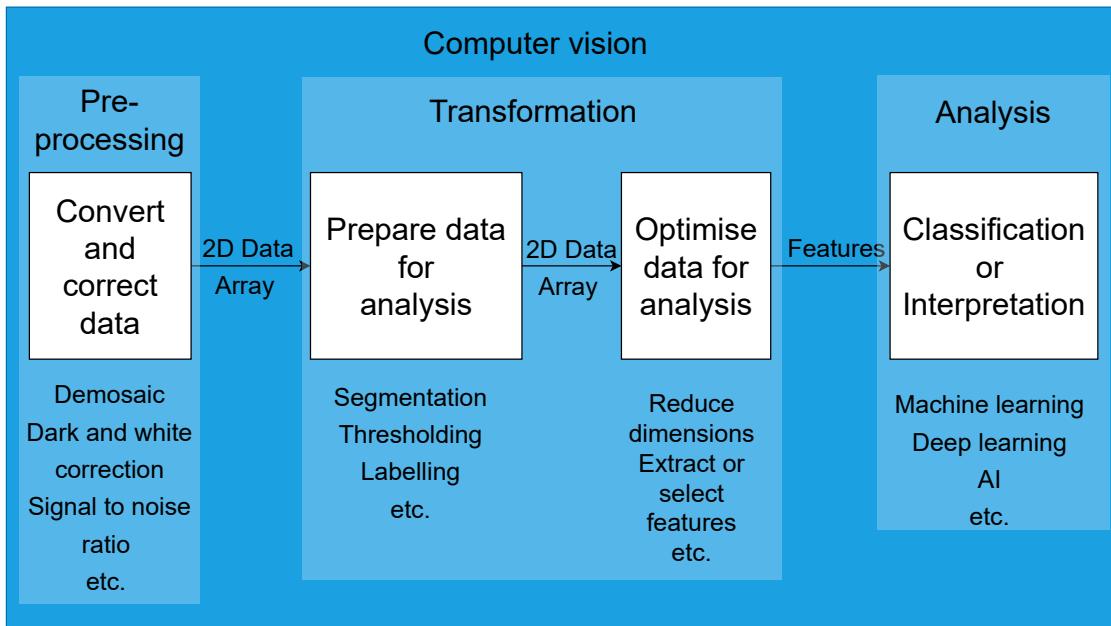


FIGURE 20 Fundamentals of computer vision. Before the results and scene descriptions are obtained, the image data is typically processed through several fundamental phases of pre-processing, transformation and analysis with varying methods. Depending on the scene, optical images are pre-processed (conversions, interpolation and corrections) and transformed; different segmentation, thresholding, labelling, feature extraction (etc.) methods are performed to achieve high accuracy and optimal performance results from the classifier or other interpretation methods.

The fundamentals of computer vision are visualised above in Figure 20. The pre-processing, transformation and analysis processes can be divided into sub-phases, which process the captured data into a form suitable for providing high-accuracy analysis results. The fundamental phases, such as demosaicing or segmentation, should be selected target-wisely and performed only if the analysis methods and the target expect those.

2.2.1 Spectral imaging and image processing

Before capturing the raw data, at least the illumination, sharpness, field of view and exposure time should be inspected. The amount of unnecessary data can be reduced by reducing the width or height of the recorded data, or we can choose to capture only the interesting wavebands if the target phenomena have known spectra of interest.

The raw data in our example consists of a dark current, white reference and raw spectral data. For each captured waveband, our HSI adjusts the FPI's position; after a short settling time, the sensor captures a spectral frame using a given exposure time (Section 2.1.8). The white reference is captured similarly to the spectral data, using the same exposure time but against a white target or illumination source. The dark current is captured with the lens cap on by recording 40 frames from the sensor and using the imaging setup's exposure time; the selected dark frame is the median frame.

The pre-processing phases for the spectral data provided by the monochromatic sensors (DNs) or colour sensors (CFAs) follow a similar workflow. The main difference is in the interpolation, which needs to be performed for the CFA data. Suppose the captured data has unnecessary border areas. In such a case, it might be a good idea to perform interpolation before cropping the image, which ensures the correct pixel channels for the convolutional operation, as presented in Section 2.1.2.5, Figure 11.

After computing the raw data into HS "radiance" or reflectance cubes, they can be evaluated as is. Then, the different pre-processing phases can be performed. For example, the areas of interest can be selected and unnecessary or vignetted areas (Section 2.1.3.4, Figure 14) can be removed. Processing methods, such as segmentation, thresholds, labelling and noise reduction, can be applied, depending on the case and interpolation needs, before or after the data is computed to spectral data.

2.2.2 From raw image to spectral data

At first, the dark current image data is subtracted from the raw data (DN or CFA). Second, demosaicing is performed using bilinear interpolation, since the CFA is a Bayer filter array. After interpolation, the data is weighted by wavelength-specific calibration coefficients and divided by exposure time. The calibration factors were obtained from VTT's calibration measurements (detailed examples of calibration in Article PIV). As a result, we have HS data that we can call the "radiance cube".

Quotation marks surround the "radiance" for a reason. Radiance is the amount of light flux relative to the camera's aperture angle and the field of view, which is relatively challenging to measure and compute. Therefore, the "radiance cube" or I in our measurements and the following denotations represents the wavelengths of light recorded by the sensor frame-by-frame. When we pre-process the captured white reference raw data similarly, we obtain the I_o , which we can call the white-reference cube.

Hemispherical reflectance and lambertian surface

"Lambertian reflectance is a scene property that distributes the energy from any incident illumination into all viewing directions equally" (Koppal 2014). We can address a hemispherical reflectance by assuming the Lambertian surface (simplified visu-

alisation of one pixel presented below in Figure 21a). The components for computing the hemispherical reflectance (I and I_0) are visualised below in Figure 21b.

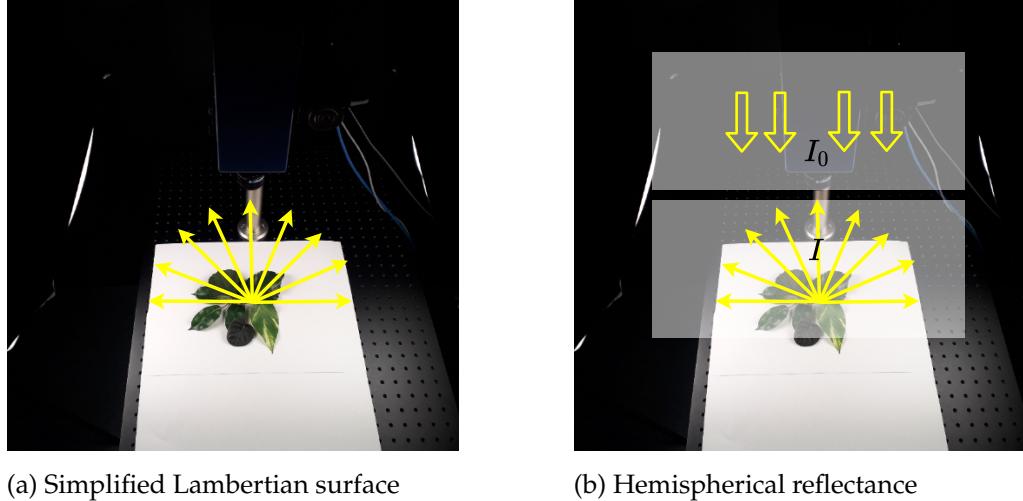


FIGURE 21 Lambertian surface and hemispherical reflectance. Hemispherical reflectance assumes a Lambertian surface. The components for computing the hemispherical reflectance are the recorded flux I and the evenly diffused illumination I_0 .

The hemispherical reflectance R_λ is computed by dividing the captured radiance cube by the white reference cube as follows (Manolakis Dimitris and Thomas 2016):

$$R_\lambda = \frac{I}{I_0}. \quad (3)$$

Hemispherical reflectance is a practical formula for calculating reflectance in basic laboratory settings and has been implemented in the CubeView software presented in Article PVI.

Bi-directional reflectance, transmittance and absorbance

Sometimes, it is impossible to create an evenly diffused illumination for the scene. In such situations, we can use angular illumination and assume a Lambertian surface for a bi-directional reflectance function. Below, Figure 22a visualises the components of computing a bi-directional reflectance, which are recorded light I , the overall flux I_0 and the angle of a flux from the light source to the scene α , which reflects in a certain angle directly to the sensor. The bi-directional reflectance R_{bi_λ} can be defined as follows (Manolakis Dimitris and Thomas 2016):

$$R_{bi_\lambda} = \frac{I}{(I_0 \cos \alpha_{x,y})}. \quad (4)$$

The computation of bi-directional reflectance is more complicated than hemispherical reflectance because the angle varies for different parts of the captured image. Ideally, each pixel spectrum is computed using its specific angle. The

imaging setup would benefit from a point-like illumination over the soft diffuse illumination shown below in Figure 22a.

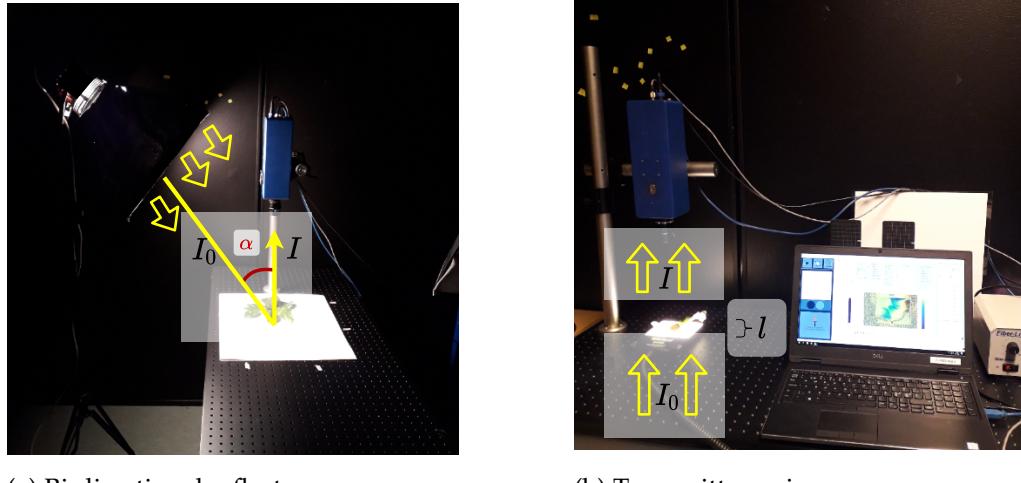


FIGURE 22 Bi-directional reflectance and transmittance imaging. (a) The components for computing the bi-directional reflectance are the recorded flux I , angle diffused illumination I_o , and angle α . (b) According components for computing transmittance are the recorded flux I , angle diffused illumination I_o . The thickness of the measured target is denoted with l ; being needed for computing the amount of concentration.

Figure 22b visualises spectral data imaging setup, where the diffused light reflects underneath, passing through the target towards the sensor. The transmittance denoted as T can be defined as follows (Manolakis Dimitris and Thomas 2016):

$$T = \frac{I}{I_o}. \quad (5)$$

The transmittance imaging setup is interesting, as it enables the data to be computed for the target's absorption and concentration evaluations. Among the radiance, reflectance and transmittance spectra, the absorption spectra (A) is an important spectral signature, describing how the target material absorbs wavelengths of light. For example, in Articles PIV and PIV, the captured wavelengths and LED illumination of the imaging system were selected to represent the known biophysical absorption peaks of human tissue. The formula for A can be derived from the formula for the transmission spectrum:

$$T = \frac{I}{I_o} = 10^{-A}, \quad (6)$$

$$A = -\log_{10} T. \quad (7)$$

As an interesting side note, when the thickness of the measured target (l) is known, and if the target material has a known absorption spectra ε , this measure setup enables the computing of the amount of concentration C , as A can be noted as:

$$A = \sum \varepsilon_i C_i l. \quad (8)$$

2.2.3 Spectral data examples

The visualisations of pre-processed hemispherical reflectance, transmittance and absorbance data can be seen in below collages (Figures 23 and 24). The results were computed using the Equation (3) (hemispherical reflectance, Section 2.2.2) and Equations (5) and (7) (Transmittance and absorbance, Section 2.2.2). The example data is captured using VTT's prototype spectral scanner imager, which is described in Section 2.1.7 and the computation of the examples were performed using Python with NumPy (Harris et al. 2020) and Scikit-learn (Pedregosa et al. 2011) libraries.

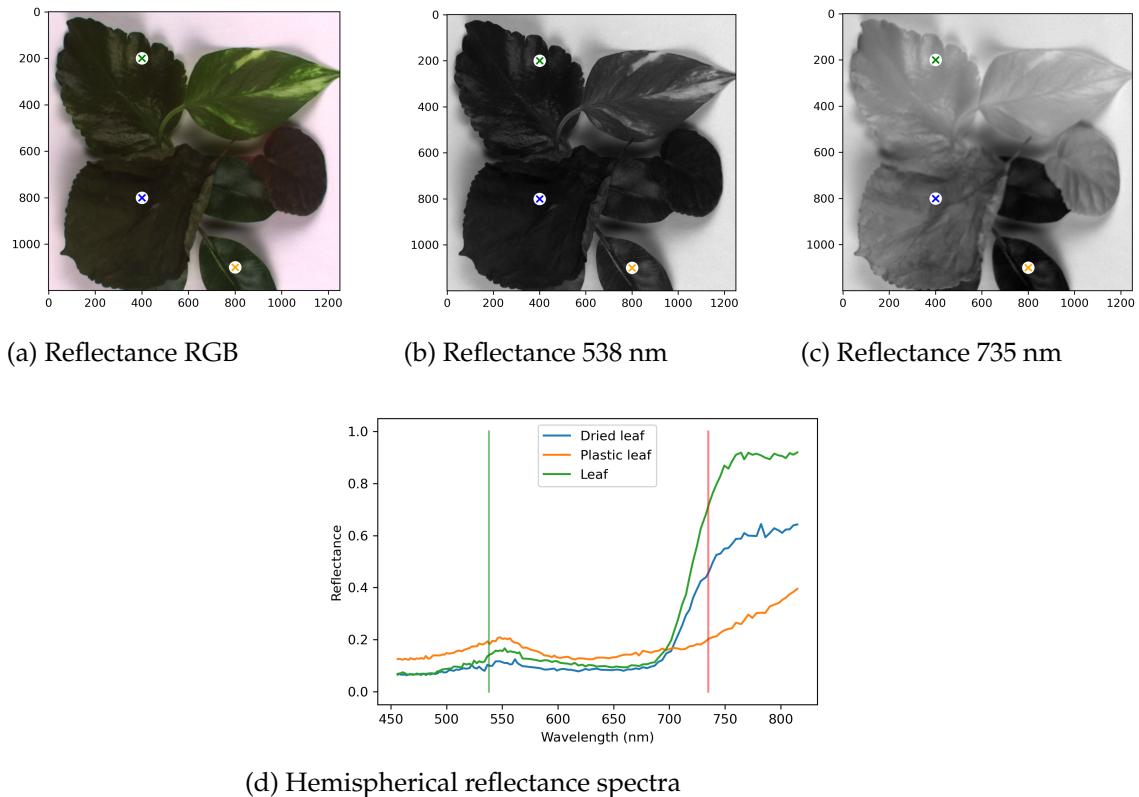


FIGURE 23 Examples of a hemispherical reflectance. (a): an representation of RGB reconstruction of the hemispherical reflectance cube. (b) and (c): visualisations of the 538 nm and 735 nm wavelengths. The selected pixels of the example spectra are marked with x to Figures (a)-(c), the spectra are shown in Figure (d). Blue represent a dried leaf, green is a alive leaf and orange represents a plastic leaf. The wavelengths of 538 and 735 are highlighted with green and red vertical lines.

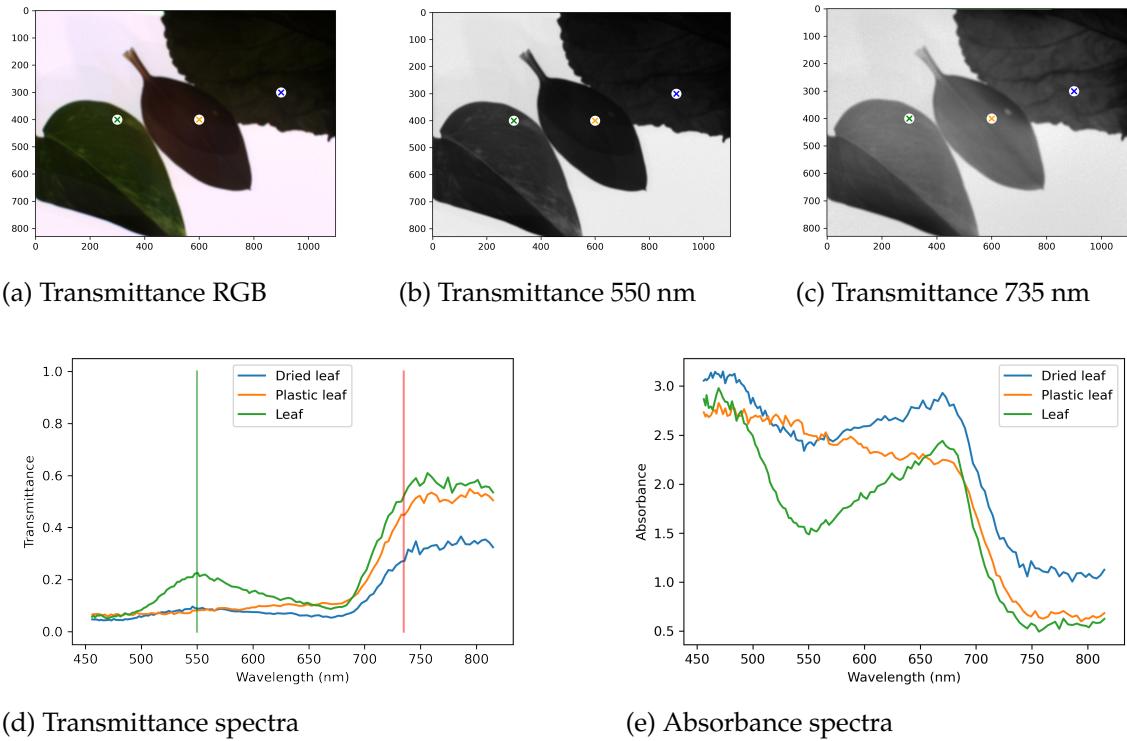


FIGURE 24 Examples of transmittance and absorbance spectral data. (a): an RGB reconstruction of the transmittance cube. (b) and (c): visualisations of the wavelengths 550 nm and 735 nm. The selected pixels of the example spectra are marked with x to Figures (a)-(c), the spectra are shown in Figures (d) and (e). Blue represent a dried leaf, green is a leaf and orange represents a plastic leaf. The wavelengths 538 and 735 are highlighted with green and red vertical lines in Figure (d).

2.3 Computer vision: Machine learning

This section aims to introduce the terminology and theory of the methods used in the articles. Some of the methods are introduced briefly, another more deeply. The level depends on the method's significance in the conducted research.

The essence of computer vision can be summarised to automated extraction of information from digital images (Planche and Andres 2019). The computer sees the images as pixels in matrices, grey scale or red-green-blue values, with no meaning. Computer vision aims to teach computers to find practical information from these pixels and matrices. Nowadays, computer vision is everywhere. For example, industrial control systems, smartphone apps, surveillance systems, medical applications and autonomous vehicles utilize various tasks. Although deep learning has enabled a revolution in computer science, there is still a need for simple, efficient and elegant machine learning methods suitable, for example, to edge computing in real-time applications with platforms constraining the available energy and computational capacity. The articles of this dissertation utilise traditional machine learning models with feature extraction (Article PI), distance-based minimalist machine learning models (Articles (PII; PIII)) and convolutional neural networks (Articles (PIV; PV)).

2.3.1 Exploring the structure and internal organisation of the data

A spectral data have spectral and spatial domains, which provide detailed information on the subject. The spatial domain of an HS image can be considered to be one two-dimensional frame, where the structures formed by the pixels provide information that a human eye can visually interpret. Each frame represents the target, being captured with different intensities of wavelengths of light. The spectral domain comes from the HS image's structure that repeats the frames. The pixel spectrum is drawn from the values of the pixels from the same coordinates through the frames. For example, if the HS cube has 70 frames, each pixel spectrum has 70 values.

A typical challenge in HS image processing is the large amount of data. One HS image can construct from tens, hundreds or even thousands of frames, which can lead to the Hughes phenomenon, whereby, after a certain number of dimensions, the accuracy begins to decrease (Hughes 1969). One known challenge is the redundancy among the samples (Ding et al. 2022). Another practical issue with HS data is that there are structures and internal organisations in multi- or high-dimensional data, but those are not easily visualised. By knowing, which structures describe the targets most, and what the corresponding wavelengths of interest are, we can avoid the Hughes phenomenon, visualise the inner structures and compute resource-wisely. These challenges can be addressed using data transformation. One option is to use computational feature extraction or feature selection methods, for which principle component analysis (PCA) is, despite its age, still one of the most popular algorithms.

PCA has been employed in Article PI. As another solution for decreasing the amount of data, an customisable HSI is introduced in Article PIV, where the wavelengths are selected to represent the known spectral absorption peaks of tissue chromophores; thus, the amount of acquired data is reduced to a minimum during the image capturing phase.

2.3.2 Principal component analysis

PCA, proposed by Pearson (1901) and Hotelling, (1933), is one of the most popular methods in data science. It exploits the idea of dimensionality reduction (Deisenroth et al. 2020). PCA can detect the geometry of data sets from the subspaces of high-dimensional spaces. It is an effective way to find a lower-dimensional representation or best low-rank approximation for the given the data (Calvetti and Somersalo 2020).

The PCA Algorithm decreases the number of parameters that describe one data point and, therefore, can be used for reducing the dimensions in high-dimensional data (Calvetti and Somersalo 2020; Deisenroth et al. 2020). Besides seeking the data structures, PCA can be used as a visualisation tool for high-dimensional data; the data is visualised by projecting them into the directions of maximum spread in a lower dimensional sub space. The PCA is based on linear algebraic foundations, especially on the singular value decomposition (Calvetti and Somersalo 2020). Data variation has an important role while the algorithm searches the principal components (Deisenroth et al. 2020).

Below, Algorithm 1 presents the steps of computing PCA (Karhunen-Loeve, KL-method). We use HS data as an example with a covariance method. With HS data, the aim is to find the most significant frames (representing the intensity of different wavelengths of light) to represent the target phenomena. The spatial dimensions will remain unchanged, but the PCA algorithm can reduce the spectral dimensions. Let $X \subset \mathbb{R}^n$ be an HS data set of $a \times b$ spatial pixels and n wavelengths, where $\mathbf{x}_i \in X$ is a single spectrum.

The goal is to find the subset $X_{sub} \subset \mathbb{R}^p$ that lies in the p dimensional subspace of \mathbb{R}^n . The X_{sub} should be as similar as possible to the data set $X \subset \mathbb{R}^n$, but the dimensionality is decreased from n to p ($n>p$). At first, the data \mathbf{X} is re-organised and normalised, so each parameter falls within the same range. Secondly, in order to maximise the variance of the first principal component's direction, the data \mathbf{X} is centered (Step 2 in below, Algorithm 1 and Figure 25a). The idea of centring with a mean of zero is to find a basis that minimises the data approximation mean square error. The procedure provides deviations that are stored in matrix \mathbf{B} . Next, the covariance matrix \mathbf{C} is needed for the eigenvectors \mathbf{V} and eigenvalues \mathbf{D} (Step 3 and 4 in Algorithm 1). The eigenvectors are orthonormal vectors that form a new basis for the normalised input data (Han and Michelini 2006).

Algorithm 1: PCA with HS data

Input: HS data matrix $\mathbf{X}^{a \times b \times n}$ and value for the cumulative threshold t

1. Initialise

Re-organise \mathbf{X} to contain row vectors $\mathbf{X}^{ab \times n}$

Normalise \mathbf{X}

2. Mean centering

Calculate the empirical mean along each wavelength $n = 1, \dots, n$

$$\mathbf{m}_j = \frac{1}{n} \sum_{i=1}^n \mathbf{X}_{i,j}.$$

Place the mean values into an empirical mean vector $\mathbf{m}^{ab \times 1}$

Calculate the deviations to $\mathbf{B}^{ab \times n}$ from the mean vector:

Subtract \mathbf{m} from each corresponding row of \mathbf{X} .

3. Covariance matrix

Calculate, use a conjugate transpose (*) to find the covariance matrix $\mathbf{C}^{n \times n}$

$$\mathbf{C} = \frac{1}{ab - 1} \mathbf{B}^* \mathbf{B}.$$

4. Eigenvectors and eigenvalues of the covariance matrix

Compute the eigenvectors ($\mathbf{V}^{n \times n}$) and its diagonal eigenvalue matrix ($\mathbf{D}^{n \times n}$) using eigenvalue decomposition:

$$\mathbf{C} = \mathbf{V} \mathbf{D} \mathbf{V}^{-1}.$$

5. Sort

Re-order pair-wise eigenvalues and eigenvectors to descending order of eigenvalues.

6. Evaluate the number of components

Compute cumulative energy of the eigenvalues contents (e) among the eigenvectors from 1 through n :

for each $j = 1, \dots, n$

$$e_j = \sum_{i=1}^j \mathbf{D}_{i,i}$$

end for.

7. Select the components and project onto the new basis

Select the p number of components from $\mathbf{V}^{n \times n}$ to the new matrix $\mathbf{W}^{n \times p}$

$\mathbf{W}_{kl} = \mathbf{V}_{kl}$, for $k = 1, \dots, n$ and $l = 1, \dots, p$,

using the cumulative energy e and the the threshold t :

$$\frac{e_p}{e_n} \geq t.$$

Output

Compute and return the projected pixel spectra of selected p principal components $\mathbf{X}_{sub}^{ab \times p}$:

$$\mathbf{X}_{sub} = \mathbf{B}^{ab \times n} \mathbf{W}^{n \times p}$$

Each column of \mathbf{X}_{sub} is the projection of the data points onto the indexed principal component in descending order.

After the eigenvalues and eigenvectors are sorted in descending order of eigenvalues significance (Step 5 in above Algorithm 1), the number of significant components can be evaluated (Step 6 in Algorithm 1). The cumulative energy of the eigenvalues (e) measures the variance among the selected components with respect to the variance of all components. Finally, the deviation matrix \mathbf{B} and the selected components \mathbf{W} , which refer to the maximum variance of the target phenomena, can be used to project and visualise the data \mathbf{X}_{sub} in a lower-dimensional sub-space, which is computed to represent the directions that maximise the spread and variance (Step 7 in Algorithm 1 and Figure 25b (below))

(Calvetti and Somersalo 2020).

Below, Figure 25 explains the effects of the mean centring and projection; the centering is a key to quide the first and second principal component correctly.

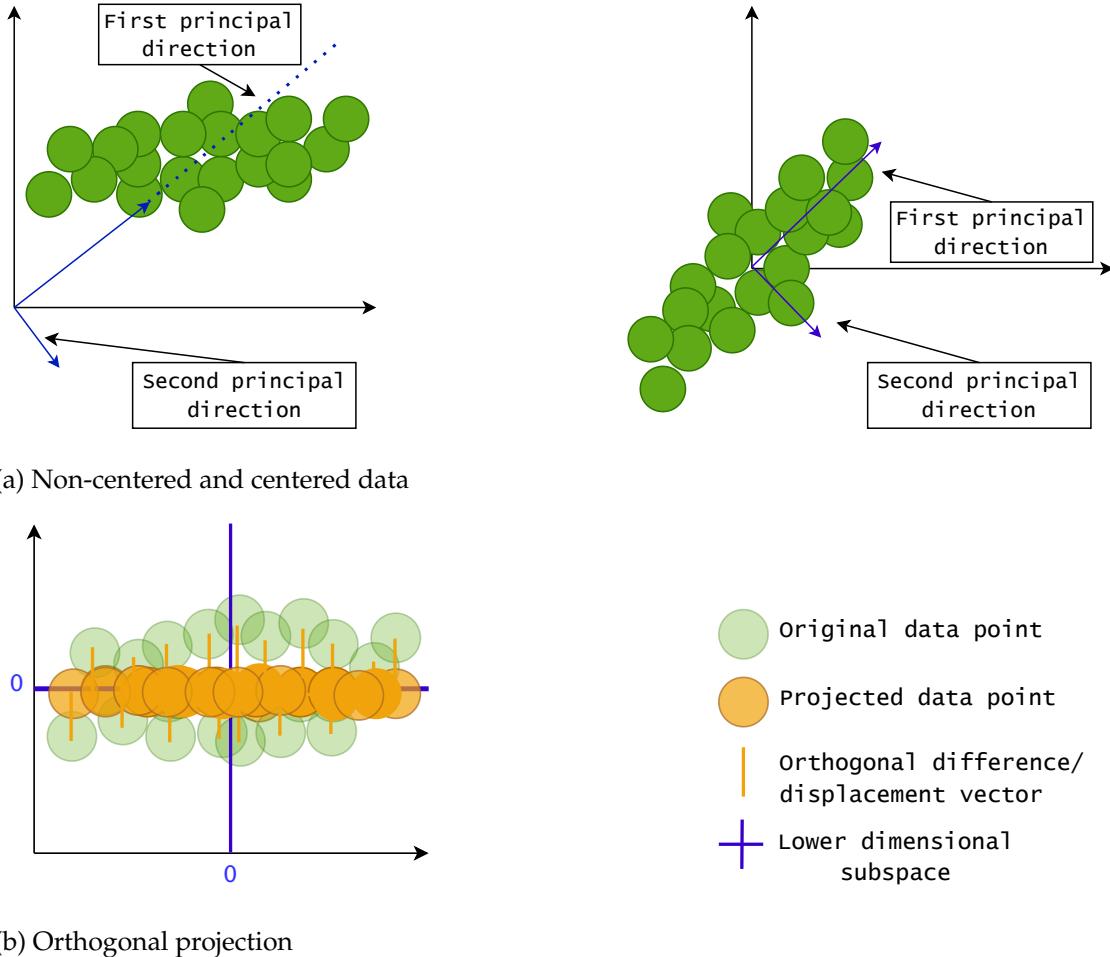


FIGURE 25 Effect of centring and projecting data. The left side of Figure (a) show the principal direction in which the data points lie (marked with a dashed line) and its orthogonal direction when the data is not centred. The centred data behaves differently. The top right image shows how moving the coordinate origin to the centre of the mass data guides the first and second principal directions correctly. The orthogonal projection is visualised in Figure (b). The original green data points are projected through orthogonal displacement vectors to the lower dimensional subspace (orange points). The displacement vector $\mathbf{X}_{sub_i} - \mathbf{X}_i$ describes the difference between the original coordinates and projection coordinates of the data point.

2.3.3 Computer vision's main tasks and machine learning terminology

In order to follow the theory there exists some basic terminology, which helps to understand and learn. This subsection lists and briefly describes the main tasks and terms of computer vision. Similarly, with the approaches in the included articles, some of the listed tasks can be implemented and completed with minimally complex methods, whereas some require more complicated deep learning

approaches.

Computer vision's main tasks

Regression is a technique aiming to solve the relationship between independent variables or features and a dependent variable or outcome. Usually, regression methods are used for predictive modelling in machine learning, especially for continuous outcomes (Deisenroth et al. 2020).

Content recognition aims to find a *sensible* information from images for object or image classification. This can be understood as a task of finding the right labels for certain pixels, images or objects in images. Object identification or instance classification aims to recognise specific class instances, such as a certain person from an image or specific pixel spectrum, which denotes a certain material (Goodfellow et al. 2016).

Object detection focuses on specific elements in an image: anomalies such as damaged components in industrial plants can be found. Similarly, the detected elements can be used as thresholds by forming bounding boxes or delineating areas to crop images before further analysis (Goodfellow et al. 2016).

Image segmentation methods can be used to form masks of labelled pixels belonging to certain instances of a class. According to Planche and Andres (2019), the segmentation task is challenging even for deep learning neural networks, which is still one of the most important elements for autonomous devices to understand the surroundings.

Besides classification and object detection, computer vision methods can estimate an object's or its sub-parts' orientations and poses. Video analysis enables these tasks in action, providing solutions, for instance, tracking, action recognition and motion estimation. The most demanding tasks relate to scene reconstruction, where the 3D geometry of a scene is retrieved from images, for example, by finding the corresponding objects from several images to derive the geometry and scale for the scene (Planche and Andres 2019).

These tasks can be accomplished with different kinds of image data. A classic approach is an image-wise classification of RGB, greyscale or even video images. However, with high-dimensional data, such as HS images, the typical approach for computer vision tasks is pixel-wise (Ahmad et al. 2022). Every pixel with its spectrum is evaluated and, for example, labelled to represent classes or denoted as anomalous.

Machine learning terminology

Machine learning methods can be divided based on the learning approaches to *supervised*, *unsupervised* and *semi-supervised* methods. Supervised methods need the training data to be fully labelled. Semi-supervised methods combine a small set of labelled data with a large amount of unlabelled training data, and unsupervised methods do not need labels for training the models (Deisenroth et al. 2020).

A *Binary classifier* is a classifier which typically seeks a rule to partition the data points into two classes (Calvetti and Somersalo 2020). A *Multi-class classifier* can be an algorithm designed for multi-class classification or an extension of a binary classifier, organised to classify multiple classes with one against all other principles. A multi-class classifier handles data that have more than two output classes.

A *Distance-based classifier* classifies the objects by their dissimilarity, which is measured by distance functions (de Souza et al. 2015). For example, the Minimal Learning Machine, introduced in Articles PI; PII and PIII is easy to implement, fast and effective, remaining relatively uncomplicated compared to many other classification methods used in this dissertation.

Term *bias* has several meanings in machine learning. With linear models, such as ($y = kx + b$), b represents the data's tendency to have a distribution in a given value that is offset from an origin. The bias is considered by including the constant b in the model (Bishop and Nasrabadi 2006). With artificial neural networks, the neuron's bias is used to control when the activation function is used; b shifts the curve left or right for delay or acceleration, affecting the model's capability to fit. Besides being a constant that influences the models, bias is a term for measuring how far the average prediction results are from the average labels of the dataset (bias/variance trade-off). The fourth worth mentioning is the data-driven bias. In this context, the prediction results might have been affected by data-driven bias, which can be, for example, an underlying feature that comes from the sample selection procedures. In this dissertation, the term bias is used as a variable with machine learning models, and in Article, PIV, the discussion is on the data-driven bias.

2.3.4 Random forest

Random forest is a decision tree-based branching algorithm. Branching algorithms consist of a flow of simple yes/no questions until the final decision is ready (Calvetti and Somersalo 2020). The branching approach is typical in medical applications, where it is common to delineate the decisions by excluding the options. The branching approach forms a tree structure (see below, Figure 26), which is effective and useful for classification tasks (Calvetti and Somersalo 2020).

The random forest algorithm was introduced in 2001 by Leo Breiman. Since the early days, the method has been extremely popular as a general purpose classification and regression method (Biau and Scornet 2016). The idea is to combine several randomised decision trees to aggregate the predictions by averaging. The excellent performance of the method is significant, for example, in tasks where the number of observations is significantly smaller than the number of variables (Biau and Scornet 2016). Random forest can be applied to large-scale problems relatively fast and straightforward by its principles. The downside is that the tree-wisely divided inference of the decisions might be difficult to explain, and the training or optimising phase might be time-consuming.

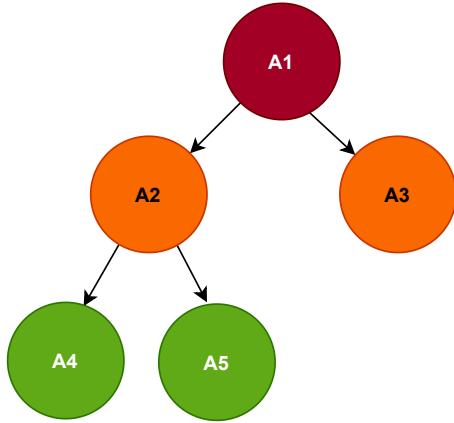


FIGURE 26 A simple tree structure formed with a branching algorithm. A1 is the root node, while A2 and A3 are the children nodes of A1. The nodes A4 and A5 are the leaves. Starting from the root node, a branching algorithm utilises a flow of yes/no questions until the final decisions (the leaves) are reached. A Random forest consists of multiple of branching trees.

Random forest classifier is a supervised method that creates artificial training and validation data ensembles. Intuitively, the algorithm grows decision trees using a different subset of the training data, but not exclusively; the trees have duplicates in their training sets. The diversity of features is ensured by bootstrapping; randomly selecting a subset from all of the data features (Calvetti and Somersalo 2020). The random forest constructs a forest of trees using the training data and selects the optimal trees among the pruned trees using validation data (Calvetti and Somersalo 2020). While growing the trees, the splitting decisions are optimised using the features. The splitting is repeated with each tree until the information from the trees is aggregated to final predictions. The algorithm uses mode or majority for classification decisions, and the average predictions of each tree are used for the regression tasks (Biau and Scornet 2016). The random forest has several parameters that can be optimised with cross-validation methods; the maximum depth of a tree, minimum samples of records in a node (in order to proceed with a split), the number of trees in a forest and the number of features per tree for bootstrapping (Biau and Scornet 2016; Calvetti and Somersalo 2020).

2.3.5 Support vector machine

The support vector machine (SVM) was introduced in 1992 by Vapnik et al. (Han and Michelini 2006). The method is suitable for linear and non-linear data classification. Intuitively, the method transforms the original data into a higher dimension and separates the data into classes with optimal hyperplane(s) (Han and Michelini 2006). The support vector machine and its kernel variations are very popular methods, being utilised in various of applications, from revealing credit card frauds from activity reports to different functions in the medical field (Noble 2006). The kernel and hyperparameters enable the linear support vector machine to adopt non-linear data classification.

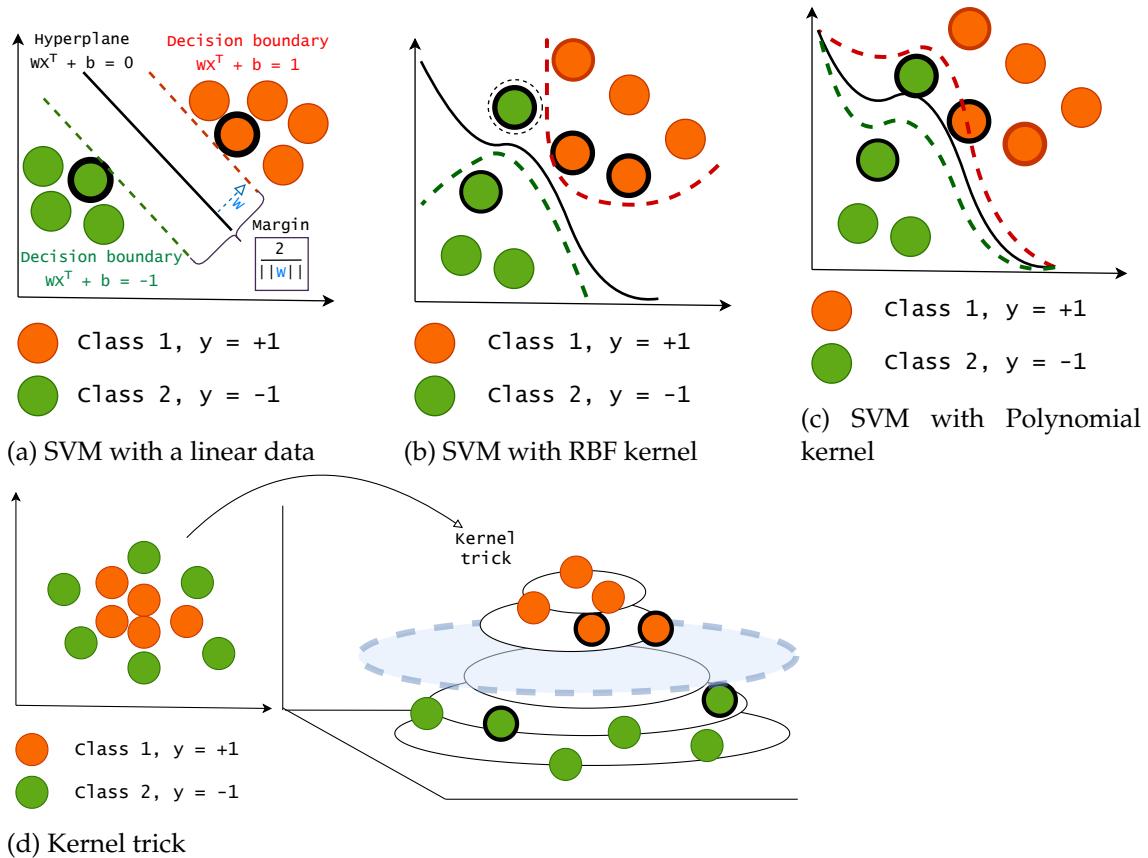


FIGURE 27 SVM and kernel SVM variations Subfigure a: SVM with a linear data. The hyperplane separates two classes with a largest possible margin. The support vectors are the data points with a stronger outline. Subfigures b to d visualises kernel SVMs. By using a kernel function, we can classify non-separable data.

Above, Figure 27 visualises three support vector machine versions and a kernel trick. Figure 27a presents the support vector machine with linear data. The decision boundaries are drawn with dotted lines, and the hyperplane is in the middle of the decision boundaries. The data is divided into two classes with the largest possible margin. Figures 27b and 27c are examples of kernel support vector machines, which can classify nonlinear data. A radial basis function (RBF) with hyperparameter gamma enables the decision boundaries to delineate non-separable data. Figure 27c visualises a polynomial kernel. Figure 27d visualises how the data can be projected using kernels onto higher dimensional space to define the hyperplane (marked in light blue). The support vectors are drawn as data points with stronger outlines. The reference support vector machine in Article PI was optimised to use the RBF kernel.

2.3.6 Artificial neural networks

We will start the introduction to neural networks by reminding the basic terminology (Section 2.3.6.1) and then, we will continue to visualisations and discussion of the basic operation principles of an artificial neural network (ANN) (Sec-

tion 2.3.6.2).

2.3.6.1 Artificial neural network related terminology

A *Feed-forward neural network* is an multi-layer ANN consisting of an input layer, at least one hidden layer, and an output layer (Han and Michelini 2006). The layers are constructed from neurons, which pass on their outputs, updated weights and biases to the neurons in the next layer. The network is called a feed-forward, if none of the weights are cycled back to the inputs. Figure 30 in Section 2.3.6.2 shows an example of a feed-forward neural network.

An *Activation function*: ANNs consists of neurons, which use the inputs, weights, biases and activation functions to obtain the output, which can be then passed on to the following layers (Planche and Andres 2019). As an example of an activation function, we can examine prediction $y(\mathbf{x}, \mathbf{w})$ ($y \in \mathbb{R}$), obtained with linear input variables \mathbf{x} , parameters \mathbf{w} and discriminant linear model $y(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + b$. This approach may be valid for a simple regression task, but classification aims to predict discrete class labels or posterior probabilities lying in the range (0,1). Therefore, the model should be generalised by transforming the linear function \mathbf{w} using a nonlinear function $f(\cdot)$ (Bishop and Nasrabadi 2006):

$$y(\mathbf{x}) = f(\mathbf{w}^T \mathbf{x} + b), \quad (9)$$

Where $f(\cdot)$ is the activation function, \mathbf{x} is the input data and b is the bias. Now, the decision surfaces correspond to $y(\mathbf{x})$ which is constant similarly with $\mathbf{w}^T \mathbf{x} + b$, thus the decision functions of x remain linear, despite the use of nonlinear $f(\cdot)$. The models with nonlinear activation functions are called *generalised linear models* (Bishop and Nasrabadi 2006). Activation functions add non-linearity to the models, which can be used for sparse activation decisions in deep feed-forward neural networks.

The choice of activation function should be determined by the data and its assumed distribution of the ground truth values (Han and Michelini 2006). A linear model $f(\mathbf{x}) = \sum_{i=1}^n x_i w_i + b$ is basically a linear activation function, or sometimes referred as *no activation function*. The output of the linear activation function can be any numerical value, so in order to evaluate the probability or likelihood, we can use *sigmoidal* activation functions as $f(\cdot)$ in Equitation (9). Sigmoidal activation functions returns, regardless the inputs, values between 0.0 and 1.0 or -1.0 and 1.0.

Sigmoidal functions are suitable for binary or multi-label classification (Han and Michelini 2006). They are typically used in the hidden layers; but in an exclusive multi-class classification, the choice for the output layer could be *softmax*. Softmax can be seen as a generalisation of a sigmoid function, representing the distribution over n classes. Softmax is typically used in the output layer; however, being suitable for hidden layers (Goodfellow et al. 2016).

Besides softmax and sigmoidal functions, a *rectified linear activation function* *ReLU* is commonly used in the context of deep neural networks. ReLU is a piecewise function that will directly pass positive output values to the next layers.

Otherwise, the value is set to zero (0) and the output is not passed forward. This will make the network sparsely connected, decreasing the training time and decreasing the computational complexity (Goodfellow et al. 2016; Han and Michelini 2006).

A *Loss function* calculates the error between the predicted result and the ground truth. The loss function should be selected task-wisely from the regression and classification loss function algorithms (Goodfellow et al. 2016).

Optimisation functions are used to calculate the partial derivative of loss functions with respect to weights. After the gradient is determined, the weights and biases are updated in the opposite direction. The current error in the results is propagated backwards, and the weights and bias are optimised to minimise the error, which enables the model to learn from the training data (Goodfellow et al. 2016). Typically he optimisation is repeated until the minima of the loss function is reached.

A variable called *learning rate* helps a method to set the weight and bias increments to serve fast learning (Han and Michelini 2006). The learning rate is usually set between 0.0 and 1.0. Too small learning rate causes slow learning, and too big values cause divergence (Duffner and Garcia 2007).

One round of computing all the layers of a feed-forward network to output results is called an *epoch*. The training data can be divided into *batches* that are passed through the model during the epoch. The backpropagation with a loss function and optimiser can be performed batch-wisely or epoch-wisely, with or without a rule; for example, until the results obtain a satisfying error level (Goodfellow et al. 2016; Han and Michelini 2006).

An *Adam optimizer* was used in Articles PIV and PV. Adam is an adaptive gradient descent algorithm that combines two advantages. It can deal with sparse gradients with a per-parameter learning rate and handle online and non-stationary problems (for example, noise) with Root Mean Square Propagation (RMSProp). RMSProp adapts per-parameter weights based on the average of recent magnitudes of the gradients for the weight (changing speed). Adam is a straightforward method, suitable for problems with a large amount of data and parameters (Kingma and Ba 2014).

Dropout was first introduced by Srivastava et al. (2014). The motivation of using dropout is to prevent the overfitting by randomly dropping neurons and their weights from the network. This way, we have a sparsely connected network during the training. For example, the implementation of Keras (a Python deep learning API), selects randomly nodes to be dropped out with a given probability (e.g., from 20% to 50%) in each weight update cycle. The dropout can be performed with incoming (visible) and hidden layers Planche and Andres (2019).

Max Pooling is typically used after activation function in convolutional layers. The convolutional layer has provided a feature map, which is down-sampled to represent the most significant features. Max pooling emphasises the features and increases the robustness of certain feature position changes in the images. Below, Figure 28 visualises how 2×2 filter down-samples the 4×4 output feature map into 2×2 matrix.

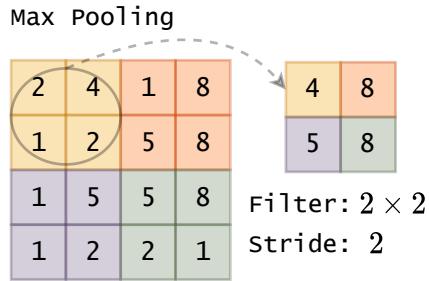


FIGURE 28 **Max Pooling** down-samples the results to represent the most significant features. The 4×4 matrix is down-sampled to 2×2 matrix using 2×2 filter that outputs one value for each stride.

2.3.6.2 Artificial neural networks operation principle

A perceptron (below, Figure 29) is a single computation unit of an ANN. It consists of a neuron, inputs, weights (gradients of coefficients of each variable) and bias values. The neuron will first compute the sum of the inputs, weights and biases. The nonlinear transformation activation function $f(\cdot)$ is applied in the second step before sending the output to the next layer.

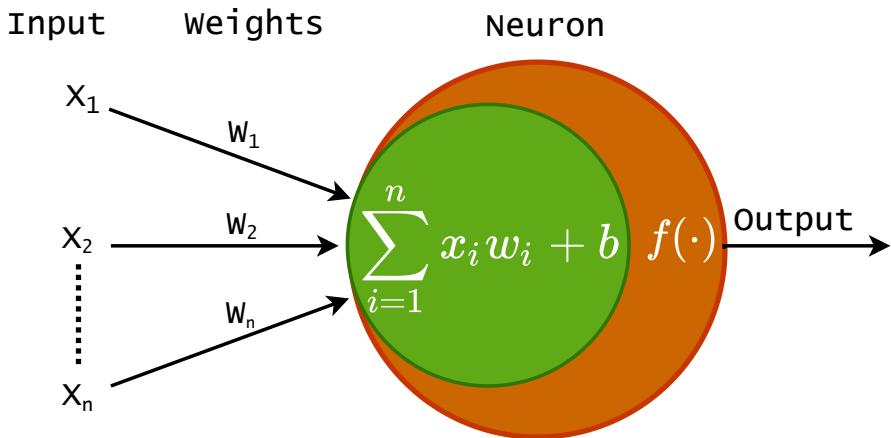


FIGURE 29 Perceptron, a single computation unit of a neural network.

A multilayer perceptron (below, Figure 30) consists of several perceptrons grouped for a layered neural network. An artificial neural network typically has a feed-forward structure, with data-wise selected activation, optimiser and loss functions. Figure 30 is an illustration of a fully connected (dense) artificial neural network, as all the neurons in the hidden layers are fully connected (Planche and Andres 2019). The neurons in the first hidden layer receive four inputs from the input layer, and the second hidden layer's neurons receive three inputs from the first hidden layer. Therefore, each neuron in the previous layer is fully connected to the neurons in the next layer. This structure is an advance in many classification cases, but it might cause challenges with image data (Planche and Andres 2019).

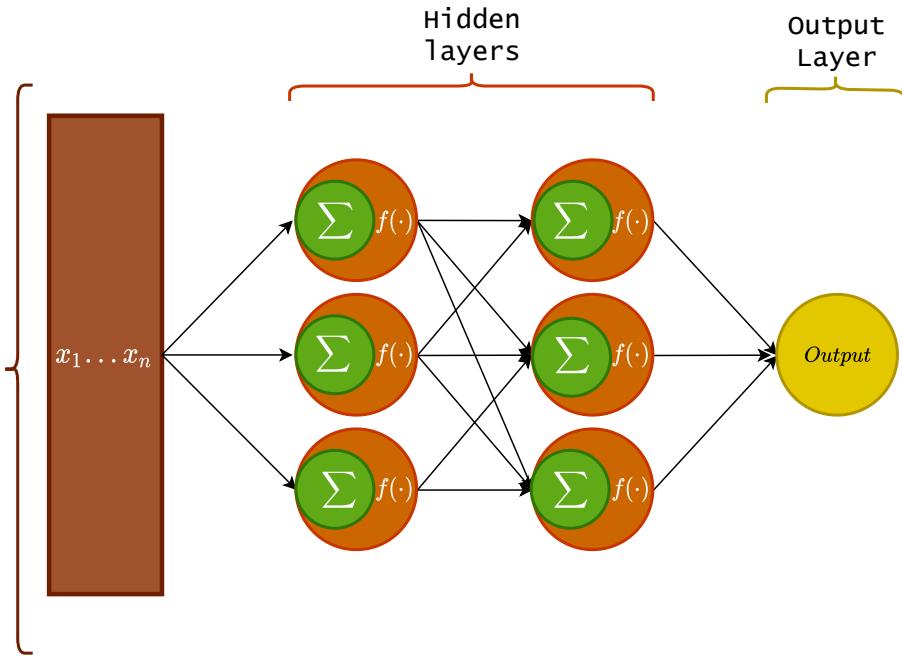


FIGURE 30 An example of a multilayer perceptron (ANN). It consists of an input layer and two hidden layers with three hidden neurons (perceptrons or nodes). The output layer has one neuron. In this example, each of the neurons in the first hidden layer receives the x vector as training input.

Typical image data have complex three-dimensional structures of the form; $H \times W \times D$, where H is the height, W is the width, and D represents the number of channels. The challenge for dense artificial neural networks is that large or high-dimensional image data might have an explosive number of parameters that need to be optimised within the hidden layers, making the dense networks computationally intensive (Planche and Andres 2019). Another challenge is the lack of spatial reasoning. The fully connected neurons receive all the values from the previous layers without the notions of distance or spatiality; therefore, the spatial information is lost. Multidimensional data suffer from losing the notion of proximity between the pixels since in the fully connected layers, all pixel values are combined regardless of their original positions (Planche and Andres 2019). Hence, it would be beneficial for image classification to take the spatial domain into account and use the neighbourhood information, which is one reason for introducing convolutional layers with dense layers.

2.3.7 Convolutional neural network

Convolutional neural networks (CNNs) offer solutions for the afore mentioned image classification challenges. The CNNs operate similarly to ANN's feed-forward and backpropagation procedures. However, the architecture is designed to handle multidimensional data, and the way the neurons are connected to the previous layers enables access to the neighbouring region's spatial information. Unlike ANNs, CNNs are taking advances on multidimensionality and local connectivity (Planche and Andres 2019).

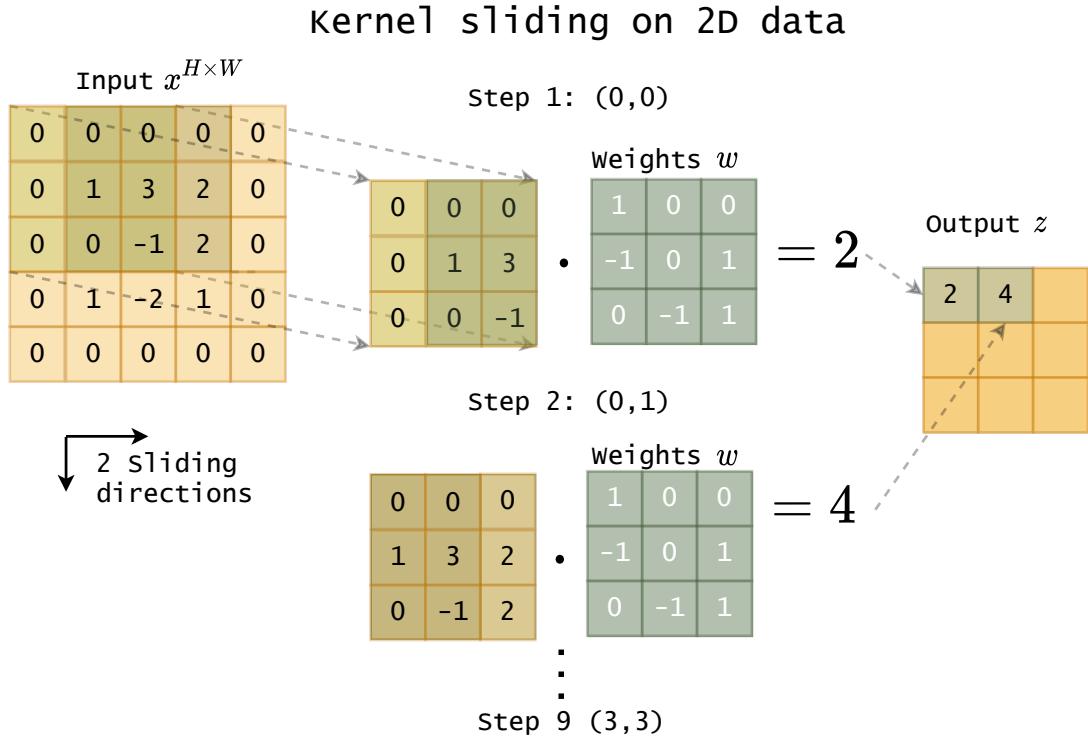


FIGURE 31 **The convolution operation** is performed at once by sliding neurons over the input matrix. The neuron can be seen as a kernel ($H_a \times W_a$), which contains the weights (w). The dot product of the input matrix at the current position and the filter matrix is computed for each step to the output response map z . The example is two-dimensional for simplicity (2D CNN), but convolution operations can be performed with high-dimensional data.

Convolutional layers reduce the number of parameters by sharing the weights and biases from the neurons connected to the same output channel. Intuitively as visualised above in Figure 31, the neuron slides over the whole input matrix with specified kernel (filter) size and steps. As an example, for image data X with dimensions $H \times W \times D$, we can define the neuron to use kernel size ($k_H \times k_W$). This limits the inputs the neurons receive from the previous layers to $k_H \times k_W \times D$, providing spatial information with limited spatial connectivity. The neuron slides through the whole input X , step-by-step. After the X is covered, the neuron combines all input values linearly, and the activation function is applied, in a manner that is similar to that for artificial neural networks.

The neuron's output at $z_{i,j}$ with the input patch starting at position (i,j) can be computed as follows (Planche and Andres 2019);

$$z_{i,j} = \sigma \left(\sum_{l=0}^{k_h-1} \sum_{m=0}^{k_w-1} \sum_{n=0}^{D-1} w_{l,m,n} \cdot x_{i+l,j+m,n} + b \right), \quad (10)$$

where $w \in \mathbb{R}^{k_h \times k_w \times D}$ is the neurons weights, the neuron's bias is $b \in \mathbb{R}$ and σ is the activation function. This operation is repeated with every step of the sliding

window until the response matrix $\mathbf{Z}^{H_a \times W_a}$ is obtained. The dimensions $H_a \times W_a$ are the number of times the neuron slides vertically and horizontally over the input tensor (a multi-dimensional matrix). The output tensor of a convolutional layer ($H_a \times W_a \times N$) is constructed from stacked response maps, obtained with shared parameters by N sets of neurons.

To be accurate, above Figure 31 visualises mathematical operation *cross-correlation*, which is commonly called as *convolution* in the machine learning communities. The convolution of matrix x with filter w ($\forall i \in [0, H_a - 1]$ and $\forall j \in [0, W_a - 1]$) is defined as follows (Planche and Andres 2019):

$$(w * x)_{i,j} = \left(\sum_{l=0}^{k-1} \sum_{m=0}^{k-1} w_{l,m} \cdot x_{i+l, j+m} \right). \quad (11)$$

Depending on the activation function, some of the neurons with negative values can be set to zero (0), preventing the neurons from sending unimportant feature maps forward. Before forwarding the positive-valued output, the max pooling algorithm down-samples the provided feature map. As a result, the features are extracted from images with respect to the spatial and spectral domains.

Since the connections between layers are limited, a convolutional layer has fewer operations to perform than a dense layer. For example with an N set of neurons, N weight matrices ($D \times k \times k$) and N biases the layer has $N \times (Dk^2 + 1)$ values to train. An artificial neural network with a dense structure has $(H \times W \times D) \times (H_a \times W_a \times D)$ parameters to work with (Planche and Andres 2019).

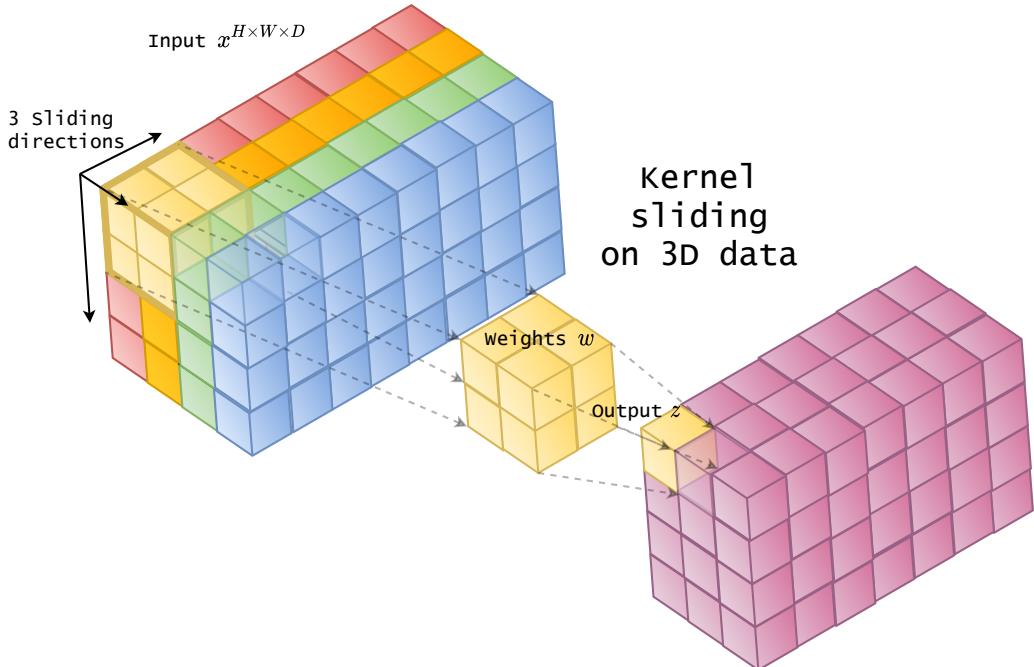


FIGURE 32 3D CNN. The kernel slides through the data using three directions and provides feature maps similar to the above-discussed 2D kernel. The difference between 2D and 3D is the number of sliding directions, which provides more information throughout the dimensions.

Above, Figure 32 visualises the convolutional operation with a 3D kernel. The kernel has three dimensions and three sliding directions, the operations are similar to 1D and 2D kernels, but with 3D kernel, the neighbourhood information and multi-dimensionality is taken into account.

2.3.8 Minimal learning machine

Minimal learning machine is a distance-based supervised method for classification and regression. Intuitively, the method assumes linear mapping between the input and output value distances. It is a computationally efficient and fast machine learning method (de Souza et al. 2015). The prediction can be seen as a multilateration problem, where quadratic optimisation can be used, or since the distances of the new labels are known, the prediction can be extended to the nearest neighbour generalisation (Mesquita et al. (2017), Article PI).

Algorithm 2: MLM training	Algorithm 3: MLM prediction
Input: X, Y, k Output: \hat{B}, R, T <ol style="list-style-type: none"> 1. Form subsets R and T; select randomly k reference points from X to R and their corresponding reference labels from Y to T. 2. Compute D_x distance matrix between R and X 3. Compute Δ_y distance matrix between T and Y 4. Calculate $\hat{B} = (D_x^T D_x)^{-1} D_x^T \Delta_y$. 	Input: R, T, \hat{B} , new data x_i Output: distances $\delta(y_i, T)$ <ol style="list-style-type: none"> 1. Compute $d(x_i, R)$ 2. Compute $\delta(y_i, T) = d(x_i, R) \hat{B}$ 3. Argsort $\delta(y_i, T)$ 4. For a new data x_i, assign n closest labels from labels T.

Above, Algorithms 2 and 3 show how the MLM model \hat{B} is trained and how the label of the new data point x_i is predicted (de Souza et al. 2015; Mesquita et al. 2017). The prediction assumes linear mapping between the input and output distance matrices, which is visualised below, in Figure 33, where we have nonlinear example data with two classes and reference points (R). The first coordinates represent the distances between all data points (X) and reference data points R (distance matrix D). The second coordinate behind represents the corresponding distances between Y and reference labels T (distance matrix Δ). The model \hat{B} can be seen as a plane representing linear mapping.

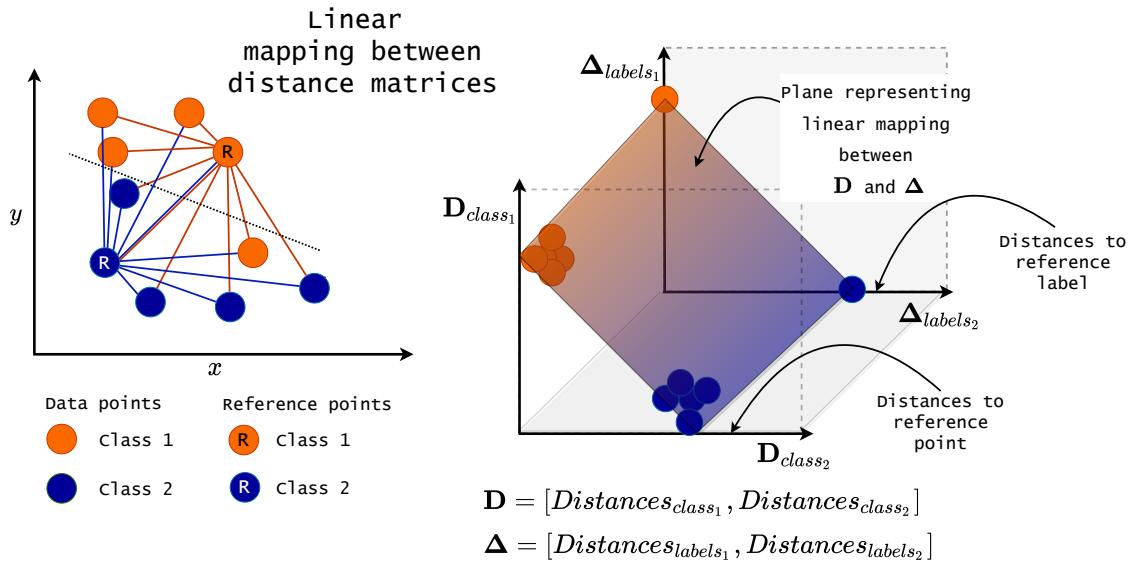


FIGURE 33 Minimal learning Machine assumes a linear mapping between the input and output matrices. Distance matrix D has the distances from each reference point R_n to all the training set's data points. Matrix Δ contains the corresponding distances between the reference point's labels to all of the training set's labels.

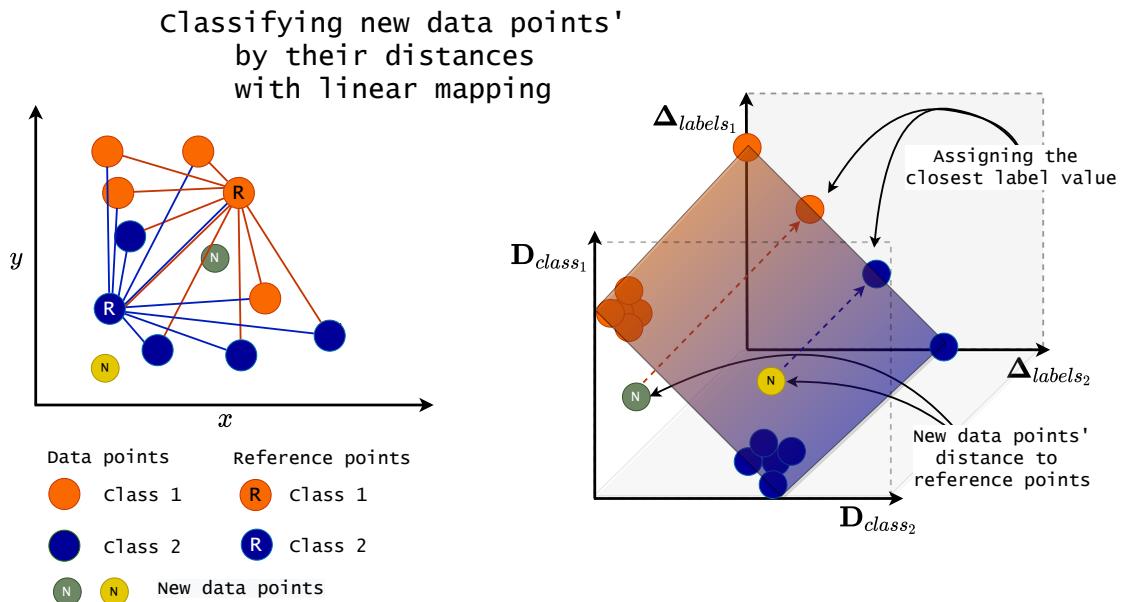


FIGURE 34 Classifying data points with minimal learning machine. MLM utilises linear mapping with new data points. The algorithm calculates distances between the new data point and reference points and uses a linear model \hat{B} for mapping the input and output distances. The labels of the new data points are selected based on the nearest neighbour labels.

Above, Figure 34 show how two new data points are classified utilising the model \hat{B} . The model does not reveal the output labels. However, the locations of the output label values are known similarly to the reference point locations;

thus, the data point labels can be assigned. The output distances $\delta(y_i, T)$ are argument sorted, and the closest labels are selected from the T , based on the nearest neighbour values.

2.4 Machine vision: Imaging systems

This section aims to combine the image capturing process phases from the fundamentals of machine vision to the SICSURFIS imager concept and the CubeView Hospital user interface (related to Articles PIV and PV). We will see examples of imaging system's target-wisely designed instrumentation, imaging and device controlling logic and user interface workflows.

2.4.1 Design principles, environmental issues and user needs

The design principle was found at the very first technical project meeting. One of the authors' colleagues said "*Just click and the magic happens*". The idea of hiding all the controls from the user was fascinating. Controlling multiple devices and the complex workflow requirements of the photometric stereo imaging needs might have to lead the design towards "*an easy to create but a complex to use*" solution.

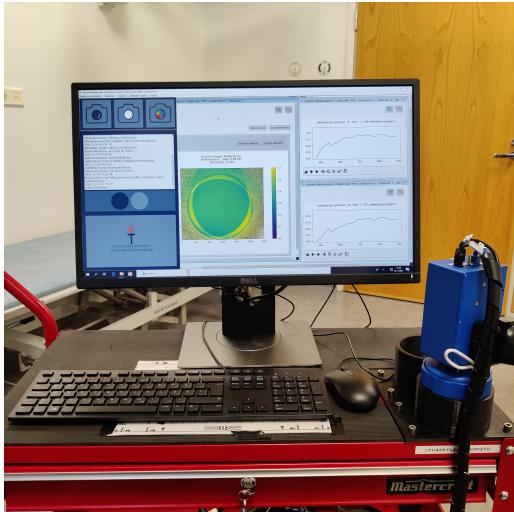
The data-gathering environment was a hospital; the users were medical doctors and nurses. Therefore, the author decided to have a user-oriented design process and visit the scene. After meeting the users, and the environment and discussing the possible use-cases, it was clear that the overall system should be as simple as possible and portable so it could be moved between hospital wards and stored securely while unused. The computer should not be connected to the hospital internet due to patient information security risks. The system's data should not contain patient information besides the patient key, which connects the collected data to the ground truth. The user interface should guide the user through the imaging process, and finally, the imaging process should not take the attention away from the patient work.

All those needs could be achieved with a carefully designed user interface and imaging system by simplifying the imaging setup and pre-setting device parameters, aiming to create a simplified "*plug and play*" experience.

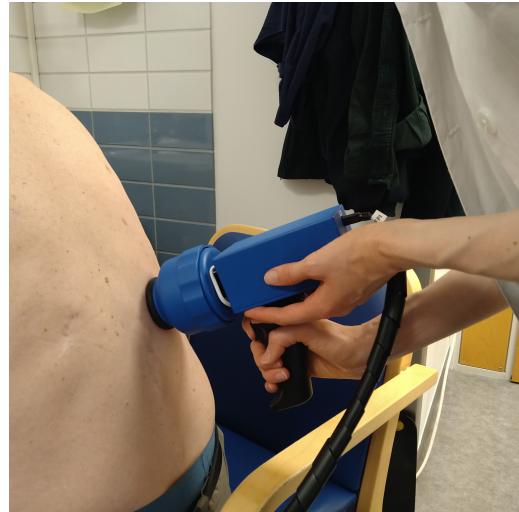
2.4.2 Instrumentation

The physical imaging system (below, Figure 35) had a two-tiered trolley with a lockable box, a compact computer CPU, a monitor, a mouse, and a keyboard. Two imager-holding brackets were 3D printed and mounted to the trolley. One served as a light shield for dark reference imaging and the other with white Teflon on the bottom served the white reference imaging. Besides reference imaging, the imager could be stored at the brackets. The devices were securely mounted to the

trolley, and the user guide was placed in the box. The SICSURFIS instrumentation, the user interface, trolley and the imager, are shown in Figure 35a, and the physical size of the imager can be seen in Figure 35b.



(a) SICSURFIS imaging system



(b) SICSURFIS imager in clinical use

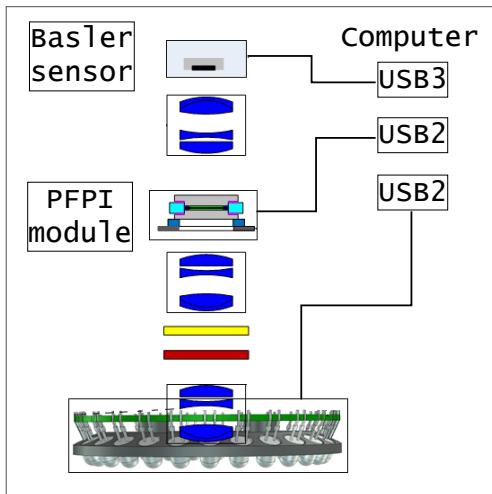
FIGURE 35 SICSURFIS instrumentation. The two-tiered trolley and the imager in use. The imaging system was mounted into a portable trolley (Subfigure a). The SICSURFIS HSI is a small and light-weight spectral imager, which is designed to reach complex skin surfaces and produce 3D skin surface models from the captured lesion HS images (Subfigure b).

The CubeView software, first introduced in Article PVI, was selected as the basis for developing the user interface.

2.4.3 Imaging processes and HSI's device modules

The imaging system and the imaging setup were designed target-wisely. As an example, the stray-light protection cones (below, Figure 36b) were designed to adapt easily into complex skin surfaces, and the captured wavelengths were selected to measure the target features, such as haemoglobin. The optics and illumination were designed for the needs of a photometric stereo algorithm, which were used to compute the 3D skin surface models. The unwanted transmission peaks for the FPI were cut with using accurate low and high band-pass filters.

In order to create the capturing workflow, three devices were controlled: the sensor (Basler acA series colour sensor), piezo actuated FPI and custom-made LED module. The LEDs were tilted at a 30-degree angle towards the optical axis. The LED module was placed around the front optics, inside the stray light protection cones, as shown below in Figure 36b.

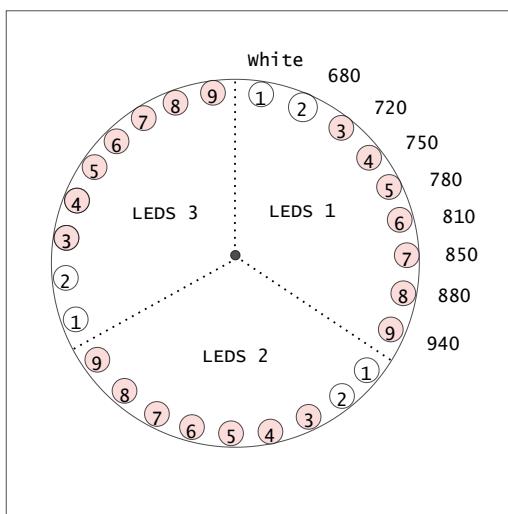


(a) SICSURFIS imager components. Yellow and red blocks are low and high band-pass filters, blue modules lenses.



(b) When a stray light protection cone is removed (on the left), the LED module surrounding the optics can be seen.

FIGURE 36 SICSURFIS imager modules are illustrated in Figure (a). The imager was connected to the computer using one USB3 and two USB2 channels. Each of the modules required individual programmatic control. Figure (b) show the imager's stray light protection cones and LED module, which was mounted around the front optics, inside the protection cones.



(a) **LED module** of the SICSURFIS imager had three sets of LEDs, which were tilted at a 30-degree angle towards the optical axis. The wavelengths of the LEDs are shown in the figure. One 3D image required three HS images, each captured with one set of LED lights, which changes the angle of light.

Imaging setup and parameters of one HS cube			
VIS		VNIR	
Spectral channels	LEDS white 680	Spectral channels	LEDS 720
477		700	750
500		725	780
524	CAPTURED with LEDS 1	735	810
582	LEDS 2	750	850
550	LEDS 3	760	880
575		765	940
578		775	
582		783	
600		790	
626		801	
630		812	
639		825	
651		834	
669		815	
677		878	
681		891	
686			

(b) **SICSURFIS HS image parameters.** One HS image for the 3D skin surface models required capturing six (6) HS cubes; VIS and VNIR channels were captured separately with two custom LED and FPI settings, and both were captured using three (3) different angles of light (led sets, Subfigure a).

FIGURE 37 SICSURFIS imager's LED module and imaging setup parameters. The pink colour in Figures (a) and (b) visualises the relation of LEDs and captured wavelengths.

The selected LEDs, and the imaging setup are visualised and explained in Figure 37. The figure collage show the relations between the selected LED wavelengths and the captured wavelength channels. The LEDs with white and 680 nm were set on while capturing the VIS channels, while the LEDs from 720 to 940 nm were on during the VNIR imaging.

2.4.4 Device workflow and user interface

The device workflow was designed to be as simple as possible, enabling the user to concentrate on the patient work. The following examples and visualisations show step-by-step user actions, visible and hidden software actions and hidden device level actions. The Figures from 38 to 46 can be found at the end of this section.

Opening the software

When the user interface is started for the first time the program will ask for the device settings (paths to GeniCam GenTL producer file, FPI gap calibration file, band-pass filter calibration files and LED settings files for preview). Paths of the given settings are saved to the Windows registry for future reference. The software has a complicated imaging setup. Thus the device configuration files have predefined names that ensure the correct LED settings are used with the correct FPI bands.

The starting procedure in clinical use is described in Figure 38. The device objects are initialised and passed to the "hsi" object, which controls the FPI positions and capturing procedure using sensor object (cam), FPI object ("fpi"), and band-pass calibration files. In the error status, the system informs the user to check device connections, add missing calibration files, free up the storage space and restart the computer. If the status is normal, the system informs that the imager is ready to work, guides the user on how close is the time to free up the storage space and gives the instructions for the next steps. The image capturing buttons are disabled, and the patient tool is opened. The system is ready to work.

Filling the patient information

Figure 39 visualises the patient tool's workflow. After selecting or filling the patient and lesion IDs, the "Confirm" button is clicked, the patient tool is moved to the background, and the LiveView tool is opened. Before opening, the sensor pixel mode is changed to RGB, the FPI position is checked, and all LEDs are set on. While the patient tool is open, the system knows which patient ID and lesion ID are under work, and the HS image will later be saved correctly. If the patient or lesion ID changes during the session, the image saving settings are updated accordingly. The LiveView tool and related user information guide the user to check the focus and instruct the user to place the imager into the dark reference bracket and click the capture dark -button.

Capturing

Capturing references and spectral data. When the user feed "Capture dark" is activated (see Figure 40), the system informs the user to wait and uses sound to inform until it is ready. After the sound, the user will see the instructions for capturing white reference since the white reference button is enabled. Behind the curtains, the video is paused, sensor pixel mode, exposure time and other settings are updated, and LED lights are off. The "hsi" object captures 40 frames and returns the mean for the controller, which saves it. The "hsi" object typically saves the dark reference itself, but our imaging setup changes the FPI band-pass filter settings several times. Each update clears up the "hsi" objects' memory, so an external solution was needed for our setup. The dark reference is captured with a similar exposure time to the frames in the white reference and spectral raw data.

Input from "Capture White" (Figures 41 and 42) and "Capture" (Figures 43 and 44) follows similar device controlling processes. For each three LED settings, the according VIS band-pass calibration file controls the recorded FPI positions, providing the spectral channels. The captured data is temporally stored before the band-pass filter calibration file is changed. When the white reference is recorded, the user interface guides to capture of spectral data.

The difference between capturing the white reference and spectral data is user related. Before capturing, the user is instructed to focus the imager. The LiveView tool with LEDs on helps the user to place the imager optimally against the patient's skin before recording the spectral data by clicking "capture" button.

Preview and continue

After recording the spectral data, the software computes a preview, opens a Pre-View tool and visualises the captured data. The user can select whether to save or dismiss the data with one click. The hidden processes are visualised in Figures 45 and 46. The system automatically saves the captured data forming the file name from patient and lesion IDs and date-time information.

The related visualisations (Figures from 38 to 46) are presented below in a workflow order. The actions are colour coded. The purple and grey boxes show the processes that are hidden from the users, the yellow box represents the messages that the interface shares with the user. The light blue box represents the user's actions, which are inputs that start the hidden processes underneath the visible user interface. The figures reveal the software design decisions and device-controlling workflows in detail.

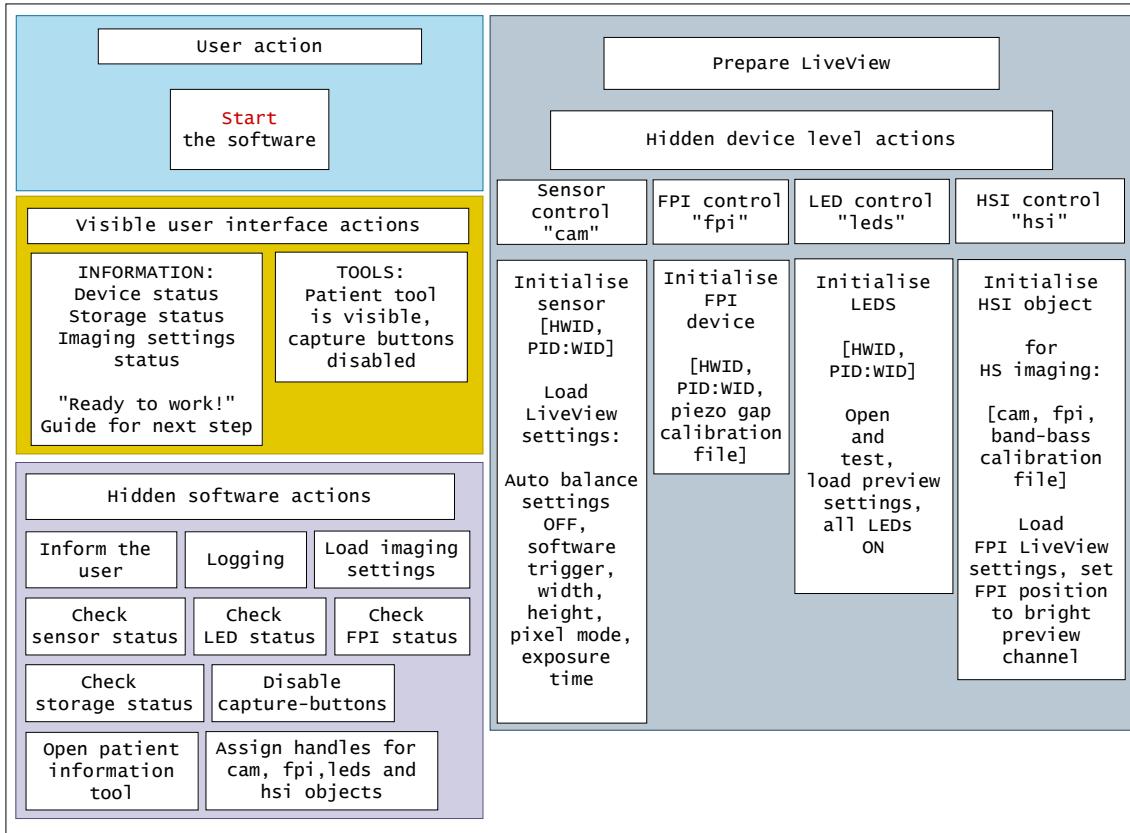


FIGURE 38 Opening the software. When the user opens the software, the software and imaging controller start acting; the devices are initialised, settings and available storage are checked, and logging is started. The imager is prepared for the next step, and the user has detailed information about the status check. After the information, the software guides the user toward the next step. The purple and grey steps are hidden from the user, the yellow box represents the communication with the user, and the light blue box represents the user's actions.

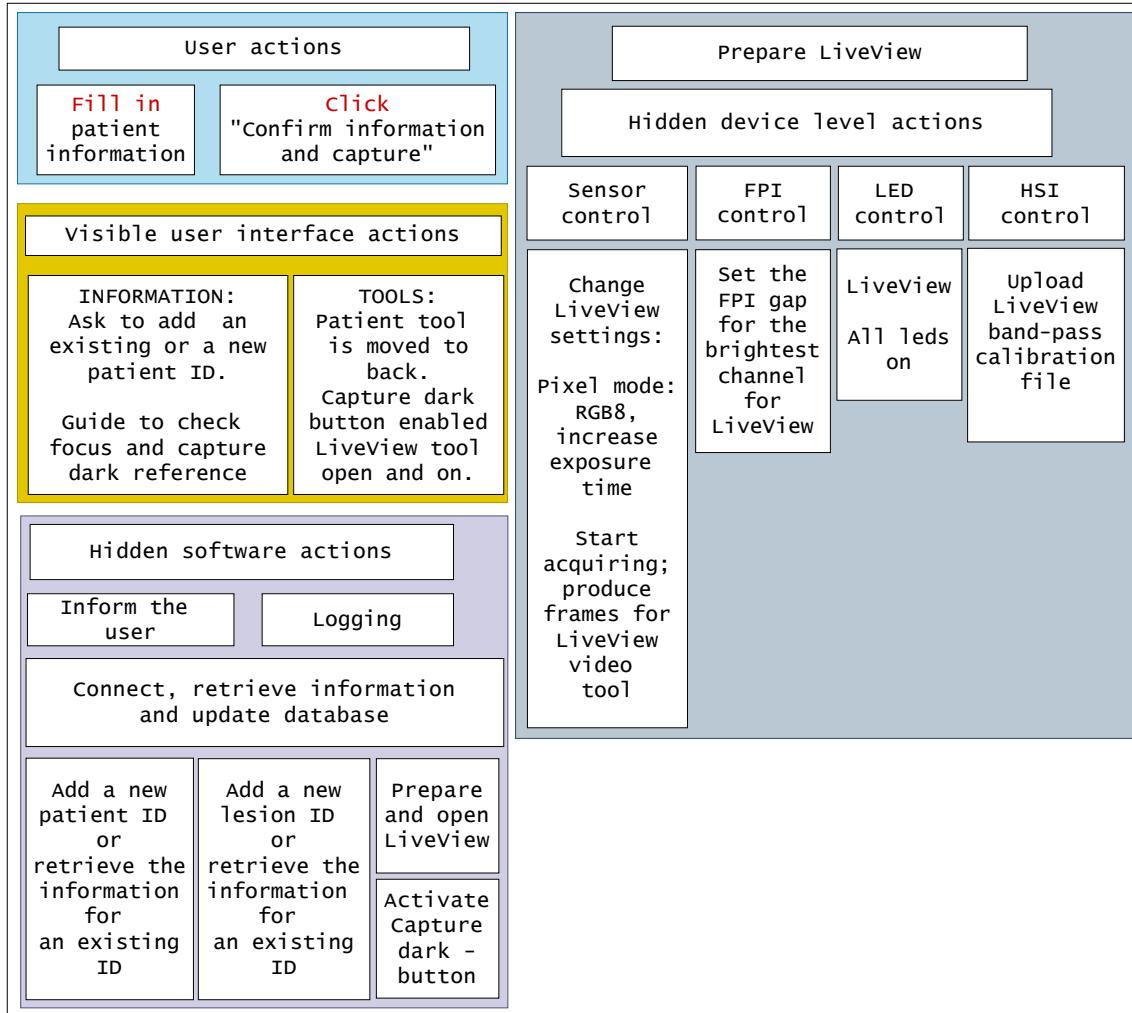


FIGURE 39 **The patient information** is filled to the patient tool. Each patient has an individual ID, which is selected from a list. A patient can have multiple lesions; each lesion has a lesion ID (a number from 1 to 50). The UI has a small database, which stores patient and lesion IDs, retrieving and updating the database through the patient tool's information. After the form is filled, the controller opens a LiveView tool, enabling the capture dark button and guiding the user to proceed. The IDs or the database do not contain any sensitive patient information; the system is pseudonymous.

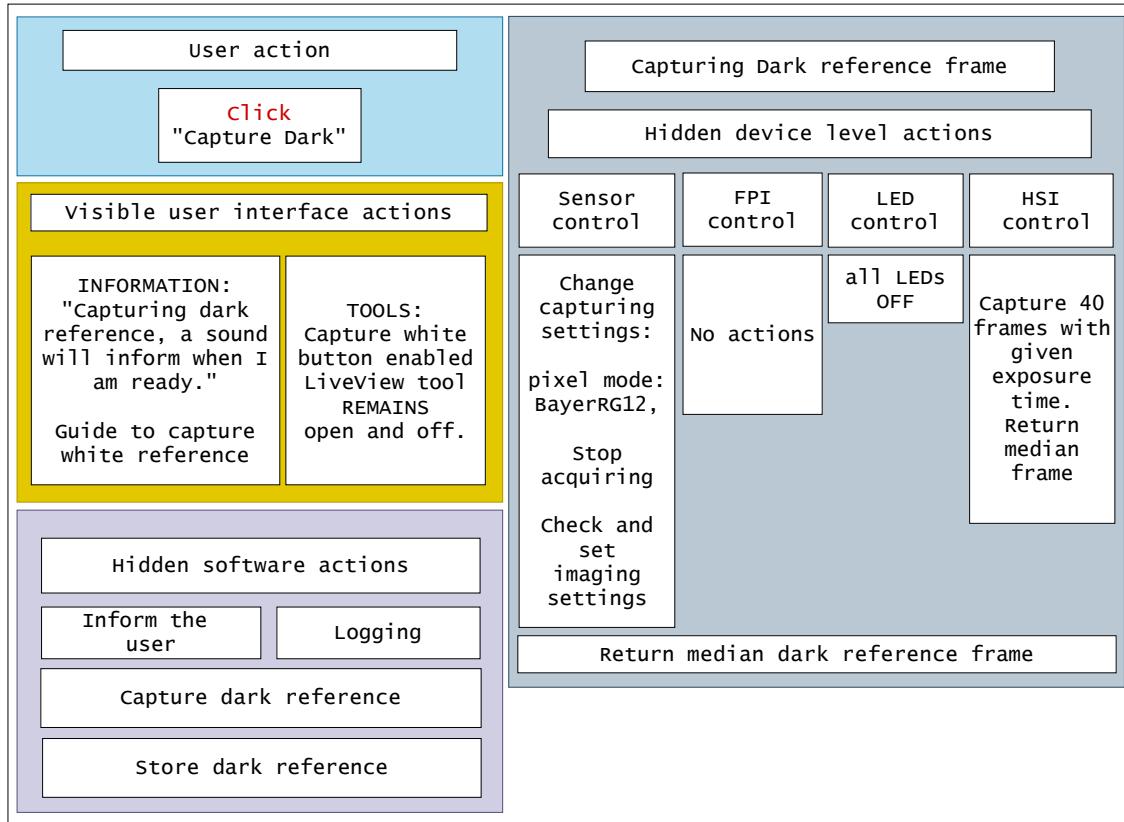


FIGURE 40 Capturing the dark reference. While capturing the dark reference, the user interface informs the user to wait. After the data is recorded, the interface informs the user with a sound and guides the user to capture the white reference. The LiveView tool is open, but the video is paused. The figure shows the user actions, visible UI actions and main hidden software and device level actions.

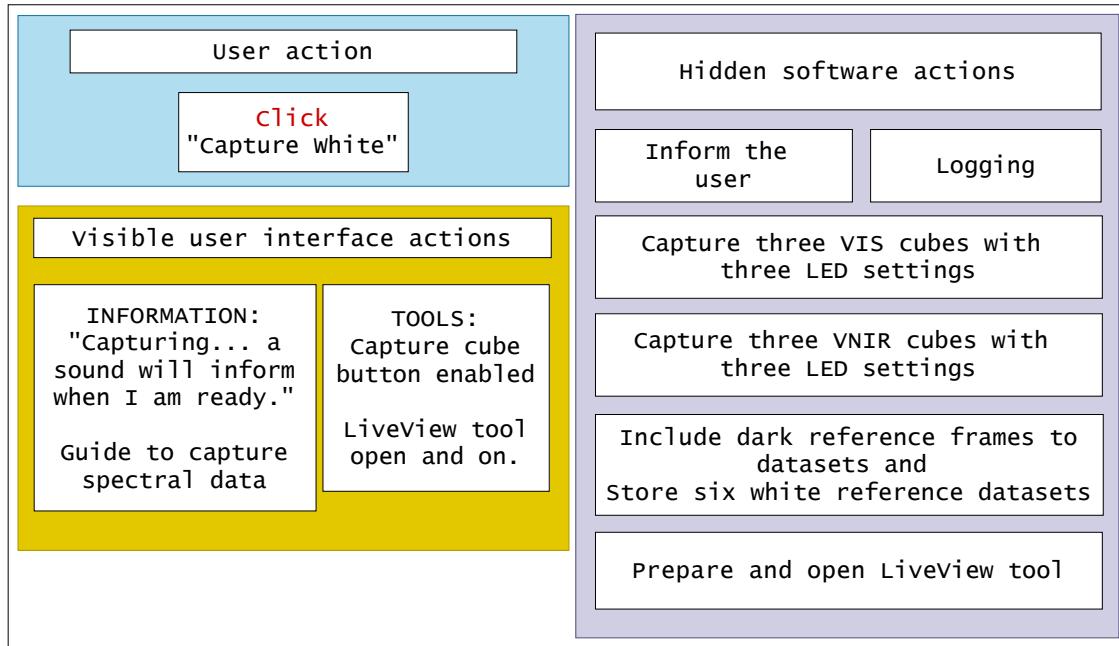


FIGURE 41 **Capturing the white reference.** User actions, visible interface actions and hidden software actions.

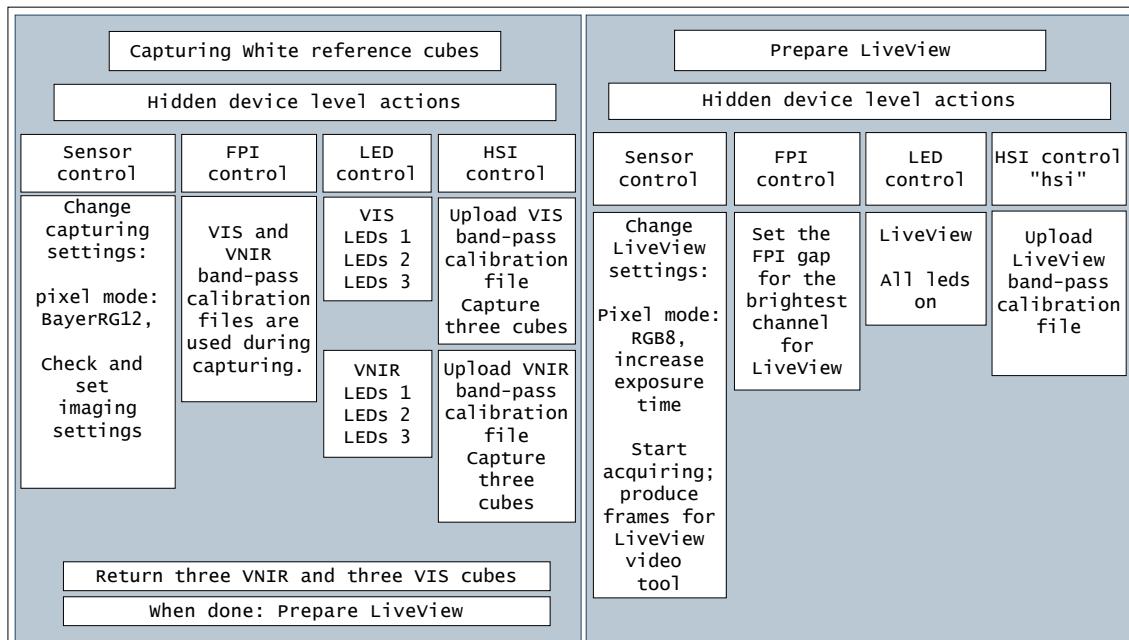


FIGURE 42 **White reference's hidden device-level actions.** The imager is placed into a white reference bracket. While capturing the white reference, the user interface informs the user to wait. After recording the data, the interface informs the user with a sound. The user is guided on how to capture spectral data. The LiveView tool opens, and the video starts.

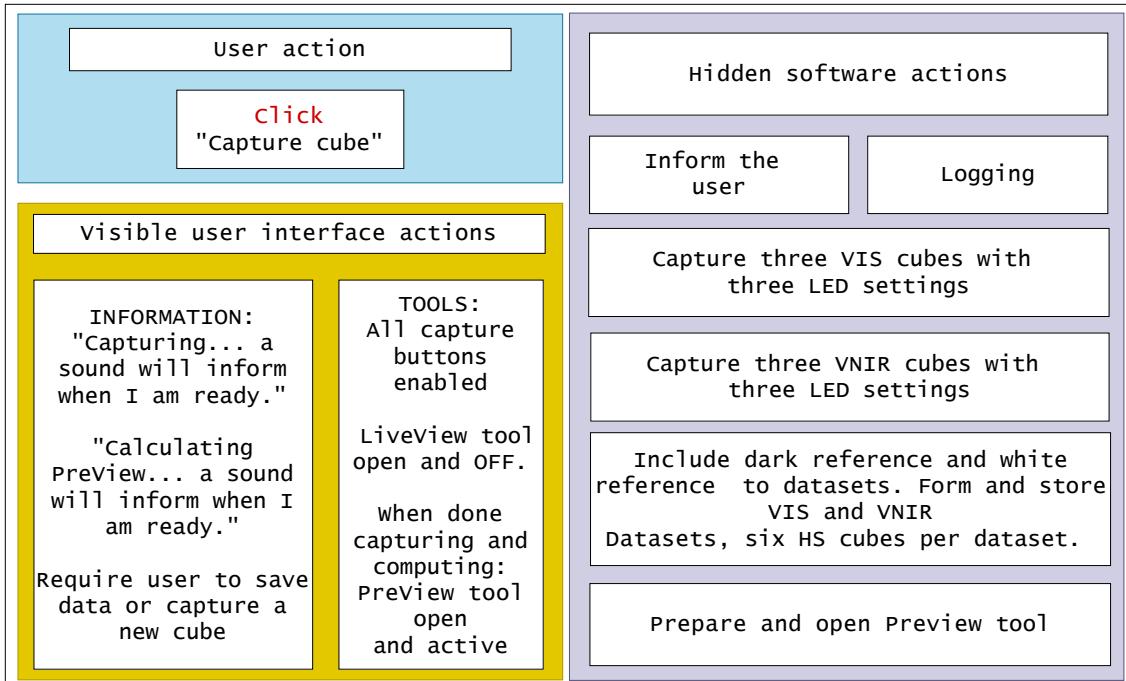


FIGURE 43 **Capturing the spectral data.** User actions, visible interface actions and hidden software actions.

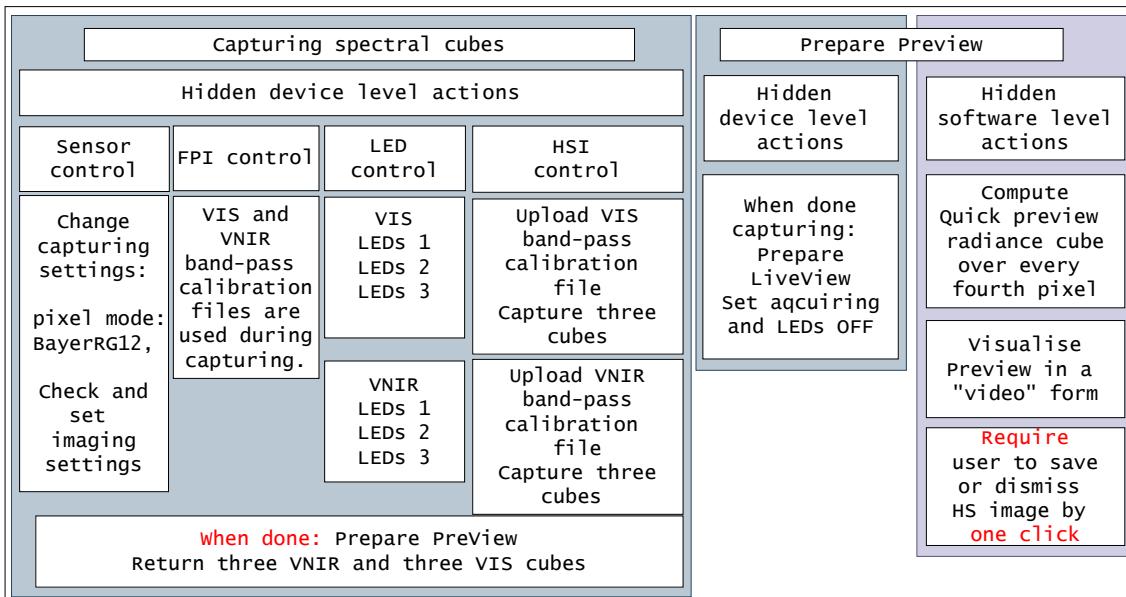


FIGURE 44 **Hidden device level actions** of capturing the spectral data. The user interface informs the user to wait. After recording the data, the interface informs the user with a sound. The user is guided to release the imager into its bracket. The system will compute a quick preview image, visualised using a PreView tool.

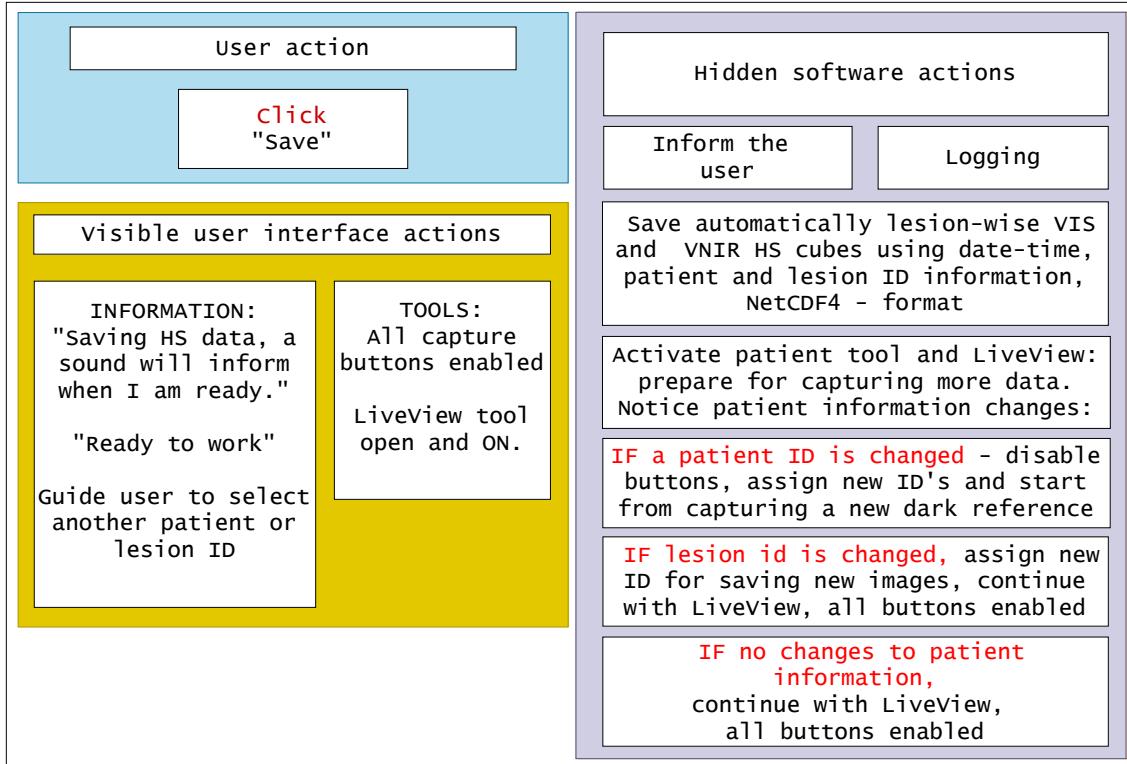


FIGURE 45 **User action "save"** activates automatically saving functions, and the captured VIS and VNIR cubes are saved. The user is informed to wait and how to continue. All capture buttons are enabled, and the LiveView tool's video is active.

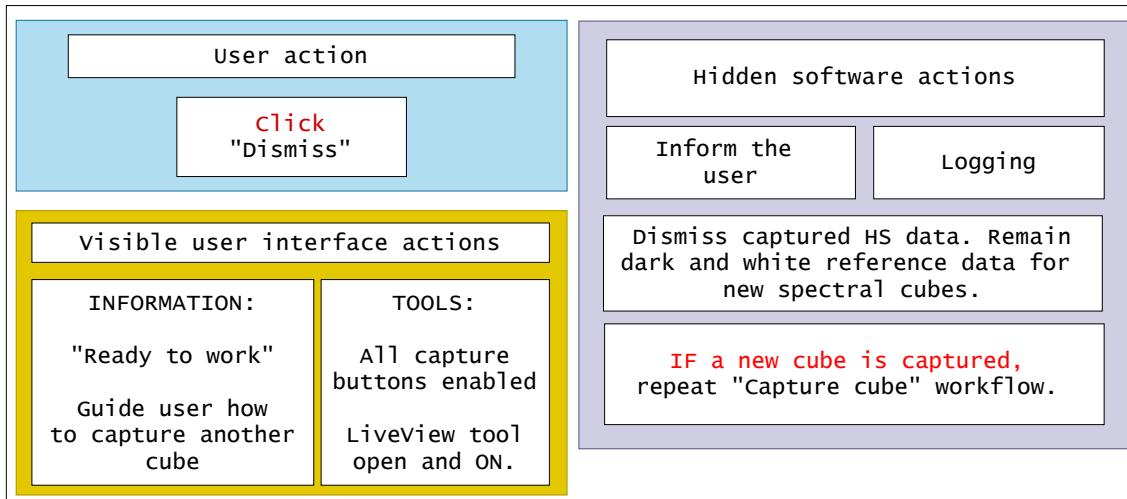


FIGURE 46 **User action; dismiss** the captured data is deleted, and the interface guides users to capture another spectral image.

2.4.5 Summary: machine vision fundamentals

Sections from 2.2 to 2.4 cover computer vision image processing and machine learning phases, approaching HS imaging systems. The main outcome that ought to be memorised is summarised as follows:

- Data provided by different types of spectral imagers (Section 2.1.7.1, Figure 18) requires different data processing methods. As an example, the VTT imagers (Section 2.1.7.2) of the related studies are frame-based with colour sensors, requiring bilinear interpolation and weighting with calibration-based coefficients.
- Different imaging setups (Section 2.1.6) and illumination affect the kind of spectral data that can be processed from the raw data. The examples of computing reflectance, transmittance and absorbance HS cubes with examples can be found in Sections 2.2.2 and 2.2.3.
- The data type is determined by the nature of the application; in HS imaging, some phenomena benefit from the transmission, and another scene is satisfied with hemispherical reflectance data.
- After pre-processing, the data can be used as input for machine learning methods to solve and study various problems and applications. Some methods benefit from pre-processing and data transformation, such as dimensionality reduction, but some methods are not expecting those.
- The devices, workflow and imaging setup can be joined together to a user interface, which simplifies the complicated processes to a few user inputs, providing high-quality raw data, and some first-level analysis tools for evaluating the phenomena. As an example, CubeView (Article PVI) offers simplified experiences for first-time users with the possibility to configure all device parameters, including data-analysis tools. CubeView Hospital (Articles PIV and PV) is an adjustable software which can be set to provide minimal effort imaging experiences, requiring no understanding of complicated parameter setup or analysis from the users. The basic workflow with software development-level visualisations can be found in Section 2.4.4.

3 EXPERIMENTS AND RESULTS

The point of view of this chapter arises from the hypothesis, aims, and research questions, which join the conducted research and results. The aim is to bridge the theory and conducted research and highlight the main findings of the articles with a new discussion.

TABLE 6 Research questions, related articles and sections with topics

Questions	Related articles	Section	Topics
1, 1a, 1b	PI	3.2.1	Machine learning; distance-based classification
1, 1c	PII	3.2.2	Machine learning; distance-based anomaly detection
1, 1d	PIII	3.2.3	Machine learning; self-learning classifier
2, 2a	PIV	3.3.1	HS imaging system and application analysis
2,2a	PV	3.3.1	HS imaging application analysis
2, 2b	PVI	3.3.2	HS imaging and analysis software introduction

Above, Table 6 enlightens how the related articles are connected to the research questions (presented in Section 1.5) and subsections of this chapter. The following section explains the review processes, and after that, we will have an introduction to the conducted research. Articles PI; PII; PIII and PVI is introduced with a summary of the main content, methods and a short discussion. Articles PIV; PV are introduced at once since the machine vision system remains relatively constant in these continuation pilot studies.

3.1 Review processes

Table 7 (below) shows the details of the review processes with impact scores and factors. Four of the articles were accepted as they were; article PV received compliments from both reviewers and was published in a journal with a 4.242 impact factor. Articles PI, PIII and PVI were accepted as they were, in peer-reviewed conferences. Details of the improvements brought about by the revisions are in-

cluded in the article-wise discussions.

TABLE 7 Review process details

Article	Reviewers	Minor revisions	Major revisions	Editorial revisions	Presentation	Poster
PI	2	-	-	-	Recorded	No
Description: At first, an extended technical abstract was submitted. The conference reviewed and approved the topic and the theoretical basis of the research. The article was submitted within a given timeline. The full article was inspected and approved without revisions by the conference chair. The author was selected to perform an oral presentation. The review process was minimal, impact score 0.39.						
PII	3	1	-	-	Recorded	No
Description: The manuscript was submitted to ISPRS conference. The ISPRS organisation publishes ISPRS Annals and ISPRS Archives, where the Annals is more demanding than achieves. The peer review is a double-blinded process. The evaluation criteria is innovation (10%), scientific formulation (10%), experiments and validation (10%), relevance (10%), quality of presentation (10%) and overall recommendation (50%). Each category is scored, and scores are percentage weighted; the maximum is 100 points. Based on the first-round reviews with 82 points, the article was accepted with minor revision for the Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences. The Annals act as the proceedings of the 2021 edition of the XXIVth ISPRS Congress. The manuscript was approved after the first revision. The author was selected to give an oral presentation at the conference. The review process was double-blind, impact score 1.34.						
PIII	3	-	-	-	Oral and recorded	Yes
Description: The manuscript was submitted to ISPRS conference. The evaluation criteria were similar to article PII. The overall score was 81. The three reviews were uniformly positive. After proofreading, the manuscript was accepted to be published as it was in the International Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences. The Annals act as proceedings of the 2022 edition of the XXIVth ISPRS Congress. The author was awarded with recorded and live oral presentations and poster session at the conference. The review process was double-blind, impact score 1.34.						
PIV	2	1	1	3	No	No
Description: Multidisciplinary Digital Publishing Institute (MDPI) is an open access publisher for scientific journals. We selected and submitted the manuscript to a Sensors special issue, Image Analysis and Biomedical Sensors. The impact factor of MDPI Sensors is 3.576. The peer-review process was half-blind, including two peer reviews and an academic editorial board. The manuscript had major and minor revisions from the peer reviewers and three editorial revisions before it was accepted for publishing.						
PV	2	-	-	-	No	No
Description: Article was submitted to MDPI's Journal of Clinical Medicine with a 4.242 impact factor. The MDPI peer-review process is half-blind and similar to article PIV. The article was carefully written with a detailed methodological description and deep medical discussion. The language was checked before the first submission. The authorship was equal between the dermatologist PhD student Vivian Lindholm and the author. The article was accepted as it was, with a positive reviewer feedback.						
PVI	2	-	-	-	No	No
Description: The article was published with a similar peer-review process than article PI and accepted by the chair without revisions. The author was not participating in the conference. However, she wrote the software introduction chapter and the software she designed and implemented was introduced at the Photonics West 2019 Finland Pavillion by the first author Roberts Trops. The review process was minimal, impact score 0.39						

3.2 Developing methods for computer vision analysis

The relationships between AI, machine learning, deep learning, computer vision, image capturing and machine vision were visualised at the beginning of Chapter 1 (Figure 3). Machine learning can be seen as an essence of computer vision, giving us, for example, the interest to study and develop methods and approaches for two common tasks; classification and anomaly detection (Section 2.3.3).

The aim of this section is to introduce three articles which study variations from a distance-based machine learning method. The application areas are in remote sensing, and the presented methods have the potential for real-time applications.

3.2.1 Effective classification method for hyperspectral imaging

Minimal Learning Machine in Hyperspectral Imaging Classification (Article PI) answers to research Questions 1a "How a distance-based machine learning model performs in hyperspectral imaging classification?" and 1b "Can we reduce the method's computational complexity by intentionally selecting the reference training points and maintaining a high accuracy?"

Summary

In this study, we introduce the *Minimal Learning Machine* (de Souza et al. 2015) (MLM) as a computational effective training and classification method for the HS imaging classification. MLM is a distance-based machine learning method that utilises mapping between input and output distances. Input distances are the distances between the training set and its subset R . The output distances are corresponding distances between the label values of the training set (Y) and its subset (T). We are using Nearest Neighbours MLM, which is a generalisation of the nearest neighbour classifier (de Souza et al. 2015). The classification is performed by sorting the predicted label distances of the new data points and assigning the closest label values from the ground truth labels. The method is easy to implement and computationally less complex than many other reference classifiers, often described as black boxes.

As the main outcome, we tested MLM against known classifiers and propose a new PC- MLM framework, which focuses on selecting the training output samples based on the underlying geometry of the data. The framework consists of a new selection algorithm (Algorithm 4) and previously presented MLM training and classification Algorithms 2 and 3 (Section 2.3.8). The new data point selection algorithm (Algorithm 4) utilises principal component analysis (PCA, Section 2.3.2) and selects only 3 data points from each of the classes in the direction of each selected principal component. As a result, we have a collection of data points representing the geometry of each class. This approach differs from the original MLM approach, which selects the reference training points randomly (de Souza et al. 2015). The R size (number of reference points) correlates with

accuracy, and it is the most influential factor in the computational complexity (de Souza et al. 2015).

The framework's objectives are to minimise the size of R , increase the accuracy, reduce the training and classification time of a basic MLM model and reduce the computational complexity. The results of the framework are validated against an MLM model with a randomly selected subset and four three other known classification methods; K nearest neighbours (kNN), Random forest (RF), dense artificial neural network (ANN) and Support vector machine (SVM) with three known HS data sets (Indian Pines, Salinas and Pavia).

Methods

The pixel-wise MLM classifier for HS data is defined as follows. The output and input distances would be $d(\mathbf{x}_i, \mathbf{m}_k)$ and $\delta(\mathbf{y}_i, \mathbf{t}_k)$, where $\mathbf{x}_i \in X \subset \mathbb{R}^D$ are training set of spectra with D wavebands and $\mathbf{m}_k \in R$ are randomly sampled subset of X and correspondingly $\mathbf{y}_i \in Y \subset \mathbb{R}$ are labels of the training set and $\mathbf{t}_k \in T$ are a subset of Y . Training set X consists of N samples, and subset R has K samples. Now, we define two matrices based on these distances $\Delta_y \in \mathbb{R}^{N \times K}$ and $\mathbf{D}_x \in \mathbb{R}^{N \times K}$. By assuming the linear mapping (visualised in Section 2.3.8, Figures 33 and 34) between these two distance matrices, we have a linear model

$$\Delta_y = \mathbf{D}_x \mathbf{B} + \mathbf{E}, \quad (12)$$

where \mathbf{B} is coefficients and \mathbf{E} , is residual. Coefficients \mathbf{B} can be approximated using the least squares estimator

$$\hat{\mathbf{B}} = (\mathbf{D}_x^T \mathbf{D}_x)^{-1} \mathbf{D}_x^T \Delta_y. \quad (13)$$

Now $\hat{\mathbf{B}}$ is linear model between distances $\delta(\mathbf{y}_i, \mathbf{t}_k)$ and $d(\mathbf{x}_i, \mathbf{m}_k)$. Distances between new spectrum \mathbf{x}_n and its label y_n is

$$\delta(y_n, T) = d(\mathbf{x}_n, R) \hat{\mathbf{B}}. \quad (14)$$

Label y_n can be estimated by solving optimisation problem

$$\min_{y_n} \sum_{k=1}^K \left((y_n - \mathbf{t}_k)^T (y_n - \mathbf{t}_k) - \delta^2(y_n, T) \right)^2. \quad (15)$$

If we try to solve Equation (15) with optimization methods, our classification is time-consuming and computationally expensive. Model $\hat{\mathbf{B}}$ is a kind of generalization of the nearest neighbour method for the data set X : Equation (14) gives us distances to the nearest label values of \mathbf{x}_n . For classification of \mathbf{x}_n , we need to perform argument sorting for $\delta(y_n, T)$ and assign the closest label value from T . By selecting k closest values and using majority voting of labels, we have similar results to kNN.

Algorithm 4: The PC-MLM algorithm

Step 1: PCA Run PCA on the whole training data X
Step 2: Evaluate and select Select the principal components for the PC
Train the model
Step 3 : For each PC in selected principal components:
for i in range(the number of classes in the ground truth)
if *ground truth class == i* **then**
 length = length(ground truth)
 perform an argument sort along axis 0
 Find the 'minimum point'
 minimum_position = sorted_component[length x 0.95]
 Find the 'maximum point'
 maximum_position = sorted_component[length x 0.05]
 Find the 'median point'
 median = sorted_component[length / 2]
 add selected 'minimum, maximum and median points to the R
 R.append(
 training_data[minimum_position,:])
 add the ground truth data points to the R's ground truth array T
end

Result: The R and T are now sorted arrays that represent the data sets shape in the direction of the selected principal components. The R and T are ready for the MLM classifier implementation with algorithms 2 and 3.

Above, Algorithm 4 introduces the pseudo-code steps of the PC-MLM implementation. The article, its summary and discussion related to computer vision's phases of preparing, optimising and classifying the data for the analysis (Section 4.3: Figure 65, Sections 2.2.1, 2.3.1 and 2.3.8).

Results

The study confirms that the MLM performs well against popular reference methods in HS imaging pixel-wise classification. We show how the performance is affected by the size of the subset R and how the idea of intentionally selected training points enables high accuracy with the reduced R size, especially with large data sets. Results confirm that MLM and PC-MLM are the fastest machine learning models to train when the training set size is large. Overall, the MLM reduces the complexity of the analysis and provides computational benefits against reference models. The proposed PC-MLM framework can improve performance against an MLM with a randomly selected subset R.

An example of the Pavia dataset's obtained class-wise geometry is visualised below in Figure 47. Due to the noisiness of the data, the minimum and maximum values were brought 5% towards the median position. The change of 5% was optimised, and the trial was performed repeatedly with several random values from 1 to 25.

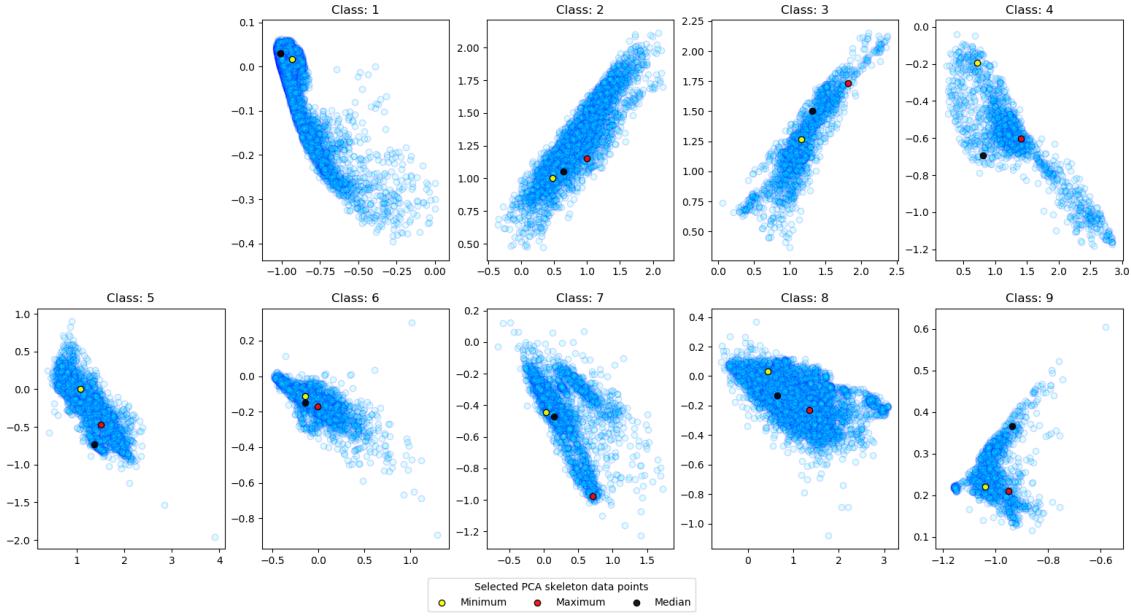


FIGURE 47 Pavia dataset's geometry is represented class-by-class. The selected data points (minimum, maximum and median) are marked on the image. The PCA reveals the geometry of the data; the first PC component has the highest variation of the data, the second component reveals the highest variance in the orthogonal direction of the first component and so on. The Pavia dataset is visualised by the first and second components.

Discussion

In some real-time applications, the data intensity causes performance challenges. Those issues relate to the platforms (e.g. drones), payload restrictions, the issues of the available energy and the complexity of the machine learning models, which decreases the explainability of those models (Ghamisi et al. 2017). With deep neural networks, it is challenging to understand why the model has made a certain decision. Thus, exploring how we can decrease the complexity is an interesting idea.

According to de Souza et al. (2015), the MLM has only one obligated parameter to optimise, the R size. The proposed PC-MLM framework had three: the distance metric, R size (related to a number of principal components) and a number of neighbours. Both MLM approaches' optimisation needs are low compared to the reference models' several hyperparameters. Since the MLM is distance-based, the distance metric might be useful to optimise with all MLM approaches, leaving the number of neighbours as the only added parameter.

The answer to Question 1a: "*How a distance-based machine learning model performs in hyperspectral imaging classification?*" is encouraging. The MLM is easy to implement and performs well in pixel-wise HS image classification. The MLM is a fast method with comparable accuracy. So far, of its minimal and simple nature, it could be a new candidate for state-of-the-art methods, especially when the computation is performed close to the data collection.

The second Question 1b: "*Can we reduce the method's computational complexity by selecting the reference training points intentionally and maintain a high accu-*

racy?" receives answers yes and no. With the effort of one additional parameter optimisation and implementation of the PC selection algorithm (Algorithm 4), we can reduce the R size and decrease the computational complexity. These actions decrease the number of distance calculations related to distance matrices and new data points, decreasing the number of prediction calculations (Equation (14)). However, it seems that MLM is quite an effective method, and only depending on the situation will the proposed framework improve the models' performance. Due to the PC algorithm, the basic MLM with random selection is faster to train, and so far, our recommendation is to select the method depending on the case. For example, when classifying a large data set, such as Pavia, the proposed approach is recommended over the basic MLM, but with a smaller data set containing a varying number of samples per class, it might not be necessary to explore the geometry.

The pixel-wise classification typically uses only one HS cube as a source for training, validation and testing, being a common approach in HS image analysis and classification studies (Ahmad et al. 2022). When the training and test pixel spectra are collected from the same HS images, the results might be biased to reach a higher accuracy than results obtained with similar models and clean test data. Because of that, it should be noted that with high-accuracy results (such as in this study with all classifiers, especially with Pavia and Salinas data sets), the results might include a data or model-driven bias. However, the size of the utilised sub-samples seems to affect the image classification bias levels - the smaller the sub-samples from the images are, the lower the possible bias effect in the results (Model and Shamir 2015). Since the approach is pixel-wise and one data point (sub-sample) is *one* \times *one* pixel spectrum, the bias level of our approach is naturally lower than, for example, in classic image classification with sub-image $n \times n$ sub-samples.

Compared to the reference methods SVM, ANN and RF, which were trained and tested with randomly selected pixel spectra (60:40 portions), the level of possible data-driven bias are probably similar to MLM's and PC-MLM's results. With black box style reference models, explainability is challenging. Any feature, a real target feature or, for example, a sensor-based artefact in an image, can affect the decisions, which might be detected by the models causing the bias (Model and Shamir 2015). As with the MLM, the distance-based method is not utilising the features but the distances; thus, it can be a benefit against the reference models with hidden features and hard-to-explain decisions. Distance-based methods with a small reference set might also be more robust with respect to overfit than the compared SVM, ANN and RF. Those reference methods need a relatively high amount of training data. The models might be prone to overfit, which causes model-based bias in the results.

The article was accepted in a two-phase review process as it was (Table 7, Section 3.1).

3.2.2 Minimalistic approach to hyperspectral image anomaly detection

Piecewise Anomaly Detection Using Minimal Learning Machine for Hyperspectral Imaging (Article PII) answers to research Question 1c "*How a distance-based machine learning model for classification can be adapted to local and global anomaly detection and how to improve the resilience with respect to noise?*"

Summary

HS applications offer promising tools for remote sensing and Earth observation. Recent development has increased the availability and quality of the sensors. Anomaly detection is a typical remote sensing application example, which can be performed as a real-time application. A real-time system has limitations. For example, due to an HS data intensity, the platform's constraints on payload, available energy and processing capability. As mentioned, the computational complexity of machine learning models might challenge those platforms; thus, choosing a computationally efficient method is wise.

This article continues the examination of the distance-based machine learning method MLM. The scope is in anomaly detection, and the article is an independent continuation study to our previous article "*Minimal learning machine in anomaly detection from hyperspectral images*", (Pölönen et al. 2020). The first study showed how MLM could be used generally in anomaly detection. The examination was done with remote sensing, and the laboratory collected HS data. The MLMs' main benefits were the low false alarm rate, computational efficiency, and easy implementation using distance functions and standard linear algebra. The MLM could effectively capture global anomalies with a low false alarm rate, especially with large datasets.

This study extends the previous MLM anomaly detection method to cover global and local anomalies and increases its robustness with respect to noise. We will show that it is possible to detect local and global anomalies by using the variance of the predicted new data point label distances (δ) with carefully selected two thresholds. The downside of the thresholds is the high sensitivity with respect to the noise. Thus, we introduce a novel approach, the piecewise MLM. As a result, the piecewise MLM can detect global and local anomalies, being significantly more robust with respect to noise than the original MLM anomaly detector introduced in (Pölönen et al. 2020).

Methods

Similarly, with nearest neighbours MLM classifier, our anomaly detector MLM does not estimate the \mathbf{y}_n , which is the computationally most expensive part of the method. Instead of estimation, we assume that if \mathbf{x}_n is inside the training set, the \mathbf{y}_n is nearby the labels in subset T . Thus, the distribution of estimated distances $\delta(\mathbf{y}_n, T)$ should be relatively similar to training phase distances in Δ_y . If \mathbf{x}_n is an anomaly or outlier, it should be detected already in $\delta(\mathbf{y}_n, T)$. It is enough to study

the behaviour of $\delta(\mathbf{y}_n, T)$. In this study, we use the variance

$$\text{Var}(\delta(\mathbf{y}_n, T)) = \frac{1}{K} \sum_{i=1}^K (\delta_i - \bar{\delta})^2 \quad (16)$$

to detect anomalies, where K is number of elements in a vector δ and $\bar{\delta}$ is average value of the δ . For the variance, we optimise two threshold values that reveal the anomalous spectra from the dataset. The lower threshold is the key to detecting local anomalies and the higher threshold exists for global anomalies. The threshold optimisation for the anomaly detection MLM can be performed with a parameter search function that loops the anomaly detection algorithm (below, Algorithm 6) with different combinations of lower and higher threshold values and compares the received accuracies.

The piecewise MLM approach re-trains class-by-class the linear model $\hat{\mathbf{B}}_i$. While the training set X and its subset R remain unchanged, its labels Y and its subset T are updated with *one against all* style, which is visualised below in Figure 48. As a result, we have a class-wisely re-labelled Y . Every spectrum that does not belong to the expected class is labelled anomalous regardless of whether it belongs to the anomalies or other classes.

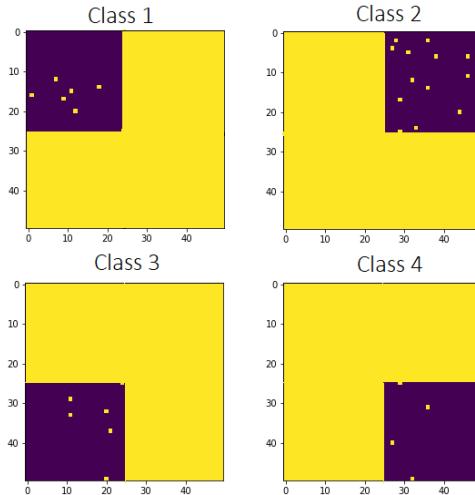


FIGURE 48 The piecewise MLM re-trains the MLM model class-by-class. The visualisation of re-labelling the Y . Only the class members we expect are labelled as zero (0). The rest of the pixel spectra, including the anomalies, are re-labelled with one (1), which means an anomaly. Zero (0): Purple, an expected spectrum, One (1): Yellow, an anomalous spectrum. The subset T is labelled similarly.

Below, the pseudo-code Algorithm 5 shows the piecewise MLM training steps, and Algorithm 6 reveals the piecewise anomaly detection phases. The idea of our algorithm is to use a piecewise trained MLM model $\hat{\mathbf{B}}_i$ for each class i . The Equation (16) is calculated for each class. These variances are summed together, and both lower and upper thresholds are used to detect anomalies.

Algorithm 5: The piecewise MLM training

Input: X, R, Y, T
Calculate distance \mathbf{D}_x
for each class i do
 Re-label class i to 0 and
 other classes to 1
 Calculate distance Δ_y
 Calculate

$$\widehat{\mathbf{B}}_i = (\mathbf{D}_x^T \mathbf{D}_x)^{-1} \mathbf{D}_x^T \Delta_y$$

end
Return: Piecewise MLM model
 $\widehat{\mathbf{B}}_{pw} = [\widehat{\mathbf{B}}_1, \widehat{\mathbf{B}}_2, \dots]$

Algorithm 6: Anomaly detection using the piecewise MLM model

Input: new data $X_{new}, R, \widehat{\mathbf{B}}_{pw}$,
upper and lower thresholds γ_l and γ_u
Calculate distance $d(X_{new}, R)$
for each $\widehat{\mathbf{B}}_i$ do
 Calculate

$$\delta(\mathbf{y}_{new}, T)_i = d(X_{new}, R) \widehat{\mathbf{B}}_i$$

 Calculate $\text{Var}(\delta(\mathbf{y}_{new}, T))_i$
end
Calculate

$$score = \sum_i \text{Var}(\mathbf{y}_{new}, T)_i$$

if $\gamma_l < score_n < \gamma_u$ **then**
 $A_n = 1$
else
 $A_n = 0$
end
Return: Anomaly detection A_n

Results

The results show that the MLM anomaly detector can detect local and global anomalies with carefully selected two thresholds. Below, Table 8 shows that the original MLM anomaly detector with one or two thresholds is faster than the piece-wise MLM approach. The computational complexity of the MLM anomaly detector is near $\mathcal{O}(NK)$, which is in relation to the number of samples in R . The piece-wise approach performs the training phases class-wisely, which increases the computation with the number of classes (C), resulting in the computational complexity to $\mathcal{O}(NKC)$.

TABLE 8 Comparison of the computation times for the ColorChecker dataset. A: MLM (cosine, variance, training set size 2500, R size 250), higher threshold: > 0.5 . B: MLM (cosine, variance, training set size 2500, R size 250), lower threshold < 0.12 , higher threshold: > 0.5 . C: Piecewise MLM (cosine, variance, training set size 2500, R size 250), lower threshold > 0.7 , higher threshold < 0.4 .

Detector	A	B	C
Training time [s]	0.084	0.069	0.290
Testing time [s]	0.047	0.059	0.234
Accuracy [%]	80	98	100

We can increase the accuracies using two thresholds with both MLM approaches since both methods can detect local and global anomalies. The MLM with only an upper threshold reached 80% accuracy, leaving local anomalies undetected (marked as red dots, below in Figure 49)). After applying the second threshold, the accuracy of the MLM detector with two thresholds was 98%, and

with the piece-wise approach, 100%.

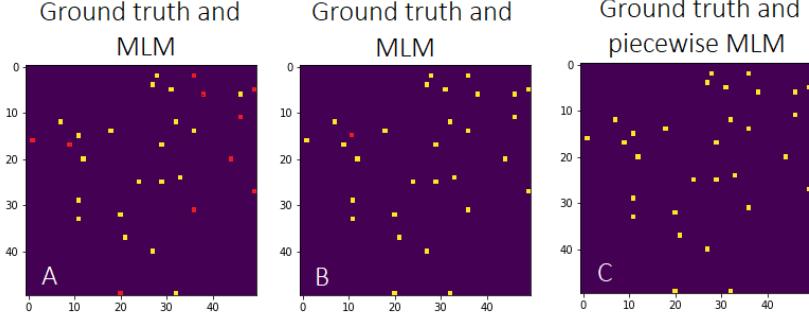


FIGURE 49 Joint anomaly detection maps of the ground truth and the results for the ColorChecker dataset. Yellow represents detected anomalies. Red spots are undetected local anomalies. A: MLM, only a higher threshold, B: MLM with two thresholds, C: the piecewise MLM

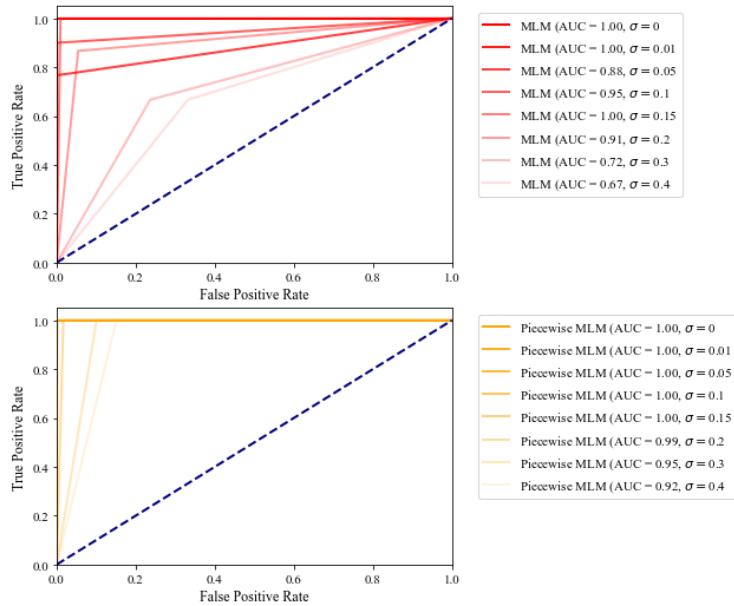


FIGURE 50 Comparison of the ROC-curves of the MLM and piece-wise MLM with a different amount of noise in the data (the ColorChecker dataset). The noise drawn from the uniform distribution was added to each pixel. The level of the noise is denoted with σ . The ROC curves show that both approaches had no false positives with a low noise level. When the noise level increases, the piece-wise MLM was significantly more robust with respect to the noise than the MLM.

The noise sensitivity test revealed that while the MLM anomaly detector might be faster, it is more sensitive with respect to noise than the proposed piece-wise approach. This can be seen clearly in above Figure 50, which visualises the area under the curve (AUC) and receiver operating characteristic curve (ROC) for the MLM and piece-wise MLM tests. The test was performed by adding different

levels of uniformly distributed noise to the training and test data. As a result, neither method had any false positives at first, but the number of false positives increased after adding noise. The effect was more drastic for the MLM than for the piece-wise MLM. The effect can be seen in how the increased noise affects the AUC. It ranges from 0.67 to 1.0 with the MLM and from 0.92 to 1.0 with the piece-wise MLM.

The test setups and results were implemented and computed with similar setups and data to the (Pölönen et al. 2020), and the results are presented accordingly. The presented results in this summary are from the artificial ColorChecker data tests, leaving the Finnish Forest remote sensing scene results and the comparison against the previous reference method comparison to be found in the included article. The ColorChecker data is captured with a VTTs prototype HSI (Saari et al. 2013), introduced in Section 2.1.7.2, and the data is recorded and pre-processed with a CubeView laboratory software.

Discussion

Since the results of ColorChecker and Forest data experiments confirm each other, we will answer the Question 1c "*How a distance-based machine learning model for classification can be adapted to local and global anomaly detection and how to improve the resilience with respect to noise?*" by focusing on the ColorChecker experiment results.

In this study, we used our previously developed variance-based anomaly detection version (Pölönen et al. (2020)), improved its performance to cover global and local anomalies, and increased the model's robustness with noisy data. As we now understand, the MLM model utilises distances from subset R and their labels T to the training data and its labels X, Y . As previously shown, the size of the subset R correlates with accuracy; by increasing the size of R , we can increase the model's accuracy (de Souza et al. (2015) and Article PI).

The new approach increased the accuracy without increasing the size of R . With colour checker data, the R size was 250 samples, containing randomly picked samples from all classes. The piecewise MLM model was trained class-by-class, re-labelling Y and T so that there were only two classes, an expected class and an anomalous one. We trained the models of the presented colour checker data results four times, which was the number of the ground truth classes. As a result, the class-by-class models seemed more accurate, even though the actual size of R remained unchanged. One of the reasons *why the piecewise MLM outperforms with the noisy data* is that the approach provides more accurately trained models since there are stricter rules on the deviations and expected values. Since the piecewise MLM is noise resistant, it is also more sensitive to detecting the local anomalies, even if they are close to the classes of the expected spectrum.

The key to extending the method to cover global and local anomalies was to use two thresholds. Low value for the locals, higher value for the global ones. In our MLM detector's first introduction (Pölönen et al. (2020)), we could not detect the local anomalies because the scale of the lower threshold values was

too high. With this second study and new experiments, we could also detect local anomalies with the original MLM detector. The key to the implementation and optimisation is that the MLM detector's lower threshold value must be set carefully, and only small enough values will work. However, with the piecewise MLM, the range of the possible threshold values is wider, and the model is not as sensitive as the original MLM detector.

Since both versions (once and piecewise-trained MLM models) are performing well, the question might be which to use. The decision should be made according to the data; If the HS image is large, containing global and local anomalies with unknown noise levels, the recommendation is to use the piecewise approach. With less noisy data, it is worth testing the MLM detector since it is faster to implement and performs faster with less computational complexity. However, both approaches are suitable for remote sensing applications, and further research is needed and welcome.

While approaching the published research results, we created the labels for the Forest HS data using k-means clustering. The ground truth masks were hand-drawn. The clustering was a fast way to perform the data transforming phases (Section 2.2: Figure 20). However, the randomly selected locations of the initial clusters seemed to affect the final results, causing variety among the reruns. The downside of the hand-drawn labels was the uncertainty around the corners and borders of the objects; in some cases, it was difficult to accurately label which pixel was anomalous and which was, for example, a little noisy normal pixel close to the anomalous one. Therefore, the model trained with the Forest data might have had faulty labels for the spectra, or the hand-drawn ground truth might have been biasing the accuracy results. A more robust method for creating labels is recommended for future research.

The review process was double-blinded, including improvement required from both reviewers. The author was instructed to clarify the aim of the study, correct the spelling and clarify how the anomalies larger than one pixel were defined and what the relation is with smaller objects (i.e. pseudo-anomalies among trees as shown in Article PI, Figure 10 A-D). The quantitative detection results were asked to be stated more clearly. The author was asked to clarify how the thresholds were selected and whether the forest data was an orthomosaic of the area.

The author clarified the aim of the study, and proofreading was done to correct the typos. The quantitative results of the forest data were improved with hand-drawn ground truth, computing accuracy results, and numerical results besides the visual evaluation. The threshold selection procedure and other questions were answered in the final version. The publisher has a limitation of 8 pages and two levels of proceedings; Annals for the high-level and Archives for the average-level articles. The manuscript is required to be stated and informative. The manuscripts are evaluated according to metrics shown in Table 7 in Section 3.1. Our manuscript was approved after the first revision and published in Annals. The article was awarded a live stream conference presentation time (2021) and the possibility to present a poster at the 2022 ISPRS live conference.

3.2.3 Self-learning model for classification

Updating Strategies for Distance Based Classification Model with Recursive Least Squares (Article PIII) answers to research Question 1d, "How to create a self-learning distance-based machine learning model which uses an extremely small amount of labelled training data?"

Summary

In this study, we updated MLM nearest neighbour classifier (NN-MLM) with a recursive least squares algorithm (RLS). The new self-learning MLM was tested with two hypotheses in two *Experiments, A and B*. The aim was to create a self-learning model which could be trained with an extremely small amount of labelled reference points, maintaining MLMs' ability to reach high accuracy.

We tested in *Experiment A* whether new classes could be introduced with RLS updates for the pre-trained self-learning-MLM model. In *Experiment B*, we simulated a bush-broom imager's working principles (Section 2.1.7.1) by updating and testing the model's performance with a continuous stream of pixels in a continuous scanning process. The aim of *Experiment B* was to train the model with a significantly small amount of labelled reference points and update it continuously with (RLS) to reach maximum classification accuracy quickly. Both experiments were performed using Pavia and Salinas datasets and Python programming language.

Methods

Below, Algorithm 7 shows the self-learning MLMs pseudo-code phases. The essence of an MLM is the assumed linear mapping between the input and output distances, which is visualised in Section 2.3.8 (Figures 33 and 34). The visualisation of the RLS updated linear mapping can be seen below (Figures 51 and 52). When new data points appear, the distances are computed similarly to the first model. Instead of re-training the model, we are updating it with the RLS algorithm.

The setup of *Experiment A* utilised RLS updates with classification rules, which are shown in Figure 53. The first model was trained with four classes from the Pavia dataset. The model was updated for each new class, and for the labels, the output distances were evaluated using classification rules. The assumption was that a majority of the new classes' misclassified data points are labelled into the biggest class (since all distances longer than the longest known distance are closer to the biggest class than to the other classes). Thus the distances of the biggest labelled class should be re-evaluated and divided into new classes. The training and test data were selected randomly to represent all current classes.

Algorithm 7: Recursive Least Squares (Romberg 2016)

Input: X, R, Y, T
Initialize:
Calculate distance matrix $\mathbf{D}_{x,0}$
Calculate distance matrix $\Delta_{y,0}$
Calculate $\mathbf{P}_0 = (\mathbf{D}_{x,0}^T \mathbf{D}_{x,0})^{-1}$
Calculate model
 $\hat{\mathbf{B}}_{x,0} = \mathbf{P}_0 \mathbf{D}_{x,0}^T \Delta_{y,0}$
for $i = 1, 2, 3, \dots$ **do**
 New data X_i, Y_i appears
 Calculate distance matrix $\mathbf{D}_{x,i}$
 Calculate distance matrix $\Delta_{y,i}$
 Calculate $\mathbf{P}_{x,i} = \mathbf{P}_{i-1} \mathbf{D}_{x,i}^T$
 Calculate $\mathbf{P}_i = (\mathbf{P}_{i-1} - \mathbf{P}_{x,i}(\mathbf{I} + \mathbf{D}_{x,i} \mathbf{P}_{x,i})^{-1} \mathbf{D}_{x,i} \mathbf{P}_{i-1})$
 Calculate $\mathbf{K}_i = \mathbf{P}_i \mathbf{D}_{x,i}^T$
 Update model $\hat{\mathbf{B}}_{x,i} =$
 $\hat{\mathbf{B}}_{x,i-1} + \mathbf{K}_i (\Delta_{y,i} - \mathbf{D}_{x,i} \hat{\mathbf{B}}_{x,i-1})$
end for

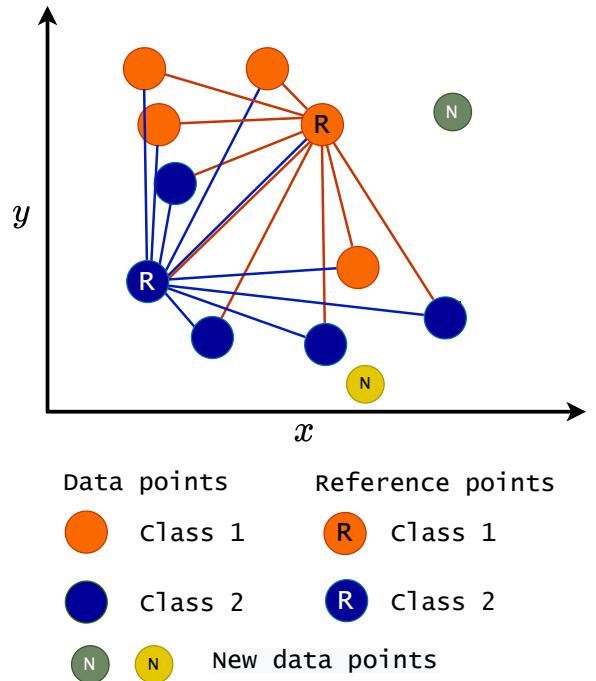


FIGURE 51 **New data points** brings new distance information for the self-learning MLM model.

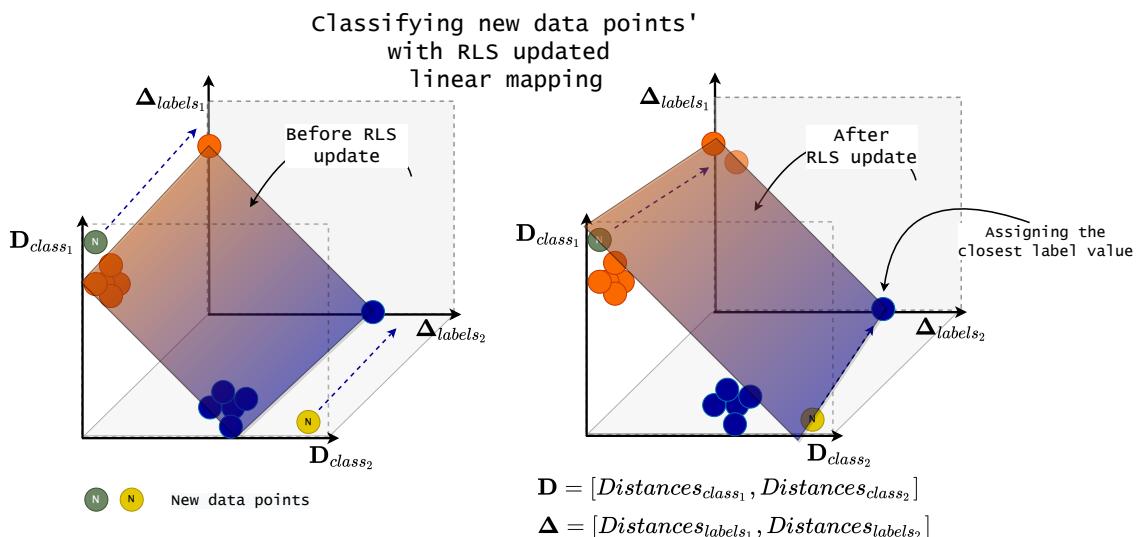


FIGURE 52 **RLS update.** A self-learning MLM model contains sophisticated class-wise information through the RLS updates. As seen above left, the new data points are not settling to the existing plane (\hat{B}). However, after the RLS update, the model adapts itself to the new distance information, and new data points can be classified based on the nearest neighbour values.

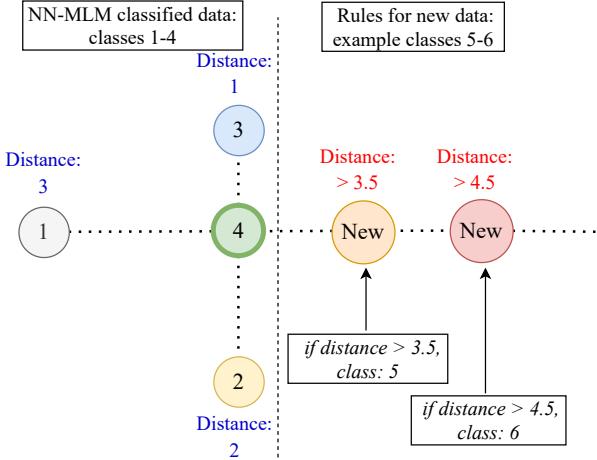


FIGURE 53 Classification rules for the new classes. Classes 1-4 were classified using the NN-MLM. The distances from the biggest known class four (4) are written in blue. The image shows the first updating rules. The longest distance of class four (4) is 3, and all distances longer than 3.5 were re-classified to a bigger class. The biggest class was updated, and again, the longest distance was set to 4.5 (5 - 0.5), and the data was re-checked. Each time a new class appeared, all of the biggest class data points were re-checked with this halfway limit as many times as we had new classes.

The *Experiment B* was implemented to use the input HS image row-by-row, top-down. Before computing updated model $\widehat{\mathbf{B}}_i$, the ground truth (\mathbf{Y}_i) of the new training data (\mathbf{X}_i) was predicted with the previous model $\widehat{\mathbf{B}}_{i-1}$. After the $\widehat{\mathbf{B}}_i$ was computed, and the prediction for the current test data stream was made, the model $\widehat{\mathbf{B}}_i$ and the pseudo-inverse matrix \mathbf{P}_i were passed to the following update. Therefore, the MLM in *Experiment B* was continuously learning. The first model could be trained using only a few known spectra and by computing only the distances between \mathbf{R} to \mathbf{R} . The model updates and results were obtained row-by-row.

Results

Experiment A; A new self-learning MLM method can classify new classes with RLS update but with a cost of decreasing accuracy. With a larger amount of reference points, one class can be introduced with reasonable accuracy; a model trained with \mathbf{R} size 800 reaches 0.94 accuracies with a new class; after the second new class, accuracy was 0.87.

Experiment B; A self-learning MLM, pre-trained with only *a few reference points*, is quickly reaching the accuracy of a once-trained model (0.99) within the first five updates. The row-by-row model's accuracy varied between 0.89 and 1.0. The self-learning MLM might be suitable for HS imaging and remote sensing applications, but further research is needed. The row-wise ground truth, classification and error maps of the Salinas data set can be seen below (on the left, Figure 54). The row-wise accuracy comparison between row-by-row tested self-learning MLM (next row as test data: RBR, whole test data: TDR) and once trained MLM

(OTM) is visualised below on the right (Figure 55).

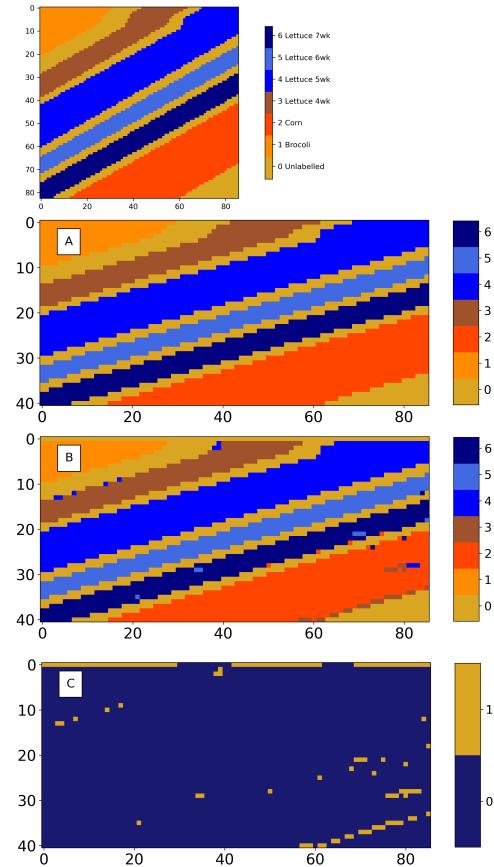


FIGURE 54 Classification and error maps of Experiment B. Top: Salinas HS image was divided from top-down, every second row to X or Y data sets. Row-wise approach updated and tested the model's performance over the Salinas HS image. Every second row was used for RLS and tested with the following row. A: Ground truth, B: Row-By-Row pixel-wise prediction map. C: Classification error map. Training R size: 20.

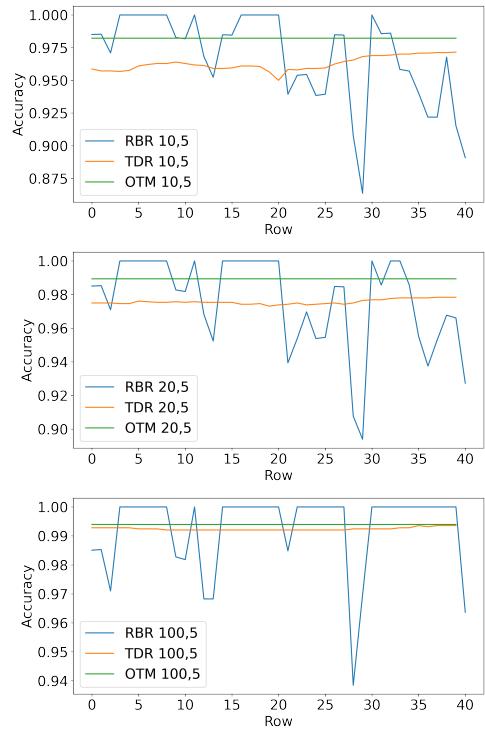


FIGURE 55 Visualisation of the self-learning MLM's accuracy results, Experiment B. RBR: self-learning MLM, one-row accuracy, test data is the original image's next row from the current update row. TDR: self-learning MLM, the accuracy of the whole classified test data, measured after very row-by-row model updates. OTM: once trained MLM model, tested with the whole test data. First image: *class-wise R size*: 10, number of neighbours: 5, second image *class-wise R size*: 20, number of neighbours: 5, third image *class-wise R size*: 100, number of neighbours: 5.

Discussion

The answer to the Question 1d: "How to create a self-learning distance-based machine learning model which uses an extremely small amount of labelled training data?" is by combining an MLM classifier with an RLS algorithm. Operating that way, we are creating an algorithm that can start with a few known spectra and learn more from the incoming pixel spectrum. As a result, the self-learning MLM could be a comparable machine learning method for the application of HS imaging and

remote sensing. In the future, it could be interesting to apply it in a real-time system. We conducted two experiments, of which the main outcomes were raised from the *Experiment B*.

When the model is implemented similarly to in *Experiment B*, the first model can be trained using only a few labelled samples per class, significantly reducing the expensive and time-taking step of preparing the data for analysis (Fundamentals of machine vision Chapter 1, Figure 4). The implementation of *Experiment B* is our strongest solution for the above-mentioned research question. Behind the interest of minimizing the training, data is the usual challenge related to HS image analysis; the availability of labelled data is limited (Ahmad et al. 2022). The low number of training data samples is also a benefit against many other state-of-the-art classifiers; for example, neural networks tend to overfit when the amount of training data is low (Planche and Andres 2019).

A self-learning model seems to be robust with respect to challenging or noisy pixel streams. The high accuracy can be reached fast, and the self-learning MLMs' overall performance does not suffer from difficult pixel streams. The overall accuracy remains high, as seen above right in Figure 55. The model can cover fast from heavily decreased one-row accuracy results while updating and predicting the labels on a continuous stream. *Experiment A* provided promising results for introducing one new class, but the method needs further development.

The review process of this article was the following; The three reviews were uniformly positive. The notes for author pointed out that the limitations of the first experiment demand further research. In contrast, the second experiment indicated that the Recursive Squares Estimator is a valid approach. Overall, fair writing quality, clear explanations and complete experiments received positive feedback. The structure was described as excellent; Aims, experiments and results are connected and supported enough. Some misspelling editing was asked. The article was proofread and submitted as it was and published in ISPRS Annals conference proceedings. The author recorded a presentation to the ISPRS digital library, and she was awarded a live presentation and poster session held in Nice 2022. At the conference, the article raised interest in being different - most of the presented methods were about neural networks, whereas a simple and elegant distance-based method stands out. Due to the interest and requirements, the author will publish MLM implementation functions from all three articles in Gitlab (Raita-Hakola (2022)), and basic information will be added to Wikipedia (pages and code release are currently under work, with the kind support of Nokia Foundation).

3.3 From sensors to analysis

This section aims to introduce three articles covering all fundamental machine vision phases from the sensor level design to HS imaging, image pre-processing and analysis. The first application area is in skin cancer research (Articles PIV and PV), and the second

scope is in laboratory HS imaging and analysis tools (Article PVI.)

3.3.1 3D Spectral imaging system for complex skin surfaces - toward optical biopsy

Articles "FPI Based Hyperspectral Imager for Complex Skin Surfaces - Calibration, illumination and application" (Article PIV) and "Differentiating Malignant from Benign Pigmented or Non-pigmented Skin Tumors - a Pilot Study on 3D Hyperspectral Imaging and convolutional neural networks" (Article PV) answers to research Question 2a; *"How to develop and clinically pre-test a hyperspectral imaging system for complex skin surfaces in skin cancer research? How does the target-wisely designed system perform in technical, user-related and classification tasks?"*

Summary

This is a summary of two continuation articles from the Spectral Imaging of Complex Surface Tomographies (SICSRFIS) project, funded by the Academy of Finland. The first article forms a technical foundation and reveals the results of the first clinical trials. The second article dives deeper into the clinical pilot study, focusing on CNNs and methodological development. The analysis methods and imaging practices are improved based on the results and notions from the first study.

HS imaging provides non-invasive methods for medical imaging applications with high spatial and spectral resolution. The included articles utilise HS imaging for delineating malignant tumours, detecting invasions, and classifying lesion types, aiming toward an optical biopsy. According to previous literature, typical challenges of these applications relate to complex skin surfaces, leaving some skin areas unreachable.

We introduce a novel HS imaging concept and conduct clinical pre-tests within Articles PIV and PV. The SICSRFIS spectral imager concept combines a piezo-actuated Fabry-Pérot interferometer (FPI) based HSI, a specially designed LED module and several sizes of stray light protection cones for reaching and adapting to complex skin surfaces. Besides traditional HS imaging, the imager is designed for the skin surface models (3D) for each captured wavelength. The captured HS images contained 33 selected wavelengths (ranging from 477 nm to 891 nm), which were captured with accordingly selected LEDs and three specific angles of light.

The first pre-test results of differing two lesions and healthy skin (Article PIV) show that the new SICSRFIS imager enables the use of the spectral and spatial domains with surface model information. The imager could reach complex skin surfaces. For the optical biopsy, healthy skin, basal cell carcinomas (BCC), and intradermal nevi (ID) lesion pixel spectra were classified, and the lesions were delineated pixel-wise with promising results. The results were obtained with a convolutional neural network.

Based on the first study's findings, the acquisition procedure, data trans-

formation and analysis methods were improved in the continuation study. The second study included 42 lesions which were studied in two analyses. Our first setup analysed seven (7) melanomas against 13 pigmented lesions. The second setup had seven (7) IDs, 10 BCCs, and five (5) squamous cell carcinomas (SCCs) against healthy skin. All lesions were excised for histological analyses and annotated by dermatologists.

Methods

The Article PIV covers widely technical details, including the imager's design, calibration, components, imaging logic, and imaging setup with the results of the first pilot data gathering. Besides the article, the details of the imager, device workflows and user interface can be found in Sections 2.1.7.2, 2.1.8 and 2.4.1.

The pilot study was conducted as follows in both articles. After the data was captured, it was pre-processed and transformed as shown below in Figures 56 and 57. After data processing details, Figure 58 visualises how annotation masks were used in pixel selection. Lesion masks and healthy skin masks were combined to select the 250 lesion pixels and 100 healthy skin pixels into the training set. The HS cubes were vertically sliced (slice-half approach, white line in Figure 58 A). The training data was selected from the HS cube's left side, and the according test pixels were selected and windowed from the right side of the HS cube.

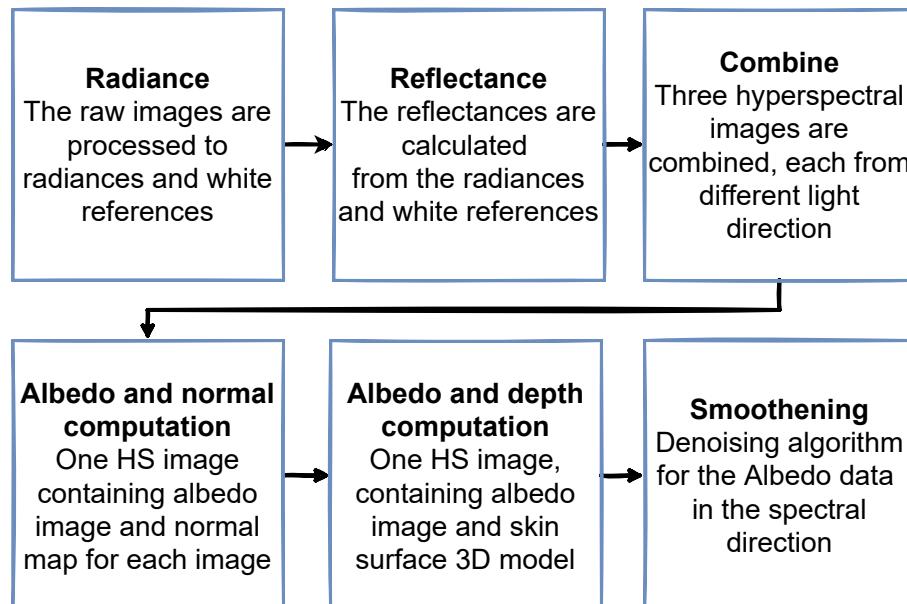


FIGURE 56 Pre-processing phases of the SICSURFIS 3D data. Total of six HS cubes, each captured with VNIR or VIS LED illumination, and one of the three angles was pre-processed in six phases. As a result, we had one HS data cube, containing 33 albedo and 33 skin surface 3D frames.

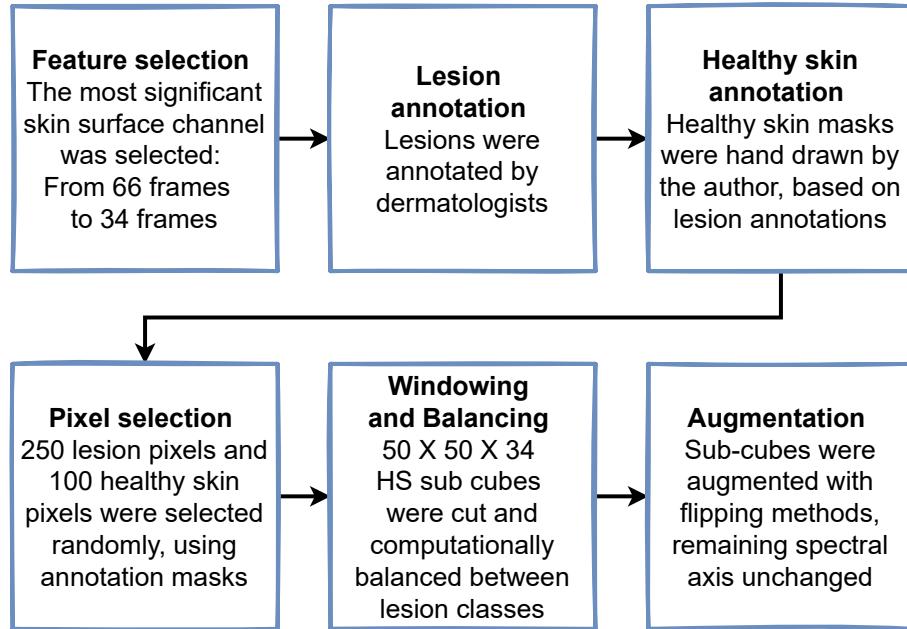


FIGURE 57 Transformation phases of the SICSURFIS 3D data. The most significant skin surface model was selected, and lesions and healthy skin were annotated. HS cubes were windowed around the selected 250 lesions and 100 healthy skin pixels into sub-cubes. The lesion classes were computationally balanced and augmented.

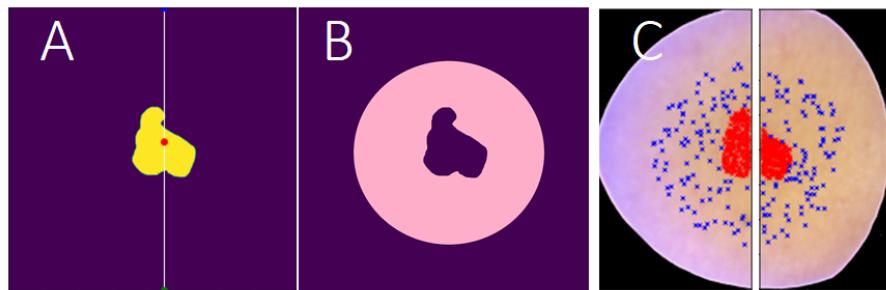


FIGURE 58 The pixel selection. HS images were annotated, vertically cut into training (left side) and test (right side), and the pixels were selected using lesion masks (A) and healthy skin masks (B). Image C visualises selected pixels; red represents lesion pixels and blue healthy skin. Each pixel was windowed into $50 \times 50 \times 34$ HS subcubes.

The transformation phase continued with analysis, which was done using CNNs. Article PIV had a classifier for two lesion types and healthy skin, and article PIV had four class classifier for differentiating three lesion types and healthy skin pixel spectra. Below, Figure 59 shows the basic structure of the used CNNs. We used 3D CNN to extract features from the spectral data (33 albedo channels) and 2D CNN for the depth frame features. Our convolutional layers used LeakyRelu and max-pooling functions. The outputs were concatenated and flattened and fed to the hidden layers. The model had an Adam optimiser with

default parameters and a categorical cross-entropy loss function. The dropout was set to 0.5.

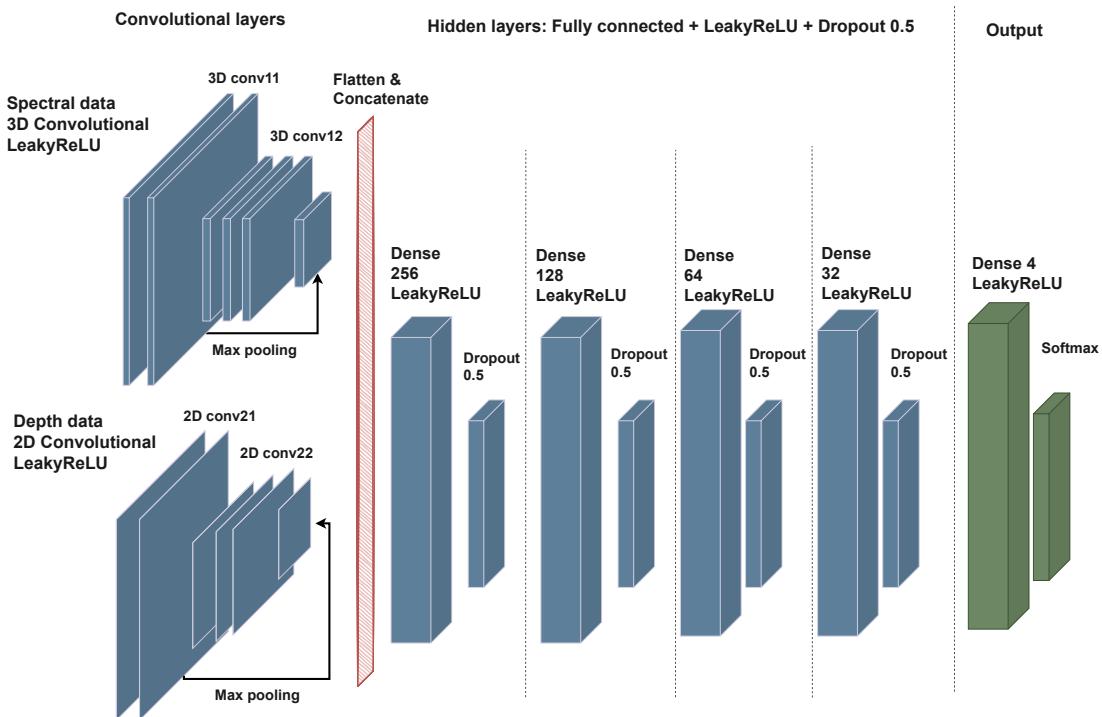


FIGURE 59 SICSURFIS CNN. We used 3D layers for the albedos (spectral data) and 2D layers for the 3D depth data. Results were flattened and concatenated and used as input for the hidden layers. The result was a three or four-class classifier for spectral data with depth information.

Results

The results of the first article PIV confirmed the system's suitability for complex skin surfaces. The imager could reach previously impossible areas, and the results confirmed that 3D HS imaging could be worth continuing studies for future clinical applications. As an example, we can see below from Table 9 and Figure 60 that the pixel-wise classifier performs relatively well in delineating and differing the malignant Basal cell carcinoma (BCC) and benign Intradermal nevi (ID) lesions.

TABLE 9 Classification report of the pixel classification, slice-half model.

	Precision	Recall/Sensitivity	F1-score	support
Healthy skin	0.63	0.81	0.71	1776
Bening intradermal nevi	0.87	0.81	0.84	1697
Basal cell carsinomas	0.89	0.76	0.82	2477
Macro avg.	0.80	0.79	0.79	5950
Weighted avg.	0.81	0.79	0.79	5950

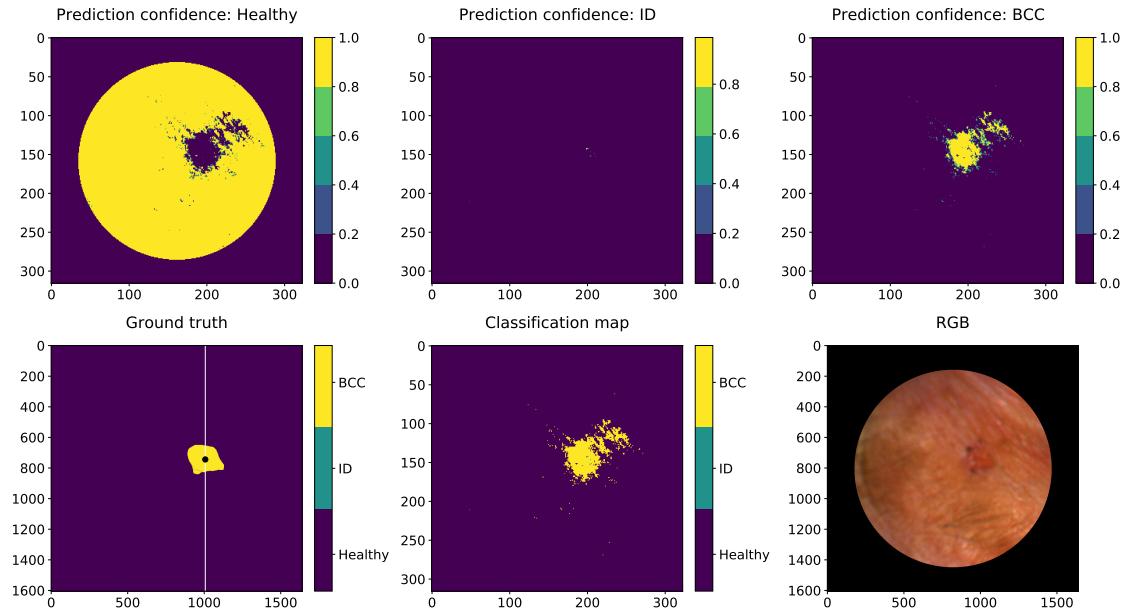


FIGURE 60 An example prediction map confirms that the CNN model was able to differentiate nodular Basal cell carcinoma (BCC) pixel spectra from the pixel spectra of Intradermal nevi (ID) and healthy skin.

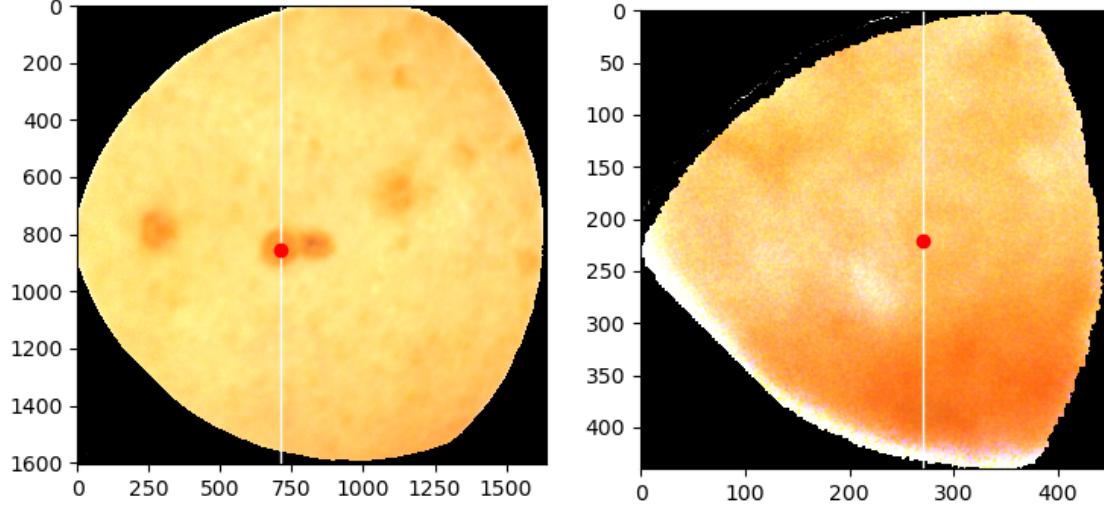


FIGURE 61 An example of the acquisition-related challenge. The SICSURFIS imager had four stray light protection cones, of which the smallest one produced images with decreased image quality. On the left, we have an example of a good-quality HS image; on the right, we can see how the smallest cone affects the image quality; the region of interest that can be used is remarkably smaller, and the image appears noisy.

The results were promising; however, they revealed some challenges hiding in the previous machine vision fundamental phases, which were used to improve the practices and results of the continuation study. Above, as an image

acquisition-related challenge, Figure 61's RGB reconstructions visualise how using the smallest stray light protection cone deteriorates the image quality. The smaller cone produced low-quality images, noise and a small region of interest, which eventually led to the decision to leave these images from the study. As a device-related learning outcome, the images to the article PV were captured, excluding the smallest cone.

Another note from the results obtained with the first article was the possibility of mistakes in the lesion border areas. The annotation was made hand-drawn by the dermatologist. The CNN classified lesions a few pixels wider than the ground truth, leading to the decision to create a safety border around the lesion annotations to avoid selecting uncertain pixels and provide accurately annotated healthy skin and lesion pixels. Below, Figure 62 visualises the improved method, which was used when selecting the pixels for the continuation study (PV).

At first, we automated the healthy skin annotation for creating the masks for pixel selection. Instead of drawing the healthy skin masks by hand, the author developed a masking algorithm that used the lesion width, height and middle pixel location. Using the input values, the algorithm drew binary images with healthy skin mask circles around the lesion areas. The lesion border was applied by copying the binary lesion masks and drawing new ones using image dilate and erosion functions. Those functions enlarged and shrank the lesion edges, enabling us to draw the border image with simple subtraction between enlarged and minimised lesion masks.

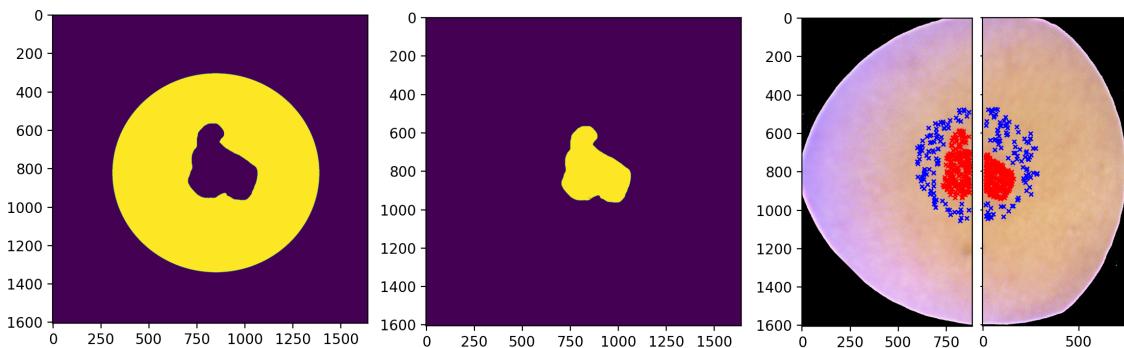


FIGURE 62 Pixel selection improvement. The results of the first analysis revealed possibilities to improve the border areas classification accuracy by improving the pixel selection rules. The windowed pixels of article PV were selected, leaving a safety area around the lesion borders, ensuring that the CNN will receive more accurately annotated data.

After the imaging procedure and transformation methods were improved, the second study was performed with two experiments. The experiments were performed for pigmented and non-pigmented lesions. At first, the CNN classifier had the non-pigmented lesions, the windowed pixel spectra from the healthy skin and ID, BCC and SCC lesions. The second setup compared the pigmented BCC and Melanoma lesions with healthy skin.

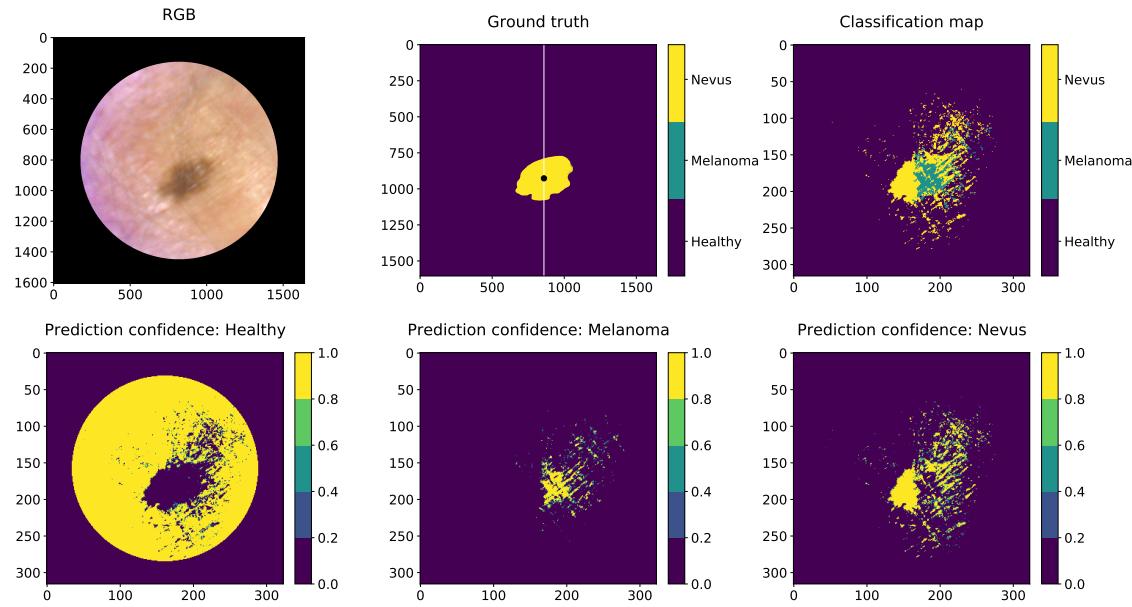


FIGURE 63 Prediction map of a pigmented nevus. A 10 mm low-grade dysplastic compound nevus of the chest, clinically and histologically diagnosed as a benign nevus but classified as melanoma by the SICSRFIS system.

The pixel-wise analysis provided prediction maps and classified pigmented lesions with sensitivity and specificity of 87% and 93% and non-pigmented lesions with 79% and 91%, respectively. The majority voting analysis provided the most probable lesion diagnosis and correctly diagnosed 41 of 42 lesions. Above, Figure 63 is an example of the pigmented classification. The test data (right half of the image) include pixels classified as melanoma and nevus. However, according to the majority voting, there were more melanoma pixels. Light reflection caused probable artefacts in the surrounding area of the lesion.

Method validation

The editorial board of article PIV insisted on leave-one-out experiments to be performed. The reason was to validate whether the half-slice approach was biased. We sliced half each HS data cube into train and test sub-cubes where the pixel spectra samples were selected. The editors' fear of biased results arose from a misinterpretation. The pixel-wise HS image classification was confused with being similar to image classification, which is more prone to bias compared to the pixel-wise spectral approach. The method validation (slice-half pixel-wise classifier against leave-one-out) results can be seen from below, Table 10.

TABLE 10 Leave-one-out (L-O-O) and slice half pixel classification accuracies

Special feature	Lesion ID	L-O-O	Slice-half	Unique features
Typical	1	0.79	0.94	
Typical	9	0.68	0.96	
Typical	8	0.62	0.84	
Typical	7	0.83	0.90	
Typical	10	0.77	0.85	
Typical	4	0.70	0.81	
Typical	15	0.69	0.79	
Typical	14	0.64	0.85	
Average (typical)		0.72	0.89	
Unique	5	0.23	0.92	Broken skin,dark skin tone, convex surface
Unique	3	0.56	0.76	Broken skin,dark skin tone
Unique	2	0.25	0.68	Wound or scab on top of lesion
Unique	17	0.26	0.94	The lesion and the healthy skin is covered by hair
Unique	12	0.003	0.85	Dark skin tone and red-brown lesion tone
Unique	11	0.49	0.71	Clear pigment, small lesion
Unique	13	0.19	0.81	Two types of lesions: naevus pigmentosus junctionalis on right side of the ID
Unique & difficult	16	0.16	0.47	Overall fair skin tone,small lesion (2mm), fair toned and uneven delineation
Unique & difficult	6	0.38	0.45	Lesion is pigmented, uneven skin tone around lesion
Average (unique & difficult)		0.28	0.73	
Average (all samples)		0.47	0.81	

Discussion

This pilot study, with its two articles, indicates that our non-invasive HS imaging system, which involves shape and depth data analysed by convolutional neural networks, is feasible in differentiating between pigmented and non-pigmented skin tumours, even on complex skin surfaces. There was a strong discussion over the slice-half method and possible bias in the results, but as the discussion in the article PIV underlines, the reason for this clinical pre-study was to introduce and pre-test a new imaging system. Without the first models and results, it would have been impossible to evaluate the technical aspects, how the imager reaches the complex surfaces, can those images be classified by CNN. It was important to test how the imager's special LED module, stray light protection and optical components affect the image quality and the results and finally evaluate the clinical procedures. The continuation studies were developed based on the first phases, as seen in the results of the article introduction.

Another note can be seen from the validation results (above, Table 10). One could say that the accuracy of the leave-one-out approach is the real performance,

but it is not the whole truth. Another opinion is that due to the unseen features, the leave-one-out results reflect the selection bias on the collected data. A model cannot classify unique features and lesions it has not seen and learned. The results of both approaches indicate that the proposed models would be practical with a larger amount of training data, which contains no unique lesions. So far, the selection bias could be minimised. The hypothesis for future studies is that the accuracy will increase when the collected data enables the models to generalise better. The accuracy difference between the leave-one-out method and the slice-half method is expected to decrease as the data amount increases. The level of selection bias should be taken into account in further studies.

The review process of the first article PIV was demanding. The reviewers approved the article after major and minor revisions. The following weeks were more or less challenging since the editorial board misinterpreted pixel-wise and image classification, which started the method validation and bias conversation. After several careful revisions, validation results and a literature review of the bias and image analysis, the editor finally approved the article. The article was improved by its content and structure. It expanded the author's knowledge of the results, and she learned to be slightly suspicious while reading her and others' results. Sometimes improvements and good results might be something other than the new state of the art - caused by a hiding artefact or good luck. Now, six months later, article PIV has raised interest at the MDPI, and the author has received two invited author acknowledgements from the editors of Sensors.

Article PV submitted to MDPI's Journal of Clinical Medicine, with 4.242 as the impact factor. The peer-review process was similar to article PIV. The article was carefully written with a detailed methodological description and deep medical discussion. The language was checked before the first submission. The authorship was equal between the dermatologist PhD student Vivian Lindholm and the author of this dissertation. The article received the following notes from two peer reviewers:

Review report 1: "This is a very useful and innovative manuscript aiming at the comparison of sophisticated, yet practical, methods in order to better differentiate malignant from benign pigmented or non-pigmented skin lesions. Congratulations on further mastering this approach. A minor English spell checking is needed."

and

Review report 2: "The authors describe interesting findings on hyperspectral, a new image diagnostic tool. The manuscript will describe the differences, convenience, and accuracy between HIS and conventional imaging tools. There are no problems with the research method and the description of the results. The limitations and future potential of this study are also accurately shown."

The editors accepted the article based on the peer reviews, and it was published as it was. It was submitted after the first MDPI technical article but published before that one.

3.3.2 Combining new imaging technology with user-friendly imaging and analysis software

Miniature MOEMS Hyperspectral Imager with versatile analysis tool PVI answers to research Question 2b "*How to design and implement a versatile user interface with analysis tools for hyperspectral imaging, and provide the experience of easy and correct imaging for any imaging scenes and scientists with no or little background in spectral imaging?*"

Summary

Article PVI was the author's first article, written before she started her PhD studies. She participated as a software engineer in a project where her position was to design and develop user-friendly data capturing and analysis software for VTT's prototype imager. The imager was to be presented at Finland Pavillion at the Photonics West, one of the world's largest photonics conferences. The paper is written in an introductory style, and the purpose was to highlight the new microoptoelectromechanical system (MOEMS, MFPI) technology, enabling miniaturised HS imagers' production. The main differences between the piezo-actuated Fabry-Pérot interferometers (FPIs) and tunable MFPIs are the actuation, physical size and the way the system can be manufactured. MFPIs enable volume-scalable manufacturing, thus potentially being a major game changer with low costs and miniaturisation advantages.

Article PVI introduces the utilisation of an FPI, discusses the differences between FPI and MFPI, and describes how an HSI can be constructed from matching optical assembly, driving electronics and imaging sensors. It discusses the calibration of HS imaging systems to acquire high-quality HS data.

The motivation for the author's contribution is the known challenge of the applicability of the produced raw data. Typically, the applicability is relatively low, and an application specific software is necessary to turn data into meaningful information. In this study, we introduce a user interface with versatile analysis tools aiming to breach the gap between raw HS data and the user application. The following subsections will concentrate on the author's contribution, leaving the article's microoptomechanical engineering details to the article.

Methods

The CubeView analysis software is developed with Python programming language, using PyQt5 (Riverbank 2019) and Python libraries Fippy, Camazing and Spectacular, which are developed at the Spectral Imaging Laboratory at the University of Jyväskylä (JYU 2022). The software was used to promote the novel MFPI spectral imagers at the conference; the pavilion imaging setup can be seen at the end of this article introduction, Figure 64.

The idea was to develop versatile image capturing and analysis tools suitable for VTTs prototype MFPI and FPI HS imagers. The system is built with

independent tools, including, for example, LiveView (video tool with FPI tuning function), HS cube PreView, device settings view (all possible settings can be changed, and the feature cross-connections, explanations and feature restrictions can be seen), configuration (i.e. which channels to capture) and a tool for HS cube analysis.

All of the tools are modular and independently usable, i.e. while the imager is not connected, the system can be used for analysis, and vice versa, the system can be used only for advanced imaging. The analysis backend is developed to be extended with new algorithms, which can be included by writing them in a certain syntax and saving them into a directory at the computer. Once the path is given, the system will remember it for the next time. Besides the user's algorithms, some selected algorithms, including spectral unmixing based on vertex component analysis and fast least-squares approach, spectral angle mapper, principal component analysis and library of spectral indices, are integrated into the basic version of CubeView Analyze. The current CubeView version uses Windows as an operating platform, but the system can be easily transformed into Linux or macOS devices. The software is a standalone executable made with PyInstaller (Cortesi 2022).

The software controls the device workflows, allowing the user to either capture images easily with guided workflow or reveal all the sensor specs via the GenICam GenTL transport layer (Section 2.1.1.2), enabling all the unlocked features to be controlled. The HS data can be saved (following the computer vision fundamentals, Figure 20, Section 2.2) as raw data, which needs further pre-processing, including possible demosaic with interpolation (Section 2.1.2.5), and data transformation to HS data (Section 2.2.1). Alternatively, the user can acquire, pre-process and transform the data into spectral cubes, and save them into netcdf4 (Unidata 2022) form, for example, as "radiance" or reflectance HS cubes, and continue analysis, with or without the interface.

Results and discussion

Since the resulting first MFPI-compatible version was published, the author has developed several versions of the software. The versions include the laboratory version, using an FPI HSI, and the special version with LED device controls and user needs in the hospital environment, used in articles PIV; PV. Figure 2 in Section 1.2 is produced using CubeView PreView and analysis tools as an example of everyday laboratory use. The image is captured in the past, but it can be re-opened, pre-viewed frame or wavelength at the time, and for example, the pixel spectra can be shown.

The article was published through a similar minimal peer-review process to article PI: The extended technical abstract was submitted and evaluated, and the research was approved for the conference. The article was submitted within a given timeline. The content and the written article were inspected and approved by SPIE.

As it is written, this article does not bring new innovations or scientific

breakthroughs made by the author. However, the aim and outcome were to lower the threshold of new applications around HS imaging. As we learned at the beginning of this dissertation from (Marx and Fuegi 2021), it may take 17 years from the scientists' table into a real-life application. Software like CubeView enables multidisciplinary entries; users with basic knowledge can test their hypothesis, whether there could be a phenomenon with new applications to be found beneath the wavelengths of light.

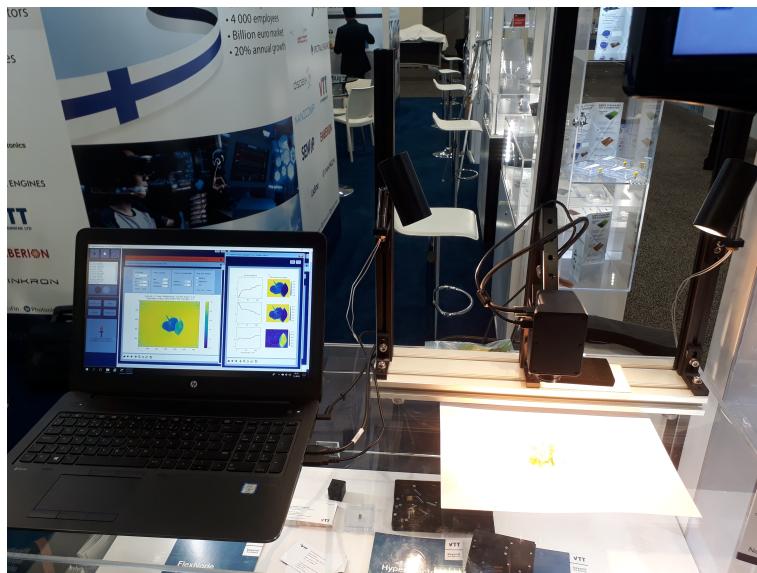


FIGURE 64 **CubeView software** was introduced with VTT's novel miniature MEMS FPI spectral imager at the Photonics West 2019 Finland Pavillion. The presenting authors at the conference in San Francisco were Roberts Trops and Ilkka Pölönen. Image: Roberts Trops

4 DISCUSSION AND CONCLUSION

This section aims to finalise the dissertation, by discussing the application domains, inspiration and importance (Section 4.1) and machine vision fundamentals' relation to the research questions (Section 4.2). Section 4.3 will visualise and conclude the relationship between the articles and the fundamentals. The journey ends to Section 4.4, which show, how the fundamentals will be used in author's future works.

4.1 Application domains, inspiration and importance

The application domain of this study is spectral imaging; however, its scope is wider. The theory explains the fundamental phases of machine vision with selected examples, which can be applied and utilised widely for machine vision imaging applications and domains. A reader who implements their machine vision applications based on the theory of this dissertation is not expected to be an expert in micromechanics, optics or imaging technologies. The research is aimed at individuals with sufficient application-specific knowledge to define the specs of the research design and some programming and data analysis experience. The conclusion regarding the usability of this dissertation theory is to be seen. The author would be pleased if, for example, the visualisations find their way into lecture notes and other materials.

The conducted theory introduced; the machine vision fundamentals. The aim is to inspire students and scientists to build machine vision systems by lowering the application threshold. The hypothesis and research questions were tied to the theory, and the chapters were written and visualised to follow the fundamentals of machine vision.

On a general level, by using new imagers and imaging systems, we can enrich the whole machine vision research field one study at a time. If the training and the testing data source is a target-wisely optimised imaging system, it might benefit the system's performance towards bias and other previously mentioned challenges. Besides new applications, the whole research field benefits from the

new data sets, enriching the current standard data selection.

It is known that the need for large data sets still exists depending on the methods used. However, the more versatile the proposed methods, models, frameworks and systems are tested, the more information is gained for scientists to argue and discuss the benefits, possibilities and weaknesses. By designing and implementing target-wisely optimised image-capturing processes, the acquired data has a high quality, which strongly influences the overall performance (Tian et al. 2021; Iqbal et al. 2018). At the end of the machine vision fundamentals, high-quality optical imaging provides outputs with higher accuracy. The data may require less computation which eases the computational bottleneck of many real-time machine vision systems (Ren et al. 2021; Meribout et al. 2022). Depending on the targets, we can decrease the amount of unnecessary data gathering and reduce the computational complexity through the decreased need for pre-processing and transformation algorithms. These changes can ease the edge computing challenges related to platforms' power consumption and computational capacity (Meribout et al. 2022).

4.2 Machine vision fundamentals and research questions

This dissertation encourages readers to design and select the system components and pay attention to the environment and the requirements of the pre-processing, transformation and analysis methods. It underlines the importance of all machine vision fundamental phases and answers to the research questions through aims. The data-analytical computer vision aspect was to show how computationally effective machine learning methods can be optimised, for example, to edge computing in real-time applications. The aim of this study was presented with two research questions, which were divided into six sub-questions (Section 1.5). Those questions will now be answered through the fundamentals of machine vision.

Question 1: "How to optimise a computationally effective distance-based machine learning method for real-time applications?".

We approached this question by developing and exploring new versions from an MLM classifier (de Souza et al. 2015). MLM is a method that could be suitable for edge computing since it is distance based (relatively robust with respect to noise), relatively easy to implement, fast and computationally efficient. Our research has indicated that MLM is suitable for HS image classification and anomaly detection, comparable with current state-of-the-art methods. Based on our experiments, it is possible to optimise the analysis method for HS data and find solutions for the according sub-questions.

With Question 1a, we examined "Can we reduce the method's computational complexity by selecting the reference training points intentionally and maintain a high accuracy?".

We explored this question with the PC-MLM classifier. Our approach utilised

pre-processing, transformation and analysis phases from the computer vision fundamentals (presented in Figure 20, Section 2.2). The data pre-processing phase was minimal; data were normalised, and unlabelled pixel spectra were removed. The transformation phase was the key to the new approach; the PCA algorithm was used to reveal the geometry of the data in order to find the minimum, maximum and mean spectra "skeleton" from each class. Finally, the method was trained and tested with and without the new approach, and the outputs were compared against known reference methods. As a result, we could find use cases (especially large HS data sets) that benefit from the new approach by paying attention to the transformation phase. The method is easy to implement, computationally light, and fast, and reaches for high accuracy. The downside of our research is the use of standard data sets. It would be interesting to see how MLM performs in a target-wisely built machine vision system. Will the accuracy decrease, and is it a fast and light enough classifier to be implemented in a real-time platform, such as a drone?

The second Question 1b was, *"How a distance-based machine learning model for classification can be adapted to local and global anomaly detection and how to improve the resilience with respect to noise?"*.

The question was answered by exploring MLM in anomaly detection. The fundamental phases were conducted as follows. The pre-processing phase included normalisation. Labelling, one of the transforming tasks, was done using a k-means algorithm, and the actual methodological development was done in the analysis phase. The approach included a one-against-all training algorithm combined with lower and higher thresholds for the local and global anomalies. As a result, our previous (Pölonen et al. 2019) experiments were improved and against the original MLM algorithm, our version was more resilient with respect to noise and, again, comparable to other state-of-the-art anomaly detectors. The experiments were done using artificial and Finnish forest data, leaving out the standard datasets' possible effects.

The third Question 1c asked, *"How to create a self-learning distance-based machine learning model which uses an extremely small amount of labelled training data?"*.

We studied this question by creating a self-learning MLM classifier. This approach utilised pre-processing and analysis phases. The pre-processing phase was minimal, including normalisation and removing the unlabelled data. The more successful *Experiment B* was simulating a real-time system with an incoming pixel spectra stream, so the data was divided; every second row was for training and every other testing. *Experiment A* used half of the data for testing and the other half for selecting the training reference points. The MLM classifier was continuously updated with the Recursive Least Squares (RLS) algorithm during the analysis phase.

The first experiment (*Experiment A*) tried to train the model with a few classes and introduce new classes "on the flight". Our model could classify one unknown class with reasonable accuracy, but overall accuracy decreased after each new class and update. The idea worked, but the algorithms inside the new classifier method must be developed further. The second experiment, *Experiment*

B, was a success. The model could be trained using only a few reference points, and it could quickly reach the accuracy of the once-trained model. Therefore, we could reduce the need for labelled training data (transforming phase). However, our case used standard data sets with their known labels. It might have affected the results, and the decision to use standard data removed the transformation phase (labelling etc.) from our implementation, making it look slightly lighter process than it would be without known ground truth. In the future, this new approach and version of the MLM classifier could be interesting to test in a real-time application's pixel stream.

Since the aim of this dissertation has data-analytical and technical domains, the aim was divided into two main questions. After presenting the data-analytical aim and answers to Question 1, which were here to bring new efficient analysis methods and studying the computer vision fundamentals, it is time to wider the scope over the machine vision fundamentals. Besides the MLM-related articles, the conducted research presents how a target-wise machine vision system can be designed and built from the sensor and component level to the user interfaces and analysis tools, including all fundamental phases of machine vision.

The second main Question 2: "*How to design and build machine vision systems target-wisely?*" is to be answered through two following sub-questions: Question 2a was "*How to develop and clinically pre-test a hyperspectral imaging system for complex skin surfaces in skin cancer research? How does the target-wisely designed system perform in technical, user-related and classification tasks?*".

We can answer Question 2a by discussing the Articles PIV and PV. The SIC-SURFIS HSI was developed and used in a process that followed the machine vision fundamentals. The SIC-SURFIS articles covered all phases from the device level design and manufacturing (VTT) to the optical image acquiring (Helsinki University Hospital, (HUS)), image pre-processing (computing spectral data from raw data), pre-processing (computing albedos, surface normals, 3D models and denoising), transforming (annotation, segmentation, cutting pixel windows, data augmentation) and finally analysis with convolutional neural network (CNN).

The system requirements, design and implementation decisions were made through all fundamental phases, evaluating the choices that affect the performance of the other phases. For example, the high resolution, the CNN's requirement, was adopted for sensor selection. Similarly, the wavelengths of interest were selected target-wisely, which defined the LED illumination (which LEDs to mount), and the requirements of computing the photometric stereo images (3D) were behind the three selected illumination angles. By knowing which wavelengths to capture, there was no need to capture unnecessary data - depending on the calibration, the imager can provide hundreds or thousands of wavelength channels, but only 33 were selected for the research, which reduced the need for computational feature selection methods. The user interface was developed to control the devices in a way that guided the users to follow the right image-capturing procedures, resulting in as high-quality data as possible. The settings were pre-designed and hard-coded (adjustable through an advanced settings tool) to maintain the exposure time and other important parameters un-

changed.

This project taught us that the way the optical images were captured during the clinical phases affected the results. The smallest stray light protection cone was abandoned due to a limited field of view. The final results were affected by the steadiness of the user's hand and how the system's sharpness was manually set before capturing. The unwanted stray light, reflections, shadows caused by oily or creased skin surface, or a complex shape (e.g. nose tip) could affect the image quality. In our continuation study, those results and notes from the first acquired data were used to improve the quality of the following data-gathering phases.

The final Question 2b was, *"How to design and implement a versatile user interface with analysis tools for hyperspectral imaging, and provide the experience of easy and correct imaging for any imaging scenes and scientists with no or little background in spectral imaging?"*.

We can approach this question by exploring Article PVI, which introduces the foundations of the CubeView user interface. It all starts with user needs, environmental issues and hardware requirements. CubeView software covers all fundamental phases of machine vision, from image acquisition to pre-processing and transformation to analysis methods. The outcome is versatile; images can be captured, pre-processed and transformed into HS cubes, which can be saved to further analysis. On the other hand, with the system, we can test new applications and demonstrate the basics of spectral imaging.

4.3 Conclusions

The machine vision systems should be designed, built and evaluated through the machine vision fundamental phases.

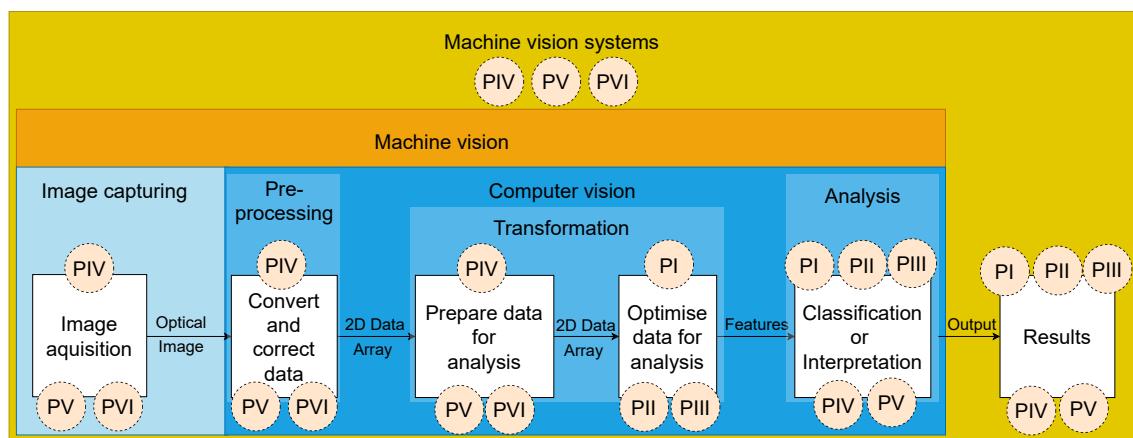


FIGURE 65 The relationship between the articles and machine vision fundamentals. The theoretical foundation has been visualised through the dissertation. The final version of machine vision fundamentals concludes articles with the fundamental phases.

In the beginning, we were introduced to machine vision fundamentals (Figure 4, Section 1.3). Now, we can draw the same picture and see, from Figure 65 (above), how the conducted research relates to the fundamentals.

Throughout the journey, we have seen how the fundamental phases are related to each other. The hypothesis *The machine vision systems should be designed, built and evaluated through the machine vision fundamental phases* has been surrounded with examples. Image capturing, pre-processing, transformation, and analysis methods can be developed separately from each other. However, when evaluating the performance and accuracy of a machine vision application or system, we will evaluate the joint performance of all previous phases. By designing systems phase-by-phase, we can, for example, select the wavelengths of interest and reduce unnecessary data gathering or adjust the illumination and sensor settings to provide high-quality optical images, which reduces the computational burden of image corrections. The data analysis development for different application areas benefits from available data, so it could be a good idea to develop new pilot systems, provide specialised data sets for unrestricted use, and share the technical specs of imaging setups.

In the end, there might be a question, can one person do it all? A small pilot system is doable, but it is great to combine the research group with scientists who are specialists in the deep and narrow phases of machine vision fundamentals while increasing the ambition. Besides the researchers with technical and mathematical skills, the science around machine vision applications benefits from multidisciplinary researchers, who will share a deep level of understanding of the application area.

4.4 Future work

For the future, a small, two-channel spectral imager is presented in Figure 66 (below). The imager is designed and built, following the theory of machine vision fundamentals, to be a small, easily adjustable imager with tunable band-pass filters from 380 nm to 800 nm with 10 nm FWHM. This future project aims to deliver a simple spectral imager and wavelength optimising software for enabling the most appropriate target-wisely selected wavelengths for new application areas.



FIGURE 66 Future work consists of two monochromatic machine vision sensors, a 50:50 beam splitter, several C-mount lenses, extension tubes, tunable filter holders and adapters between the optomechanical components. The system will have ten (10) selected band-pass filters and a software to optimise the best wavelength combination target-wisely.

YHTEENVETO (SUMMARY IN FINNISH)

Konenäköjärjestelmät tulisi suunnitella, rakentaa ja kehittää konenäön perusteiden näkökumasta kokonaisuutena. Tässä tutkimuksessa tarkastellaan konenäköjärjestelmien perustaa, joka kattaa kaikki konenäköjärjestelmien vaiheet sensoritasolta kuvantamiseen, optisen kuvan esikäsittelyyn, kuvadataan muuntamiseen, analysointiin ja tuloksiin. Konenäön perusta koostuu seuraavista termeistä: Kattotermi konenäkö kattaa kuvantamisen ja tietokonenäön kaikki vaiheet. Tietokonenäkö sisältää kuvan esikäsittely, muunnos ja analyysimenetelmät ja prosesseineen. Konenäköjärjestelmä koostuu näistä kaikista. Konenäköjärjestelmä voi olla osa autonomisia järjestelmiä, toimien esimerkiksi navigointipäätösten tuottajana, tai tuottaa analyysien avulla informaatiota päätöksenteon tueksi, esimerkiksi lääketieteellisissä soveltuksissa.

Konenäön perusteita tarkastellaan hyperspektrikuvantamisen esimerkein, jolloin kohdetta kuvataan spektrikameralla, joka erottlee optisen komponentin avulla sensorille valon eri aallonpituuksia. Hyperspektrikuvan voidaan ajatella olevan kuvakuutio, jossa valon eri aallonpituuudet ovat kuvattuna kuvakuution sisältämiin yksittäisiin kuviin samasta kohteesta, jolloin jokaisella kuvan pikselillä on spektri. Menetelmän avulla voidaan kuvata myös ihmiskäytäntöjä valon aallonpituuksia, säilyen kuitenkin turvallisilla valon sähkömagneettisen säteilyn aallonpituuksilla. Väitöskirjan teoria koostuu konenäön perusteista, alkaen komponenttisesta, kuvantamisen perusasioista, syventyneen kuvantamiseen, kuvan esikäsittelyyn, muuntamiseen ja analysoinnin teorioihin. Taustotavan metodiikan jälkeen esitellään artikkeleissa käytetyn hyperspektrikuvantamisjärjestelmän toimintalogiikka ja toteutus.

Päätuloksena esitellään kolme tehokasta muunnelmaa etäisyysperustaiseen koneoppimismenetelmään (Minimal Learning Machine (MLM)) ja optiseen biopsiaan tähtäävä ihosyöpäkuvantamisen konsepti. MLM muunnelmat soveltuват esimerkiksi poikkeamien havaitsemiseen hyperspektrikuvista ja kuvien luokittelemiseen. MLM muunnelmat ovat yksinkertaisia soveltuuen esim. reaalialkaisiin sovelluskohteisiin. Ihosyöpäkuvantamiskonseptin tulosten kautta tasotetaan tieitä optiselle biopsialle. Ihosyöpien ollessa maailman kolmanneksi yleisimpiä syöpätyypejä, optisen biopsian avulla voidaan tulevaisuudessa vähentää diagoosi- ja hoitokustannuksia, sekä aikaisen hoidon myötä jopa säästää ihmishenkia.

Konenäön perusta on tärkeä. Mikäli tutkimus ja tekeminen painottuu konenäoppimismalleihin ja konenäköön ilman kuvankaappausta, voi menetelmäkehitys biasoitua. Kehittämällä malleja datoilla, jotka on kerätty erilaisilla sensoreilla haastaa tekijäänsä, mutta toisaalta tämä tuo tutkimukseen lisää näkökulmia, ja tuloksia, jotka ovat saavutettu toisistaan riippumattomilla kuva-aineistoilla, näin ollen kehittäen kokonaisuutena alaa luotettavammin eteenpäin. Kehittämällä konenäköjärjestelmää sovelluskohteeseen mukaan voidaan vaikuttaa systeemin laskennalliseen vaativuuteen, menetelmän tuloksiin ja esimerkiksi siihen, kerääkö systeemi myös tarpeetonta dataa, jonka käsittely ja tallentaminen kuluttaa aikaa ja energiota vaikuttaen systeemin tehokkuuteen ja toimintakustannuksiin.

REFERENCES

- Adão, T., Hruška, J., Pádua, L., Bessa, J., Peres, E., Morais, R. & Sousa, J. J. 2017. Hyperspectral imaging: A review on uav-based sensors, data processing and applications for agriculture and forestry. *Remote sensing* 9 (11), 1110. doi:<https://doi.org/10.3390/rs9111110>.
- Ahmad, M., Shabbir, S., Roy, S. K., Hong, D., Wu, X., Yao, J., Khan, A. M., Mazzara, M., Distefano, S. & Chanussot, J. 2022. Hyperspectral image classification—traditional to deep models: A survey for future prospects. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 15, 968-999. doi:<https://doi.org/10.48550/arXiv.2101.06116>.
- Alleysson, D., Süsstrunk, S. & Marguier, J. 2003. Influence of spectral sensitivity functions on color demosaicing. In *Color and Imaging Conference*, Vol. 2003. Society for Imaging Science and Technology, 351–357.
- Basler 2022. Basler Group, Basler area scan cameras. <URL:<https://www.baslerweb.com/en/products/cameras/area-scan-cameras/>>. (A leading manufacturer of imaging components for computer vision applications. Accessed on 7.4.2022).
- Baumer 2022. Baumer Group, Operating principles of CMOS sensors. <URL:<https://www.baumer.com/es/en/service-support/function-principle/operating-principle-and-features-of-cmos-sensors/a/EMVA1288>>. (A leading manufacturer of sensors, encoders, measuring instruments and components for automated image-processing. Accessed on 7.4.2022).
- Biau, G. & Scornet, E. 2016. A random forest guided tour. *Test* 25 (2), 197–227. doi:<https://doi.org/10.1007/s11749-016-0481-7>.
- Bioucas-Dias, J. M., Plaza, A., Camps-Valls, G., Scheunders, P., Nasrabadi, N. M. & Chanussot, J. 2013. Hyperspectral remote sensing data analysis and future challenges. *IEEE Geoscience and Remote Sensing Magazine* 1 (2), 6–36. doi: <https://doi.org/10.1109/MGRS.2013.2244672>.
- Bishop, C. M. & Nasrabadi, N. M. 2006. *Pattern recognition and machine learning*, Vol. 4. Springer.
- Calvetti, D. & Somersalo, E. 2020. *Mathematics of Data Science: A Computational Approach to Clustering and Classification*, Vol. 1. Siam.
- Camps-Valls, G. & Bruzzone, L. 2005. Kernel-based methods for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing* 43 (6), 1351-1362. doi:[10.1109/TGRS.2005.846154](https://doi.org/10.1109/TGRS.2005.846154).

- Cortesi, D. 2022. PyInstaller, documentation. <URL:<https://pyinstaller.org/>>. (PyInstaller bundles a Python application and all its dependencies into a single package. The user can run the packaged app without installing a Python interpreter or any modules. Accessed on 9.9.2022).
- Deisenroth, M. P., Faisal, A. A. & Ong, C. S. 2020. Mathematics for machine learning. Cambridge University Press.
- Ding, S., Keal, C. A., Zhao, L. & Yu, D. 2022. Dimensionality reduction and classification for hyperspectral image based on robust supervised isomap. *Journal of Industrial and Production Engineering* 39 (1), 19–29. doi:<https://doi.org/10.1080/21681015.2021.1952657>.
- Doctor, V. 2022. Vision doctor, home page. <URL:<https://www.vision-doctor.com/en/>>. (A private, independent, non-commercial website project providing solutions to machine vision. Accessed on 4.5.2022).
- Duffner, S. & Garcia, C. 2007. An online backpropagation algorithm with validation error-based adaptive learning rate. In J. M. de Sá, L. A. Alexandre, W. Duch & D. Mandic (Eds.) *Artificial Neural Networks – ICANN 2007*. Berlin, Heidelberg: Springer Berlin Heidelberg, 249–258. doi:[10.1007/978-3-540-74690-4_26](https://doi.org/10.1007/978-3-540-74690-4_26).
- EMVA 2016. The European machine vision association, EMVA Standard 1288, Release 3.1. <URL:<https://www.emva.org/standards-technology/emva-1288/emva-standard-1288-downloads-2/>>. (Sensor and camera standards. Accessed on 8.4.2022).
- EMVA 2022. The European machine vision association, home page. <URL:<https://www.emva.org/>>. (Organisation behind sensor and camera standards. Accessed on 8.4.2022).
- Ejiri, M., Uno, T., Yoda, H., Goto, T. & Takeyasu, K. 1972. A prototype intelligent robot that assembles objects from plan drawings. *IEEE Transactions on Computers* C-21, 161-170. doi:<https://doi.org/10.1109/TC.1972.5008921>.
- Ejiri, M. 2007. Machine vision in early days: Japan's pioneering contributions. In Y. Yagi, S. B. Kang, I. S. Kweon & H. Zha (Eds.) *Computer Vision – ACCV 2007*. Berlin, Heidelberg: Springer Berlin Heidelberg, 35–53. doi:[10.1007/978-3-540-76386-4_3](https://doi.org/10.1007/978-3-540-76386-4_3).
- Eskelinan, M. 2019. Computational methods for hyperspectral imaging using Fabry-Perot interferometers and colour cameras. <URL:<http://urn.fi/URN:ISBN:978-951-39-7967-6>>.
- Esteva, A., Chou, K., Yeung, S., Naik, N., Madani, A., Mottaghi, A., Liu, Y., Topol, E., Dean, J. & Socher, R. 2021. Deep learning-enabled medical computer vision. *NPJ digital medicine* 4 (1), 1–9. doi:<https://doi.org/10.1038/s41746-020-00376-2>.

- Fei, B. 2020. Hyperspectral imaging in medical applications. In Data Handling in Science and Technology, Vol. 32. Elsevier, 523–565. doi:<https://doi.org/10.1016/B978-0-444-63977-6.00021-3>.
- Geirhos, R., Narayananappa, K., Mitzkus, B., Thieringer, T., Bethge, M., Wichmann, F. A. & Brendel, W. 2021. Partial success in closing the gap between human and machine vision. Advances in Neural Information Processing Systems 34, 23885–23899. doi:<https://doi.org/10.48550/arXiv.2106.07411>.
- GenICam 2022. The European machine vision association, GenICam standard, version 2.1.1. ⟨URL:<https://www.emva.org/standards-technology/genicam/introduction-new/>⟩.
- Ghamisi, P., Yokoya, N., Li, J., Liao, W., Liu, S., Plaza, J., Rasti, B. & Plaza, A. 2017. Advances in Hyperspectral Image and Signal Processing: A Comprehensive Overview of the State of the Art. IEEE Geoscience and Remote Sensing Magazine 5 (4), 37–78. doi:[10.1109/MGRS.2017.2762087](https://doi.org/10.1109/MGRS.2017.2762087).
- Golnabi, H. & Asadpour, A. 2007. Design and application of industrial machine vision systems. Robotics and Computer-Integrated Manufacturing 23 (6), 630-637. doi:<https://doi.org/10.1016/j.rcim.2007.02.005>. ⟨URL:<https://www.sciencedirect.com/science/article/pii/S0736584507000233>⟩.
- Gonzalez, R. C. & Woods, R. E. 2007. Digital Image Processing (3rd edition). Pearson international edition, Prentice Hall inc.
- Goodfellow, I., Bengio, Y. & Courville, A. 2016. Deep Learning. MIT Press. (<http://www.deeplearningbook.org>).
- Graña, M., Veganzons, M. & Ayerdi, B. 2022. Hyperspectral Remote Sensing Scenes. ⟨URL:http://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes⟩.
- Greivenkamp, John, E. 2004. Field Guide to Geometrical Optics, Vol. FG01. SPIE, 117. doi:<https://doi.org/10.11117/3.547461>.
- Han, Jianwei, K. & Michelini 2006. Data mining, Concepts and Techniques (Second Edition) (Second edition). Morgan Kaufman, Elsevier.
- Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., del Río, J. F., Wiebe, M., Peterson, P., Gérard-Marchant, P., Sheppard, K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C. & Oliphant, T. E. 2020. Array programming with NumPy. Nature 585 (7825), 357–362. doi:[10.1038/s41586-020-2649-2](https://doi.org/10.1038/s41586-020-2649-2). ⟨URL:<https://doi.org/10.1038/s41586-020-2649-2>⟩.
- Hughes, G. F. 1969. Comments “on the Mean Accuracy of Statistical Pattern Recognizers”. IEEE Transactions on Information Theory IT-15 (3), 420–423. doi:<https://doi.org/10.1109/TIT.1969.1054314>.

- Iqbal, Z., Khan, M. A., Sharif, M., Shah, J. H., ur Rehman, M. H. & Javed, K. 2018. An automated detection and classification of citrus plant diseases using image processing techniques: A review. *Computers and electronics in agriculture* 153, 12–32. doi:<https://doi.org/10.1016/j.compag.2018.07.032>.
- JYU 2022. University of Jyväskylä, Spectral Imaging Laboratory. <URL:<https://www.jyu.fi/it/en/research/our-laboratories/hsi>>. (Accessed on 9.4.2022).
- Jia, Y., Abbott, J. T., Austerweil, J. L., Griffiths, T. & Darrell, T. 2013. Visual concept learning: Combining machine vision and bayesian generalization on concept hierarchies. *Advances in Neural Information Processing Systems* 26.
- Jääskeläinen, S., Eskelinen, M., Annala, L. & Raita-Hakola, A.-M. 2019a. Camazing Python library. <URL:<https://pypi.org/project/camazing/>>. (Machine vision library for GenICam-compliant cameras. Developed at the University of Jyväskylä, Spectral Imaging Laboratory. Released under MIT-licence. Accessed on 9.4.2022).
- Jääskeläinen, S., Eskelinen, M., Annala, L. & Raita-Hakola, A.-M. 2019b. Spectacular Python library. <URL:<https://www.jyu.fi/it/en/research/our-laboratories/hsi>>. (Available on demand. Accessed on 4.5.2022).
- Kingma, D. P. & Ba, J. 2014. Adam: A Method for Stochastic Optimization. doi:<https://doi.org/10.48550/ARXIV.1412.6980>. <URL:<https://arxiv.org/abs/1412.6980>>.
- Koppal, S. J. (Ed.) 2014. Computer Vision: A Reference Guide. Boston, MA: Springer US, 441–443. doi:10.1007/978-0-387-31439-6_534. <URL:https://doi.org/10.1007/978-0-387-31439-6_534>.
- Krupnik, D. & Khan, S. 2019. Close-range, ground-based hyperspectral imaging for mining applications at various scales: Review and case studies. *Earth-science reviews* 198, 102952. doi:<https://doi.org/10.1016/j.earscirev.2019.102952>.
- Li, Q., He, X., Wang, Y., Liu, H., Xu, D. & Guo, F. 2013. Review of spectral imaging technology in biomedical engineering: achievements and challenges. *Journal of biomedical optics* 18 (10), 100901. doi:<https://doi.org/10.1117/1.JBO.18.10.100901>.
- Lind, L., Laamanen, H. & Pölönen, I. 2021. Hyperspectral imaging of asteroids using an fpi-based sensor. In *Sensors, Systems, and Next-Generation Satellites XXV*, Vol. 11858. SPIE, 65–78. doi:<https://doi.org/10.1117/12.2599514>.
- Lindholm, V., Annala, L., Raita-Hakola, A.-M., Jeskanen, L., Koskenmies, S., Pitkänen, S., Isoherranen, K., Pölönen, Ilkka Salmivuori, M. & Annamari, R. 2022. Discriminating basal cell carcinoma and bowen's disease from benign lesions with a 3d hyperspectral imaging system and convolutional neural networks. *JAAD international Status: (Submitted)*.

- Makki, I., Younes, R., Francis, C., Bianchi, T. & Zucchetti, M. 2017. A survey of landmine detection using hyperspectral imaging. *ISPRS Journal of Photogrammetry and Remote Sensing* 124, 40–53. doi:<https://doi.org/10.1016/j.isprsjprs.2016.12.009>.
- Manolakis Dimitris, L. R. & Thomas, C. 2016. Hyperspectral Imaging and Remote Sensing, physics, sensors and algorithms, Vol. 1. Cambridge University Press.
- Marx, M. & Fuegi, A. 2021. Reliance on science by inventors: Hybrid extraction of in-text patent-to-article citations, 1-24. doi:<https://doi.org/10.1111/jems.12455>. (URL:<https://onlinelibrary.wiley.com/doi/abs/10.1111/jems.12455>).
- Meribout, M., Baobaid, A., Khaoua, M. O., Tiwari, V. K. & Pena, J. P. 2022. State of art iot and edge embedded systems for real-time machine vision applications. *IEEE Access* 10, 58287–58301. doi:<https://doi.org/10.1109/ACCESS.2022.3175496>.
- Mesquita, D. P., Gomes, J. P. & Souza Junior, A. H. 2017. Ensemble of Efficient Minimal Learning Machines for Classification and Regression. *Neural Processing Letters* 46 (3), 751–766. doi:<https://doi.org/10.1007/s11063-017-9587-5>.
- Mirbod, O., Choi, D., Thomas, R. & He, L. 2021. Overcurrent-driven leds for consistent image colour and brightness in agricultural machine vision applications. *Computers and Electronics in Agriculture* 187, 106266. doi:<https://doi.org/10.1016/j.compag.2021.106266>.
- Model, I. & Shamir, L. 2015. Comparison of data set bias in object recognition benchmarks. *IEEE Access* 3 (1), 1953–1962. doi:<https://doi.org/10.1109/ACCESS.2015.2491921>.
- Noble, W. S. 2006. What is a support vector machine? *Nature biotechnology* 24 (12), 1565–1567. doi:<https://doi.org/10.1038/nbt1206-1565>.
- O'Mahony, N., Campbell, S., Carvalho, A., Harapanahalli, S., Hernandez, G. V., Krpalkova, L., Riordan, D. & Walsh, J. 2019. Deep learning vs. traditional computer vision. In *Science and information conference*. Springer, 128–144. doi:https://doi.org/10.1007/978-3-030-17795-9_10.
- Pathmanaban, P., Gnanavel, B. K. & Anandan, S. S. 2019. Recent application of imaging techniques for fruit quality assessment. *Trends in Food Science and Technology* 94 (October), 32–42. doi:[10.1016/j.tifs.2019.10.004](https://doi.org/10.1016/j.tifs.2019.10.004). (URL:<https://doi.org/10.1016/j.tifs.2019.10.004>).
- Paullada, A., Raji, I. D., Bender, E. M., Denton, E. & Hanna, A. 2021. Data and its (dis)contents: A survey of dataset development and use in machine learning research. *Patterns* 2 (11), 100336. doi:<https://doi.org/10.1016/j.patter.2021.100336>.

- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M. & Duchesnay, E. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* 12, 2825–2830. doi:<https://doi.org/10.48550/arXiv.1201.0490>.
- Perot, A. & Fabry, C. 1899. On the application of interference phenomena to the solution of various problems of spectroscopy and metrology. *The Astrophysical Journal* 9, 87. doi:<https://doi.org/10.1086/140557>.
- Planche, B. & Andres, E. 2019. Hands-On Computer Vision with TensorFlow 2: Leverage deep learning to create powerful image processing apps with TensorFlow 2.0 and Keras. Packt Publishing Ltd.
- Pölönen, I., Rahkonen, S., Annala, L. & Neittaanmäki, N. 2019. Convolutional neural networks in skin cancer detection using spatial and spectral domain. In *Photonics in Dermatology and Plastic Surgery 2019*, Vol. 10851. International Society for Optics and Photonics, 108510B. doi:<https://doi.org/10.1117/12.2509871>.
- Pölönen, I., Riihiaho, K., Hakola, A.-M. & Annala, L. 2020. Minimal learning machine in anomaly detection from hyperspectral images. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives* 43 (B3), 467–472. doi:<https://doi.org/10.5194/isprs-archives-XLIII-B3-2020-467-2020>.
- Rahkonen, S., Lind, L., Raita-Hakola, A.-M., Kiiskinen, S. & Pölönen, I. 2022. Reflectance measurement method based on sensor fusion of frame-based hyperspectral imager and time-of-flight depth camera. *Sensors* 22 (22). doi:<https://doi.org/10.3390/s22228668>. ⟨URL:<https://www.mdpi.com/1424-8220/22/22/8668>⟩.
- Raita-Hakola 2022. Minimal learning machine functions. ⟨URL:<https://gitlab.jyu.fi/hsj/MLM>⟩. (Minimal Learning Machine functions will be released (2023) in GitLab repository, which is maintained by the author at the Spectral imaging laboratory, University of Jyväskylä. Accessed on 16.11.2022).
- Ren, Z., Fang, F., Yan, N. & Wu, Y. 2021. State of the art in defect detection based on machine vision. *International Journal of Precision Engineering and Manufacturing-Green Technology* 9 (-), 1–31. doi:<https://doi.org/10.1007/s40684-021-00343-6>.
- Riihiaho, K. A., Eskelinen, M. A. & Pölönen, I. 2021. A do-it-yourself hyperspectral imager brought to practice with open-source python. *Sensors* 21 (4). doi:<https://doi.org/10.3390/s21041072>. ⟨URL:<https://www.mdpi.com/1424-8220/21/4/1072>⟩.

- Riverbank, c. 2019. Riverbank computing, PyQt5 documentation. <URL:<https://www.riverbankcomputing.com/>>. (PyQt brings together the Qt C++ cross-platform application framework and the cross-platform interpreted language Python. Accessed on 9.9.2022).
- Romberg, J. 2016. ECE 6250 Fall 2016 Lecture notes. <URL:<https://cpb-us-w2.wpmucdn.com/sites.gatech.edu/dist/2/436/files/2017/07/22-notes-6250-f16.pdf>>. (Lecture notes available from Georgia Tech University).
- Saari, H., Pölönen, I., Salo, H., Honkavaara, E., Hakala, T., Holmlund, C., Mäkinen, J., Mannila, R., Antila, T. & Akujärvi, A. 2013. Miniaturized hyperspectral imager calibration and UAV flight campaigns. Sensors, Systems, and Next-Generation Satellites XVII 8889 (October 2013), 88891O. doi:<https://doi.org/10.1117/12.2028972>.
- Sadeghipoor, Z., Lu, Y. M. & Süsstrunk, S. 2012. Optimum spectral sensitivity functions for single sensor color imaging. In Digital photography VIII, Vol. 8299. SPIE, 26–39. doi:<https://doi.org/10.1117/12.907904>.
- Salmi, P., Calderini, M., Pääkkönen, S., Taipale, S. & Pölönen, I. 2022. Assessment of microalgae species, biomass, and distribution from spectral images using a convolution neural network. Journal of Applied Phycology 34, 1–11. doi:<https://doi.org/10.1007/s10811-022-02735-w>.
- Smith, M. L., Smith, L. N. & Hansen, M. F. 2021. The quiet revolution in machine vision - a state-of-the-art survey paper, including historical review, perspectives, and future directions. Computers in Industry 130, 103472. doi:<https://doi.org/10.1016/j.compind.2021.103472>.
- de Souza, A. H., Corona, F., Barreto, G. A., Miche, Y. & Lendasse, A. 2015. Minimal Learning Machine: A novel supervised distance-based approach for regression and classification. Neurocomputing 164 (January), 34–44. doi:<https://doi.org/10.1016/j.neucom.2014.11.073>.
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. & Salakhutdinov, R. 2014. Dropout: A simple way to prevent neural networks from overfitting. Journal of Machine Learning Research 15 (56), 1929–1958. doi:<https://doi.org/10.5555/2627435.2670313>. <URL:<http://jmlr.org/papers/v15/srivastava14a.html>>.
- Stemmer 2022. Stemmer Imaging, The Imaging and Vision Handbook. <URL:<https://www.stemmer-imaging.com/en/the-imaging-vision-handbook/>>. (A leading international machine vision technology provider. Accessed on 7.4.2022).
- Takemura, Y. 2019. The development of video-camera technologies: Many innovations behind video cameras are used for digital cameras and smartphones. IEEE Consumer Electronics Magazine 8 (4), 10–16. doi:<https://doi.org/10.1109/MCE.2019.2905482>.

- Thorp, K. R., Wang, G., Bronson, K. F., Badaruddin, M. & Mon, J. 2017. Hyperspectral data mining to identify relevant canopy spectral features for estimating durum wheat growth, nitrogen status, and grain yield. *Computers and Electronics in Agriculture* 136, 1–12. doi:10.1016/j.compag.2017.02.024. <URL:<http://dx.doi.org/10.1016/j.compag.2017.02.024>>.
- Tian, Z., Ma, W., Yang, Q. & Duan, F. 2021. Application status and challenges of machine vision in plant factory—a review. *Information Processing in Agriculture* 9. doi:<https://doi.org/10.1016/j.inpa.2021.06.003>.
- Uma, J. & Sukanya, B. 2021. A survey on recent trends in machine vision. SSRN, Proceedings of the Yukthi 2021- The International Conference on Emerging Trends in Engineering – GEC Kozhikode, Kerala, India -. doi:<https://doi.org/10.2139/ssrn.3972999>. <URL:<https://ssrn.com/abstract=397299>>.
- Unidata 2022. Unidata netcdf4-python documentation. <URL:<https://unidata.github.io/netcdf4-python/>>. (Unidata, Data services and tools Geoscience. Unidata is a diverse community of education and research institutions with the common goal of sharing geoscience data and the tools to access and visualize that data. netcdf4-python is a Python interface to the netCDF C library. Accessed on 9.9.2022).
- Usama, M., Piatetsky-Shapiro, G., Smyth, P. & R, U. 1996. Advances in knowledge discovery and data mining. AAAI Press.
- Wang, Y. W., Reder, N. P., Kang, S., Glaser, A. K. & Liu, J. T. 2017. Multiplexed optical imaging of tumor-directed nanoparticles: a review of imaging systems and approaches. *Nanotheranostics* 1 (4), 369. doi:<https://doi.org/10.7150/ntno.21136>.

ORIGINAL PAPERS

PI

MINIMAL LEARNING MACHINE IN HYPERSPECTRAL IMAGING CLASSIFICATION

by

Hakola Anna-Maria and Pölönen Ilkka 2020

SPIE - Society of Photo-Optical Instrumentation Engineers, Proceedings Volume
11533, Image and Signal Processing for Remote Sensing XXVI; 115330R,
<https://doi.org/10.1117/12.2573578>

Reproduced with kind permission of SPIE.

Minimal learning machine in hyperspectral imaging classification

Anna-Maria Hakola^a and Ilkka Pölönen^a

^aFaculty of Information Technology, University of Jyväskylä, 40100, Jyväskylä, Finland

ABSTRACT

A hyperspectral (HS) image is typically a stack of frames, where each frame represents the intensity of a different wavelength of light. Each spatial pixel has a spectrum. In the classification of the HS image, each spectrum is classified pixel-by-pixel. In some of the real-time applications, the amount of the HS image data causes performance challenges. Those issues relate to the platforms (e.g. drones) payload restrictions, the issues of the available energy and to the complexity of the machine learning models.

In this study, we introduce the minimal learning machine (MLM) as a computationally cheap training and classification machine learning method for the hyperspectral imaging classification. MLM is a distance-based method that utilizes mapping between input and output distances. Input distance is a distance between the training set and its subset R . Output distance is corresponding distances between the label values of the training set and the subset R . We propose a training point selection framework, which reduces the number of data points in the R by selecting the points class-by-class, in the direction of the principal components of each class.

We test MLM's performance against four other classification machine learning methods: Random Forest, Artificial Neural Network, Support Vector Machine and Nearest Neighbours classifier with three known hyperspectral data sets. As the main outcomes, we will show how the performance is affected by the size of the subset R . We compare our subset selection method MLM's performance to the random selection MLM's performance. Results show that MLM is an computationally efficient way to train large training sets. MLM reduces the complexity of the analysis and provides computational benefits against other models. Proposed framework offers tools that can improve the MLM's classification time and the accuracy rate compared to the MLM with randomly picked training points.

Keywords: Hyperspectral Imaging, Minimal Learning Machine, Classification, Principal Component Analysis, Distance Learning

1. INTRODUCTION

Hyperspectral (HS) images contains information that allows the characterization, identification and classification of the targets, such as land-covers with improved accuracy and robustness.¹ Since the technical evolution of optical sensors² has been improving the imagers, there has been several new application domains, for example in the medical diagnosis,³ skin cancer research,⁴ forest industry⁵ and agricultural applications³.

Hyperspectral image classification is a process, where single pixels are assigned into a set of classes². Classification approaches can be split into categories of *supervised*, *unsupervised* and *semi-supervised classifiers*⁶. Supervised methods classifies based on model, which is created with training samples and their labels. Unsupervised classifiers are using clustering methods without labelled training samples and semi-supervised classifiers are using both, labelled and unlabelled training samples^{2,6}.

HS images can be classified pixel-wise with *spectral classifiers* or spectral-spatial with *spectral-spatial classifiers*⁷. The spectral classifier considers the HS images as a list of spectral information, while the spectral-spatial-classifier uses both, the spectral and the spatial information⁶.

Further author information: (Send correspondence to Anna-Maria Hakola)

Anna-Maria Hakola.: E-mail: anna.m.hakola@jyu.fi, Telephone: +358 (0)40 54 92 12 0

Ilkka Pölönen : E-mail: ilkka.polonen@jyu.fi, Telephone: +358 (0)400 248 140

The HS image classification can be a challenge². For example the accuracy results that can be reached with standard classifiers and multispectral images are typically compromised with HS images¹. The typical challenges of the spectral classifiers are the relatively small size of the training set from the high-dimensional data,⁷ the high number of spectral channels,¹ the spatial variability of the spectral signature¹ and the quality of the spectral data. As an example, the challenges are related to the Hughes phenomenon,⁸ the conditions, such as incident illumination or instrument noises⁷ or to the high cost of the true sample labelling¹.

Despite the challenges, there are many classic classification methods that can perform well with HS images^{1,3,6}. Some of the popular machine learning classification approaches utilises the neural networks, support vector machines, random forests or deep learning classification methods⁶. Because those methods can be complex and time-consuming, it is an interesting idea to introduce HS images to the relatively new classifier, which is an easy to implement and has had a promising results on performance and accuracy⁹.

*Minimal learning machine*⁹ (MLM) is a supervised, distance-based machine learning classification method that utilizes mapping between input and output distances. The input distance is a distance between the training set and its subset R . The output distance is corresponding distances between the label values of the training set X and subset R . The MLM classification model is a generalization of a nearest neighbour classifier⁹. This approach requires argument sorting for the distances between the input and output values and assigning of the closest label value from the ground truth labels.

Previous studies confirms, that one of the main advantages of the basic MLM is that there is only one parameter that requires tuning⁹. The parameter is the number of the training samples (R). The closer the amount of the selected training points (R) and the size of the entire training data X goes, the more accurate are the results. That might encourage to increase the amount of the selected training points, but it will also increase the computational efficiency and the used time.

When we are presenting the HS images to a supervised spectral MLM classifier, we need to pay attention to a few features of the HS images. HS image consists of a large amount of spectral bands of which are the dimensions of the spectral data. The high number of dimensions and the amount of the data makes the processing computationally and memory costly. Other challenges are the curse of the dimensionality^{8,10} and the redundancy among the samples¹¹. Those challenges might have an impact to the classifiers performance and accuracy.

Dimensionality reduction methods offers one solution for those challenges. One of the most widely used method¹¹ is the Principal Component Analysis (PCA) with its different extensions. With PCA, the data is projected with the orthogonal projections of which maximises the variance of the data. The data is yielded to a new uncorrelated coordinate system¹². With PCA, we can reduce the dimensions and select the training points intentionally for the classifier.

As a main results in this study, we propose a new approach which can increase the accuracy rate and training time, and reduce the classification time of the MLM classifier. The proposed framework (subsection 2.3) focuses on the selection of the training output samples, and it consists from the data point selection algorithm, the MLM training and the MLM classification algorithms.

The data point selection algorithm 1 utilises the PCA and selects only 3 data points from each of the classes in the direction of each used principal components. The aim is to have a collection of data points which represents the geometry of each class. This approach differs from the original MLM approach of picking the training points randomly⁹. The objective is to minimize the size of R , as it is the most influential factor⁹ in the computational complexity of this method, without reducing the accuracy with reduced training points. The new framework is called the PC-MLM.

The proposed framework uses three hyper-parameters. The amount of principal components (PC), the number of neighbours and optionally the distance metrics (Euclidean, Manhattan or Cosine). Those hyper-parameters controls the size of the subset R . The focus of the new framework is to improve the performance and still maintain the accuracy rate in an acceptable level.

We will test the PC-MLM framework against four other supervised classification machine learning methods: Random Forest, Artificial Neural Network, Support Vector Machine and k-Nearest Neighbours (kNN) classifier with three known hyperspectral data sets (Indian Pines, Salinas and Pavia City). As the main outcomes, we

show how the performance is affected by the size of the subset R . Our hypothesis is that the MLM is the fastest model to train when the size of the training set is large. MLM reduces the complexity of the analysis and provides computational benefits against other models. PC-MLM framework offers tools that can improve the accuracy rate and classification time compared to the MLM with randomly picked training points with large data sets.

The content of the paper is organized as follows. The section 2 describes methods, PC-MLM frameworks algorithms and the demonstration materials. The results are introduced in the section 3. The analysis of the results are discussed into the section (4), and the final section 5 concludes the study.

2. MATERIAL AND METHODS

On this section, we will present the MLM extension to the hyperspectral imaging. We explain the idea of the intentionally selected training points (PC-MLM) and we will introduce the framework with step-by-step algorithms. Finally there are a short overview to the comparison methods and the introduction of the selected HS images as an demonstration material.

2.1 Minimal learning machine

MLM is an supervised machine learning algorithm that can be extended to classification tasks⁹. MLM's learning process has three phases¹³. It consists of building a linear mapping between the matrices of input and output distances. Input distance is a distance between the training set and its subset R . Output distance is corresponding distances between the label values of the training set and subset R . The generalization phase utilises the learned mapping and provides an estimation of the distances between output values and the target output value. The third phase is an optimization problem, based on the predicted output distances and the ground truth points. The third phase is the computationally most complex⁹.

In our framework, the basic MLM model is introduced to hyperspectral image classification. The aim is to reduce the complexity of the optimization by selecting three training points from the directions of principal components of the data set. Attempt is to mimics data sets geometry.

In the case of HS images, the training set of spectra with d wavebands is $\mathbf{x}_i \in X \subset \mathbb{R}^d$ and $\mathbf{m}_k \in R$ is the intentionally selected sampled subset of the X . Correspondingly $\mathbf{y}_i \in Y \subset \mathbb{R}$ are the labels of the training set and $\mathbf{t}_k \in T$ are the subset of the Y . The training set X consist of N samples, and subset R has K samples. Now $d(\mathbf{x}_i, \mathbf{m}_k)$ and $\delta(\mathbf{y}_i, \mathbf{t}_k)$ are the linear mapping distances.

After selecting the samples, we can define two matrices based on these distances $\Delta_y \in \mathbb{R}^{N \times K}$ and $\mathbf{D}_x \in \mathbb{R}^{N \times K}$. By assuming the linear mapping between these two distance matrices, we have a linear model

$$\Delta_y = \mathbf{D}_x \mathbf{B} + \mathbf{E}, \quad (1)$$

where \mathbf{B} is coefficients and \mathbf{E} is the residual. Coefficients \mathbf{B} can be approximated using the ordinary least squares estimator¹³

$$\widehat{\mathbf{B}} = (\mathbf{D}_x^T \mathbf{D}_x)^{-1} \mathbf{D}_x^T \Delta_y. \quad (2)$$

Now $\widehat{\mathbf{B}}$ is a linear model between distances $\delta(\mathbf{y}_i, \mathbf{t}_k)$ and $d(\mathbf{x}_i, \mathbf{m}_k)$. Now, for the new spectrum \mathbf{x}_n the distance between its label y_n and set T is

$$\delta(y_n, T) = d(\mathbf{x}_n, R) \widehat{\mathbf{B}}. \quad (3)$$

Label y_n can be estimated by solving an quadratic optimisation problem

$$\min_{y_n} \sum_{k=1}^K (y_n - \mathbf{t}_k)^T (y_n - \mathbf{t}_k) - \delta^2(y_n, T). \quad (4)$$

The equation 4 is the bottle neck of the MLM¹³. If we solve it with optimization methods, our classification is time-consuming and computationally expensive. We can consider that the model $\widehat{\mathbf{B}}$ is a of generalization

of the nearest neighbour classifier. Equation 3 gives us distances to the nearest label values of \mathbf{x}_n . For the classification of the \mathbf{x}_n , we need to perform argument sorting for $\delta(\mathbf{y}_n, T)$ and assign the closest label value from T . By selecting k closest values and use majority voting of labels; we have similar results than with kNN. The detailed implementation is shown in algorithms 2 and 3.

In MLM, the subset R is selected picking data points randomly⁹. When the size of R approaches the size of X , the accuracy of the results improves⁹. The objective is to minimize the size of R , as it is the most influential factor in the computational complexity of this method. In our framework, we utilise previous MLM steps with the idea of using principal components to select the data points intentionally to represent the shape of the selected component.

2.2 The selection of the reference points

The Principal Component Analysis (PCA), finds the directions of the data variance and projects the data according to the variance orthogonally, in as many directions as we have dimensions in the data¹². The PCA reveals the geometry of the data; the first PC component has the highest variation of the data, the second component reveals the highest variance on orthogonal direction of the first component and so on.

On this study, we utilise the properties of the PCA, by using the component directions to select intentionally the data points from training set X to the reference set R . As a result, the R represents the geometry of each class of the HS image classification targets. We performed this separately for each class of the training set X . We selected the minimum, maximum and median positions from each of the selected principal components (PC) directions. Because of the noise in the data sets, the minimum and maximum positions were tuned by moving both of the extreme values 5% towards the median position. By using this strategy, we can significantly reduce the size of the reference set R .

On figures 1, 2 and 3 the training data from the each of the HS images is visualised class-by-class after the PCA. The figures shows the first and the second principal components on the X and Y axis. The selected reference points are marked to the figures with colors. Black represents the first direction, red is the second and yellow is the direction of the third principal component. The details of the ground truth classes can be seen on the subsection 2.4, tables 2 and 1.

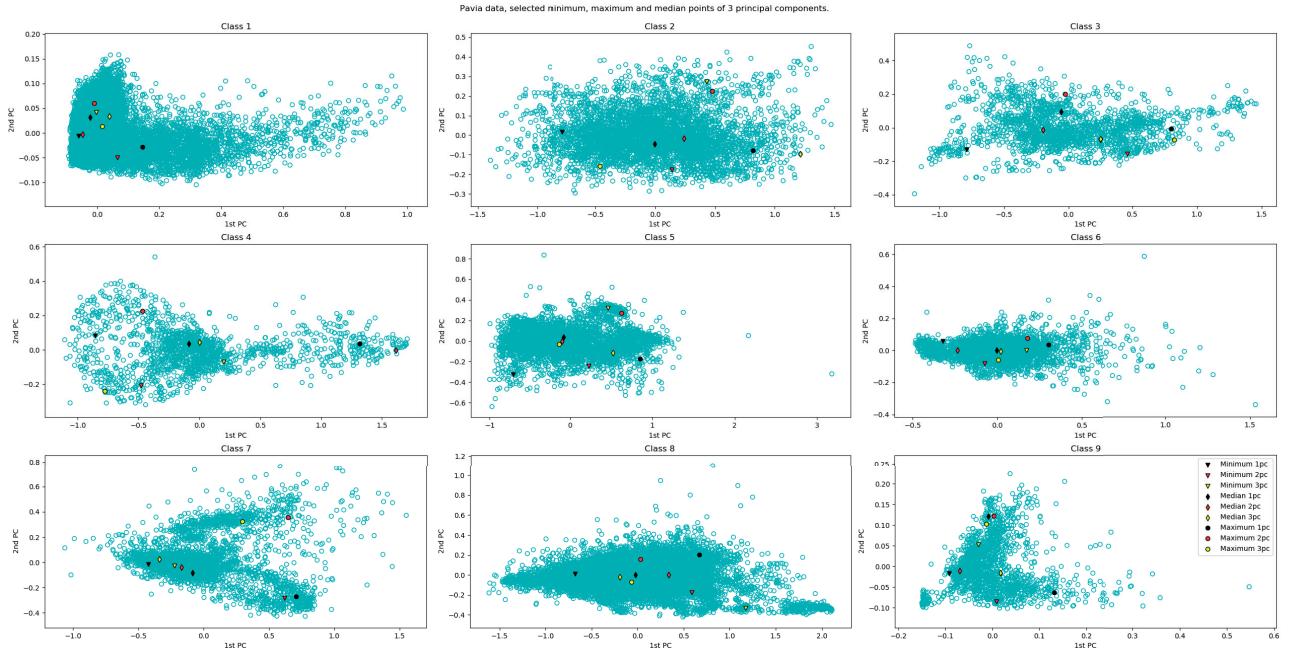


Figure 1. Pavia Centre HS image. The class-by-class visualisation of the first and second principal components. The selected data points from the first three principal components of the class, of which represents the geometry of the each class are marked with colors. The total amount of selected data points with three principal components was 81.

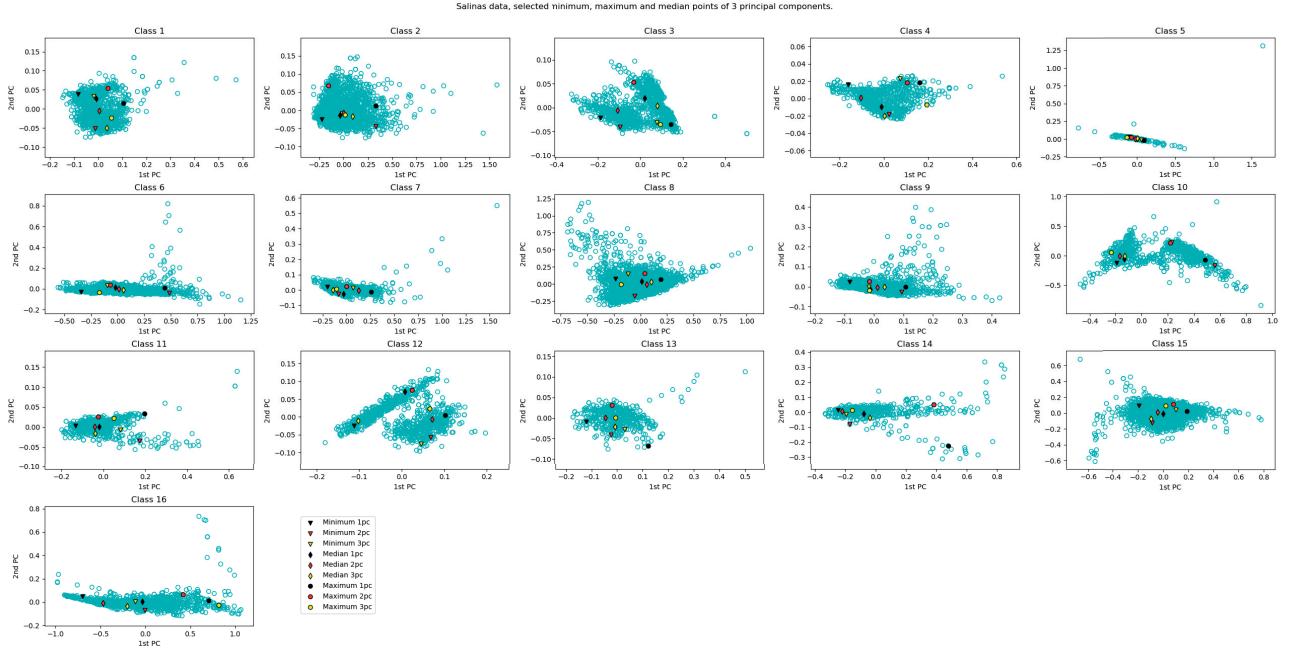


Figure 2. Salinas HS image. The class-by-class visualisation of the first and second principal components. The selected data points from the first three principal components of the class, of which represents the geometry of the each class are marked with colors. The total amount of selected data points with three principal components was 144.

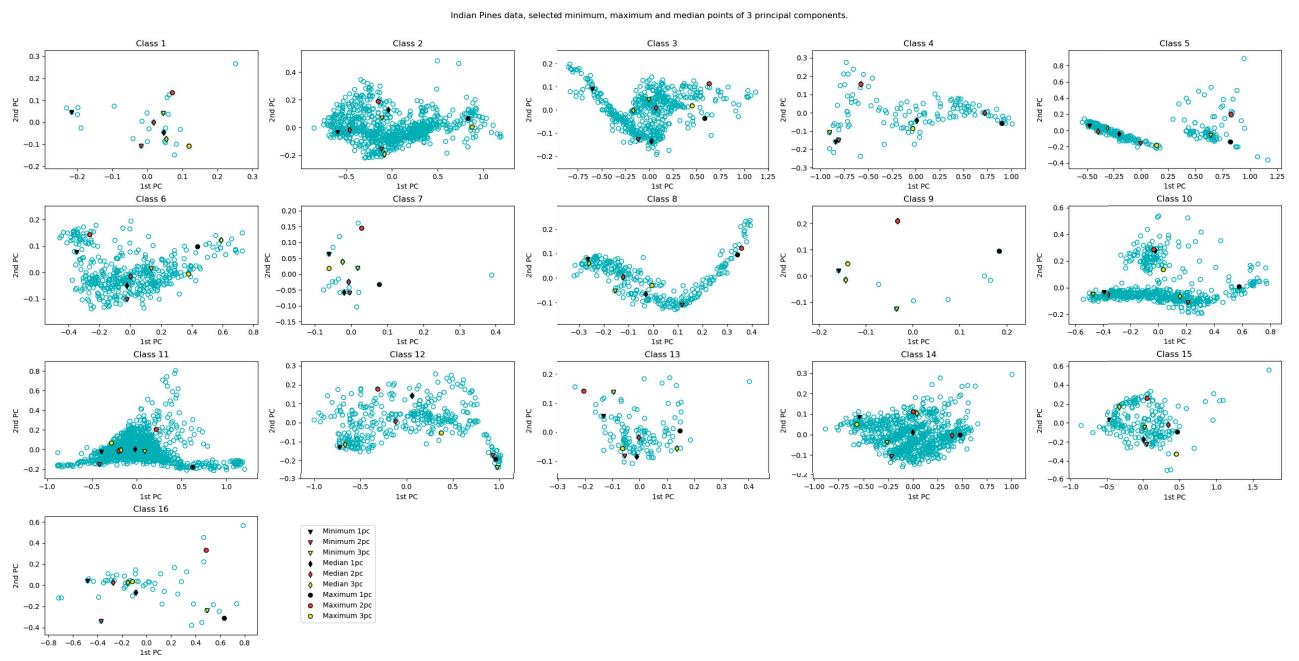


Figure 3. Indian Pines HS image. The class-by-class visualisation of the first and second principal components. The selected data points from the first three principal components of the class, of which represents the geometry of the each class are marked with colors. The total amount of selected data points with three principal components was 144.

2.3 The PC-MLM framework

The proposed framework consists of three algorithms. At first, the randomised training arrays are introduced to the PC algorithm 1. The training phase starts from the reference points selection algorithm 1 and ends to the first MLL algorithm 2. Prediction is the last step of the framework, the implementation can be seen on algorithm 3. The training time of this study is calculated from performing the algorithms 1 and 2, and the classification time is the time used on performing the algorithm 3. The framework was implemented in Python. The PCA, accuracy rate, distance metrics were implemented with Scikit-learn¹⁴ metrics and decomposition methods.

Algorithm 1: The selection of the reference points

Input: Training data, the number of components

Step 1: Perform PCA class-by-class

For each class

 Compute number of components with PCA from the training data

Step 2: Select training points from each component

For each component

 Argument sort component

 Select median point from sorted score

 Select subtracted minimum from sorted score

 Select subtracted maximum from sorted score

Step 3: store selected points

 Append selected points to set R and their labels to set T

Result: The R and T are now sorted arrays that represents the data sets shape in the direction of the selected principal components. The R and T are ready for the MLM classifier implementation

Algorithm 2: The MLM training phase

Input: X (Training data), Y (Labels of the X), R (Selected reference points), T (Labels of the R), distance metric

Step 1: Calculate output distances

Use Euclidean method and calculate the distances between Y and T .

Step 2 : Calculate input distances

Use distance metric and calculate the distances between X and R

Step 3: Approximate the coefficients

Use the ordinary least squares estimator (equation 2), and mirror the input distances to output distances

Output: Trained model $\hat{\mathbf{B}}$

Algorithm 3: The MLM prediction phase

Input: New data, T , R , $\hat{\mathbf{B}}$, distance metric

Step 1: Calculate new distances

Calculate new_distances, use selected metrics and calculate distances between New data and R

Step 2: Solve the equation 3

$$\delta(y_n, T) = \text{new_distances}.dot(\hat{\mathbf{B}})$$

perform an argument sort to $\delta(y_n, T)$

Step 3: Select labels

if *number of neighbours* != 1 **then**

| select neighbours (n shortest distances from sorted $\delta(y_n, T)$)

| The result label is the mode of the neighbours

else

| select the nearest neighbour from the T , use the indexes of the sorted $\delta(y_n, T)$

end

Result: The data is classified with PC-MLM framework.

2.4 Spectral images and preprocessing

The experiments were done with three known hyperspectral images, downloaded from the Grupo De Intelligencia Computational (GIC)¹⁵. The Pavia Centre HS image is the largest one of these three data sets. It has been acquired by the ROSIS sensor with 102 spectral bands with the geometric resolution of 1.3 meters and with a spectrum coverage ranging from 430 to 860 nm¹⁶. The image size is 1096 x 1096 pixels. The image ground truth (table 1) is divided into 9 classes¹⁵. The geometry of Pavia HS image's each class can be seen on figure 1.

The Salinas scene is collected with AVIRIS sensor. It has 224 spectral bands and the labels are divided in to 16 classes (table 2). The size of the data set is 512 x 217 pixels with 3.7m spatial resolution over the range of 400–2500 nm¹⁷. Salinas data set is preprocessed, 20 water absorption bands has been removed from the data¹⁵. The Salinas HS image's class-by-class geometry can be seen on figure 2.

Third HS image is the smallest one, Indian Pines. It consists of 145 x 145 pixels and 224 spectral reflectance bands in the wavelength range of 400 - 2500 nm. It has been captured with AVIRIS sensor over the Indian Pines test site. The ground truth (table 2) has been divided in to 16 classes that represents agriculture, forest and other natural perennial vegetation. 20 Water absorption bands has been removed, leaving the total amount of bands to 200. Indian Pines data set differs from the other data sets with its ground truth coverage. Some of the ground truth classes has only 5% coverage.¹⁵ The geometry of Indian Pines classes can be seen on figure 3.

Table 1. Pavia Centre HS image ground truth classes and number of the samples

#	Class	Samples
1	Water	824
2	Trees	820
3	Asphalt	816
4	Self-Blocking Bricks	808
5	Bitumen	808
6	Tiles	1260
7	Shadows	476
8	Meadows	824
9	Bare Soil	820

Table 2. Salinas and Indian Pines HS image, ground truth classes and number of the samples

#	Salinas		#	Indian Pines	
#	Class	Samples	#	Class	Samples
1	Brocoli green weeds 1	2009	1	Alfalfa	46
2	Brocoli green weeds 2	3726	2	Corn-notill	1428
3	Fallow	1976	3	Corn-mintill	830
4	Fallow rough plow	1394	4	Corn	237
5	Fallow smooth	2678	5	Grass-pasture	483
6	Stubble	3959	6	Grass-trees	730
7	Celery	3579	7	Grass-pasture-mowed	28
8	Grapes untrained	11271	8	Hay-windrowed	478
9	Soil vineyard develop	6203	9	Oats	20
10	Corn senesced green weeds	3278	10	Soybean-notill	972
11	Lettuce romaine 4wk	1068	11	Soybean-mintill	2455
12	Lettuce romaine 5wk	1927	12	Soybean-clean	593
13	Lettuce romaine 6wk	916	13	Wheat	205
14	Lettuce romaine 7wk	1070	14	Woods	1265
15	Vinyard untrained	7268	15	Buildings-Grass-Trees-Drives	386
16	Vinyard vertical trellis	1807	16	Stone-Steel-Towers	93

All of the three HS images were preprocessed as follows. At first, the spectral data was scaled between 0 and 1. The spectral data and the ground truth data were converted and reshaped to two-dimensional arrays. On the array, each spatial pixel has its values, which is the pixel spectrum. On second, the spectral data and the ground truth data were randomised with random permutation method, and all non-classified values were removed from the both of the data sets. The randomised spectral data and its ground truth were divided to training (60%), validating (20%) and testing (20%) portions. The framework's model was trained with the training data, performance and parameters were validated with validation data and the results were achieved with the test data.

2.5 Reference methods

We tested PC-MLM against four other classification machine learning methods: Random Forest (RF), Artificial Neural Network (ANN), Support Vector Machine (SVM) and k-Nearest Neighbours (kNN).

SVM is a supervised machine learning model which is based on the theory of statistical learning¹⁸. The basic idea is to separate the classes with hyperplane in a high or infinite dimensional space¹⁴. Hyperplane is the decision surface of which the classifier uses to make the decisions on the classification phase. The hyperplane is optimal, when the margin between the nearest positive and nearest negative training sample is maximized.¹⁸

ANN methods are widely used in image analysis. Basic model of ANN consists of the input, hidden and output layers¹⁴. The variables in the input layer are called nodes, in our case the nodes are the training set samples. The nodes on the output layer represents the values of the output classes. There are weighted links between the layers, which controls the flow from input layer thru the hidden layers finally to the output layer.¹⁸

RF classifier is an ensemble learning algorithm of which are more accurate and robust towards noise than single classifiers^{19,20}. The original RF classifier consists of a group of a single voting classifiers. Each classifier votes for the assignation of the most frequent class for the input vector.²⁰ On this study, we implemented the RF classifier from the Scikit-learn Python library, of which combines the classifiers by averaging their probabilistic prediction, which differs from the Breimans original voting strategy^{14,20,21}. The fourth reference method, the Nearest Neighbours classifier (kNN) was used with reference points R . We used the Scikit-learn Nearest neighbour classifier implementation¹⁴ to predict the classes.

The implementations of all the classifiers were from the Scikit-learn Python library¹⁴. SVM, ANN and RF were optimized with two rounds of hyper-parameter search with the largest data set Pavia Centre. At the first phase, we narrowed down the selection of possible hyper-parameter values with Scikit-learn randomized Search CV method¹⁴. On the second phase, based on the results of the first search, we narrowed down the values and run the Scikit-learn's Grid Search CV method¹⁴. As a result, we found the hyper-parameters of the best accuracy rate for each classifier. The kNN classifier was tested with the same number of neighbours than in our PC-MLM classifier.

All computations were done using Dell laptop with Intel Core i5-7300U CPU 2,6 GHz processor and 8 GB memory.

3. RESULTS

The first part of the section presents the results of a comparison of the distance metrics of the MLM and PC-MLM classifiers. Second, we present a classifier comparison with figures and tables, using the distance metrics selected at the beginning of the chapter. Finally, we will see how the PC-MLM framework can improve the MLM classifier performance on large data sets.

3.1 MLM with different distance metrics

Since the MLM is a distance based method, it is important to evaluate different distance calculation metrics and use the most suitable metric in the PC-MLM implementation. Fig. 4 shows that the Euclidean distance metric reached the overall best accuracy rate with all of the HS images. The Cosine distance metrics accuracy rate was significantly lower than with the other methods.

Based on these results, we selected the Euclidean distance method to our framework and all of the results shown this study are produced with it.

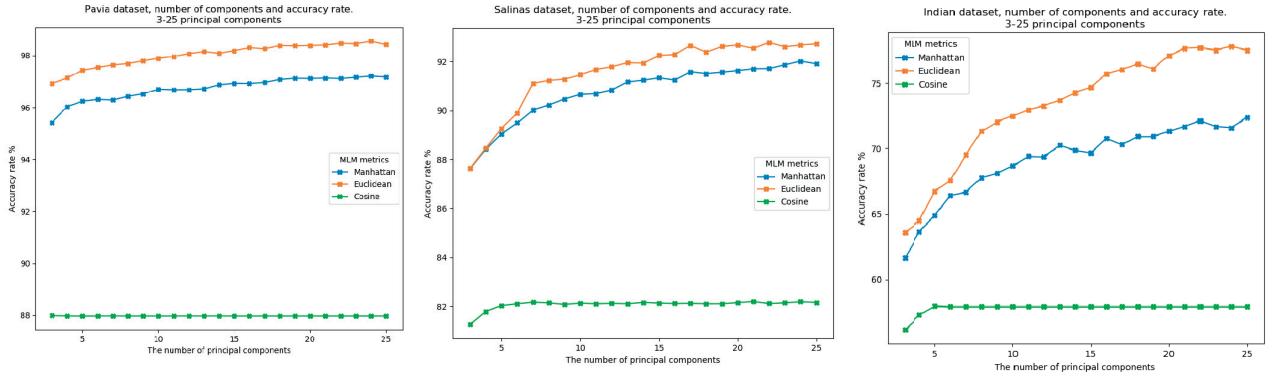


Figure 4. The comparison of the distance methods. The accuracy rate with different principal components (sizes of R).

3.2 The classifier comparison

Tables 3, 5 and 4 and figures 5, 7 and 6 shows that the MLM is one of the fastest classifiers in the comparison, and it can produce comparable accuracy results against other spectral classifiers. MLM can train the largest Pavia data set faster than the other classifiers. With smaller data sets, the training time of MLM is close to the fastest model, the kNN classifier. PC-MLM has reached the third place on the training time comparison with all of the data sets.

The classification maps (Figs. 5, 6 and 7) visualises the prediction results of all of the classifiers on the comparison. Maps were produced by training the models with each HS images training data, and by predicting with the whole data of the HS images.

Table 3. Pavia Centre method comparison, 25 principal components, 30 nearest neighbours

	MLM	25 PC-MLM	25 PC kNN	SVM	ANN	RF
Number of training samples	$X: 88891, R: 675$	$X: 88891, R: 675$	675	88891	88891	88891
Training time	5.92	12.39	6.93	24.21	552.54	36.86
Classification time	2.06	1.82	0.10	10.75	1.61	0.33
Accuracy rate	94.48	98.49	95.17	99.34	99.00	98.87

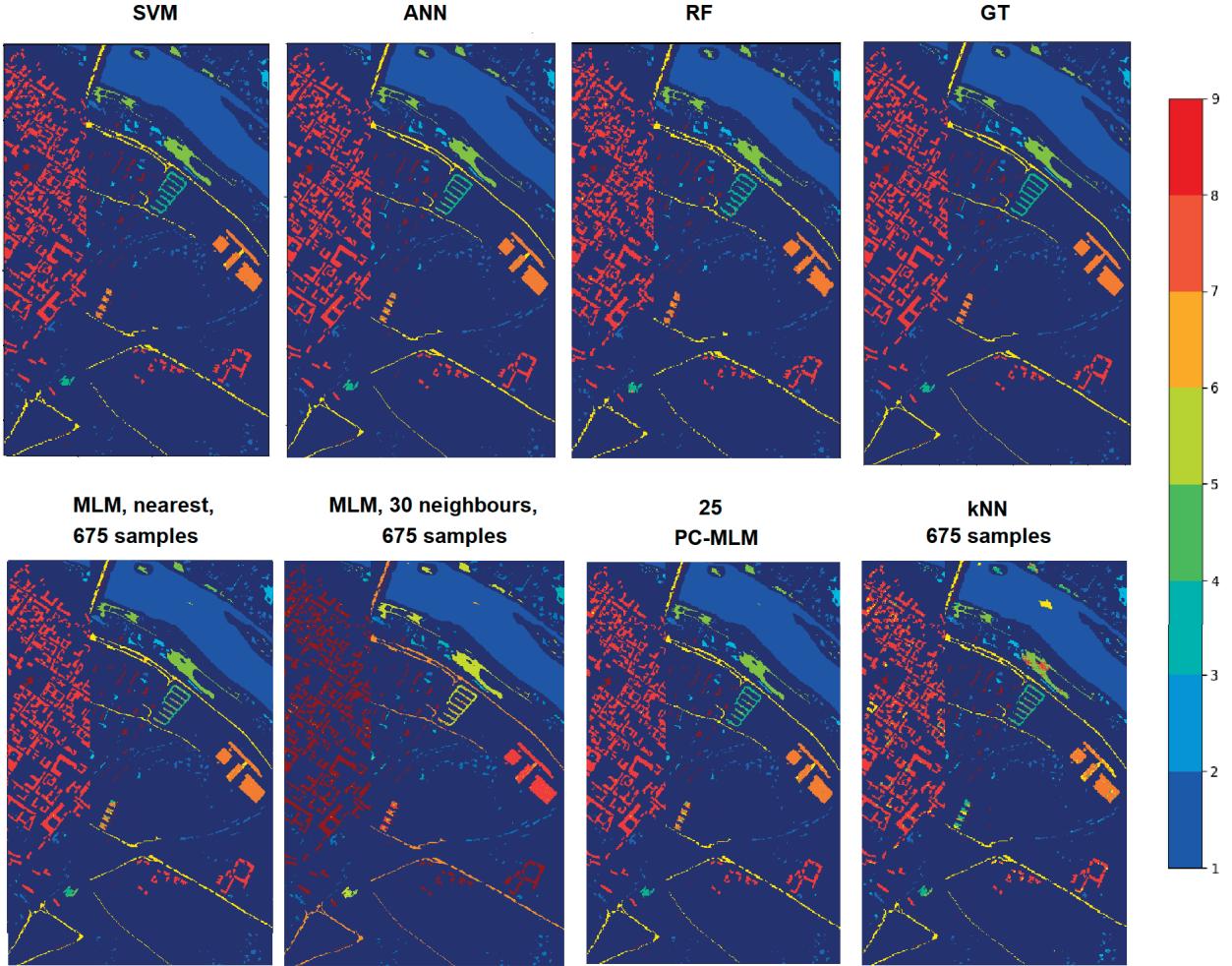


Figure 5. The classification maps of the Pavia Centre HS image. Maps are produced with the comparison classifiers. Numerical results of these maps are shown in the table 3.

The classification time comparison shows that the PC-MLM classifies all of the data sets faster than the MLM classifier. With Pavia data, the fastest classifier was kNN. RF was the fastest with Salinas and the Indian Pines comparison winner was ANN classifier.

The Accuracy rate comparison shows that the best results with Pavia data were achieved with SVM classifier. Smaller data sets reached best accuracy scores with RF classifier. Table 3 shows that the PC-MLM with 25 PC components reached promising results against the other classification methods in all of the categories. The Pavia data's results shows that with 25 PCA components and R size of 675, the PC-MLM is overall faster than SVM, ANN and RF and it reaches 98.49% accuracy.

Table 4. Indian Pines method comparison, 25 principal components, 30 nearest neighbours

	MLM	25 PC-MLM	25 PC kNN	SVM	ANN	RF
Number of training samples	$X: 6149, R: 1200$	$X: 6149, R: 1200$	1200	6149	6149	6149
Training time	1.51	2.12	0.88	5.14	6.82	5.39
Classification time	0.26	0.20	0.28	1.99	0.01	0.04
Accuracy rate	78.31	77.89	60.18	88.78	77.17	86.68

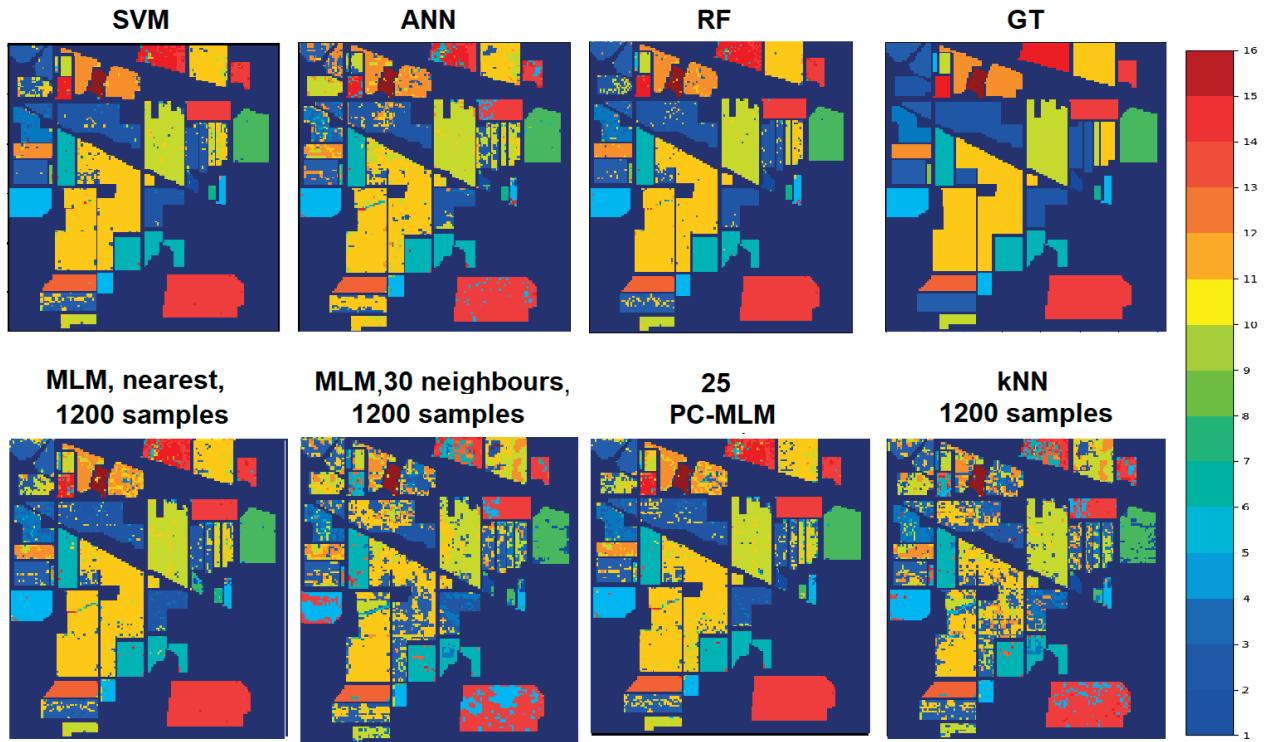


Figure 6. The classification maps of the Indian Pines HS image. Maps are produced with the comparison classifiers. Numerical results of these maps are shown in the table 4.

Table 5. Salinas method comparison, 25 principal components, 30 nearest neighbours

	MLM	25 PC-MLM	25 PC kNN	SVM	ANN	RF
Number of training samples	$X: 35477, R: 1200$	$X: 35477, R: 1200$	1200	32477	32477	32477
Training time	7.21	11.34	3.85	34.03	184.40	20.33
Classification time	1.62	1.40	0.06	22.57	0.93	0.22
Accuracy rate	93.01	92.82	85.31	93.82	92.57	94.71

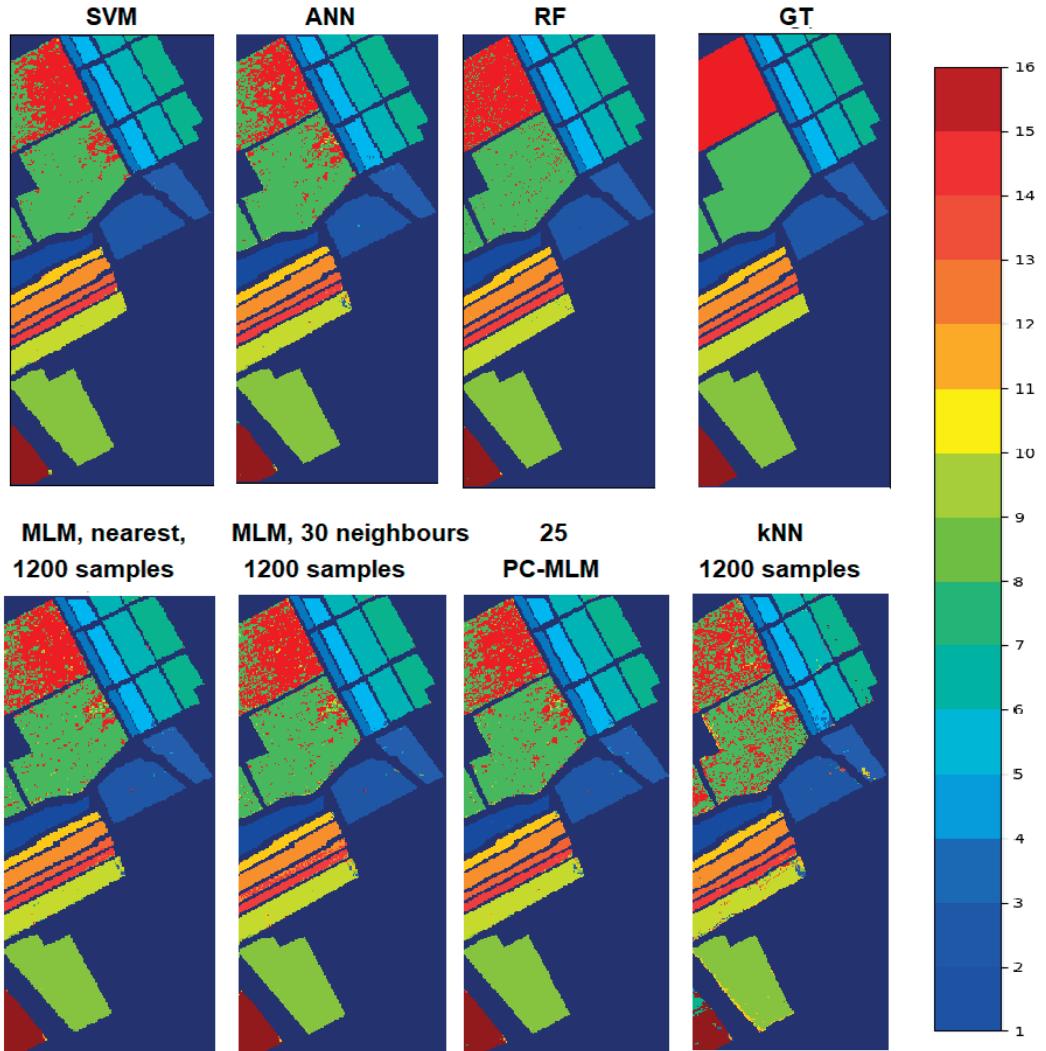


Figure 7. The classification maps of the Salinas HS image. Maps are produced with the comparison classifiers. Numerical results of these maps are shown in the table 5.

The previous results shows that the MLM and PC-MLM classifiers can reach promising accuracy rates against the other classifiers and decrease the time of training and classification. Those advances can be seen on results of all of the data sets, regardless of the size of the data, but the best results can be accomplished with larger ones.

3.3 The performance of the MLM and PC-MLM models

Figure 8 shows the comparison between the PC-MLM classifier and MLM classifier with randomly picked training points (randomised MLM). The reference set R was formed with 6-30 principal components. The amount of the randomly picked training points were equal with the size of the R .

The results of each data set is presented separately on rows in figure 8. The left side represents the MLM and PC-MLM nearest neighbour classifiers. On the right side, there are the results of the MLM and PC-MLM 30 neighbour classifiers. The markers on the lines on the figures represents the amount of principal components, x-axis shows the number of selected training points and y-axis shows the accuracy rate percent or time in seconds.

Results were measured 10 times and the final results are the average of 10 results. The training data X was randomised in every round for the randomised MLM.

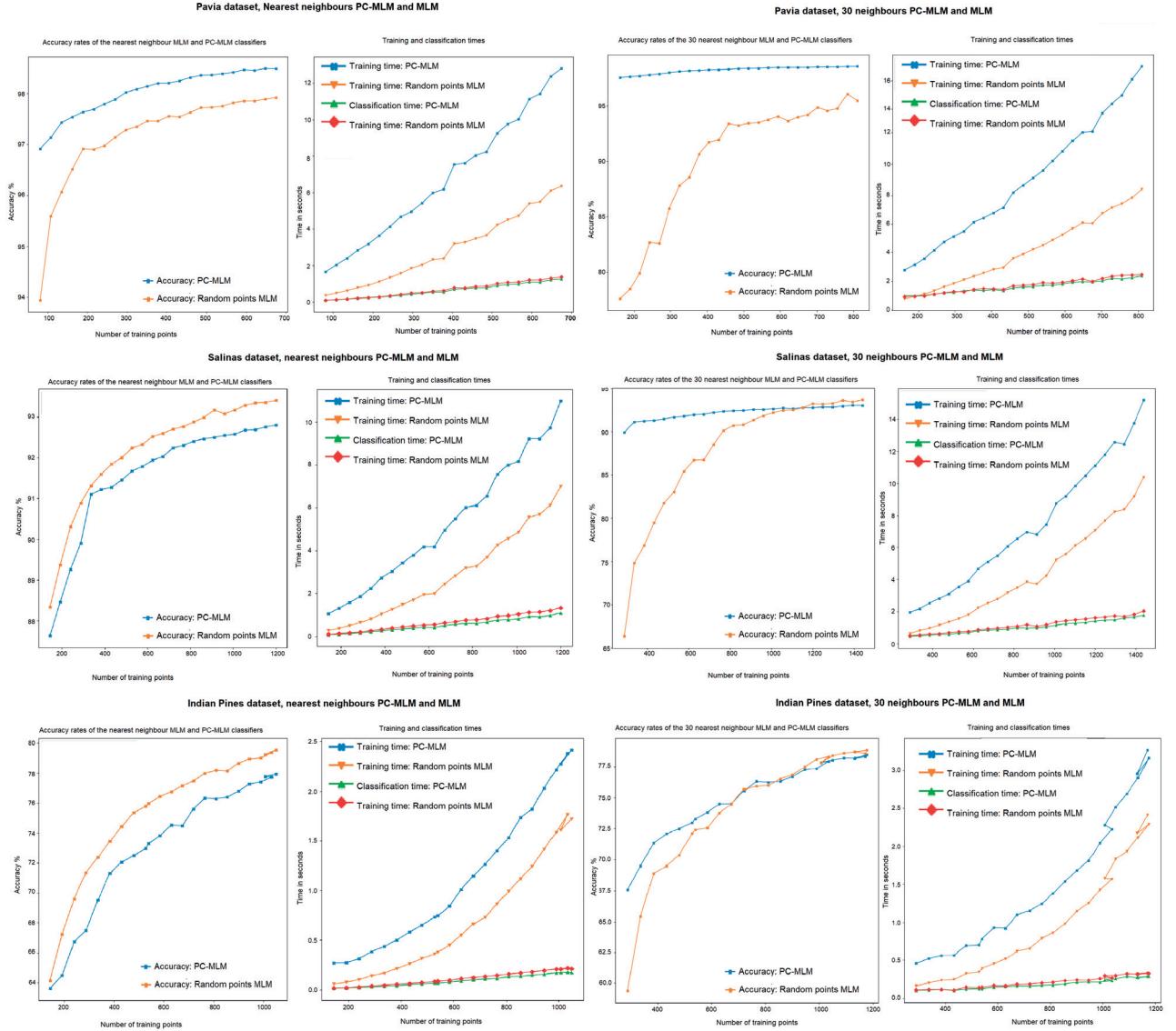


Figure 8. Pavia, Salinas and Indian Pines HS images, the comparison of the PC-MLM and randomised MLM with nearest and 30 nearest neighbour classifiers. The accuracy rate and the number of training points of different amount of principal components.

Figure 8 confirms an observation that the accuracy rate increases when the size of the training points increases. The structure of the data affects on the results. If the data set is large and it has less ground truth classes, it will most likely perform better than a smaller data set with high number of classes.

Table 6 shows that with Pavia data, the PC-MLM reached 97.54% accuracy rate with only 6 principal components, which means 162 selected training points from the whole training data (X) of 88891 samples. With the same amount of training points, the accuracy rate of randomised MLM was 96.47%. The Pavia data set's accuracy rate was better with the PC-MLM in all of the evaluation rounds with different amounts of selected training points. The best comparable accuracy rate of this comparison was nearest neighbour PC-MLM's 98.48%

with 675 training points. The result of the randomised MLM with same amount of training points was 97.92%. This PC-MLM's result can be improved by increasing the number of neighbours; the results in the tables 6, 8 and 7 shows, that the increased number of neighbours decreases the accuracy of the MLM classifier, but it may have a positive impact on the accuracy of the PC-MLM model.

Table 6. Pavia Centre, the number of the principal components, size of the R and the accuracy rates: PC-MLM nearest neighbour ans 30 neighbours classifiers against MLM nearest neighbour and 30 neighbours classifiers. The size of the training set X : 88 891 samples.

PC	The R size	PC-MLM nearest	MLM nearest	PC-MLM 30 neighbours	MLM 30 neighbours
3	81	96.91	95.20	46.50	73.16
4	108	97.13	95.24	70.35	74.72
5	135	97.43	96.11	70.56	75.92
6	162	97.54	96.47	97.54	76.85
10	270	97.88	97.14	97.88	84.97
15	405	98.02	97.55	98.23	90.78
20	540	98.39	97.75	98.37	93.41
25	675	98.48	97.92	98.49	94.48
30	810	-	-	98.55	95.33

We can see from the tables 6, 7, 8 and figure 8, that the results of the Pavia data differs from the results of the other data sets. The PC-MLM has better accuracy rates than randomized MLM in Pavia results. The trend of the increasing accuracy rate with the increments of the size of R is similar with all of the data sets. Salinas and Indian Pines results (figure 8, tables 7 and 8) shows, that the accuracy rate of the randomised MLM exceeds over the accuracy rate of the PC-MLM's accuracy.

With Salinas data, the MLM with randomised training points performed better on the accuracy rate comparison. For example with 25 principal components, the results were 92.82% (PC-MLM, 30 neighbours) and 93.41% (randomised MLM, nearest neighbour). Indian Pines data followed the similar trend with the Salinas data, but the difference of the accuracy rates was bigger. With 25 components, the MLM with nearest neighbours classifier reached 79.52% accuracy, where the PC-MLM could reach only 77.94% accuracy.

Table 7. Salinas, the number of the principal components, size of the R and the accuracy rates: PC-MLM nearest neighbour ans 30 neighbours classifiers against MLM nearest neighbour and 30 neighbours classifiers. The size of the training set X : 32 477 samples.

PC	The R size	PC-MLM nearest	MLM nearest	PC-MLM 30 neighbours	MLM 30 neighbours
3	144	87.63	88.34	14.15	48.21
4	192	88.46	89.38	49.06	52.52
5	240	89.27	90.31	49.07	61.50
6	288	89.90	90.89	89.90	68.66
10	480	91.45	92.00	91.46	81.41
15	720	92.25	92.70	92.23	88.82
20	960	92.55	93.08	92.61	92.12
25	1200	92.80	93.41	92.82	93.01

The training and classification time (fig. 8, table 3) follows the same pattern with all of the data sets. The randomised MLM was faster on the training, but more time-consuming on the classification than the PC-MLM. For example the average classification time of with Pavia data and 675 training points was 2.06 seconds with randomized MLM, PC-MLM performed the same classification tasks on average of 1.82 seconds. The according average training times were 12.39 seconds (randomised MLM), and 5.92 seconds (PC-MLM).

Table 8. Indian Pines, the number of the principal components, size of the R and the accuracy rates: PC-MLM nearest neighbour ans 30 neighbours classifiers against MLM nearest neighbour and 30 neighbours classifiers. The size of the training set X : 6149 samples.

PC	The R size	PC-MLM nearest	MLM nearest	PC-MLM 30 neighbours	MLM 30 neighbours
3	144	63.61	65.15	12.49	48.29
4	192	64.49	67.23	34.78	53.49
5	240	66.73	69.60	36.93	56.53
6	288	67.49	71.37	67.46	61.41
10	480	72.49	75.53	72.35	71.18
15	675	74.49	77.15	74.47	74.68
20	900	76.79	78.65	76.63	77.58
25	1050	77.94	79.52	77.89	78.31
30	1170	-	-	78.31	78.54

4. DISCUSSION

The MLM and PC-MLM classifiers seems to perform well with hyperspectral images. The accuracy results are promising against the other classifiers. MLM and PC-MLM reaches comparable training and classification times towards most of the other methods. Those advances can be seen on all of the data sets, regardless of the size of the data set. However, the classifiers were optimized with the largest Pavia data set, which may have an affect on the performance with smaller data sets, like Salinas and Indian Pines.

Compared to the original MLM classifier, the new PC-MLM framework slightly increases the number of hyper-parameters with two obligatory and one optional parameter (distance method, the number of principal components and the number of neighbours). One of the benefits of the original MLM is that it has only one hyper-parameter to tune, which is the size of the randomly selected training points (R)⁹. Our model extends the amount of parameters, but it seems to still maintain MLM's overall advances of the easy implementation, fast classification time and good accuracy rate compared to the other models (see tables 3, 5, 4 and classification maps 5, 7 and 6). With the reference classifiers, the hyper-parameter tuning was time consuming, and especially the ANN classifier was hard and time-consuming to optimize, which confirms previous observations of using ANN classifier²². The RF and SVM classifier were easier and faster to optimise.

When we compare the PC-MLM's performance against the MLM with randomly selected points, we can see from the figure 8, that the PC-MLM is more time-consuming on the training, but it is faster on the classification phase. The reason, why the PC-MLM is fast in classification, is inside of the implementation of the reference point selection (algorithm 1). We present in the MLM chapter 2.1, that the optimization problem can be solved with the nearest neighbours method. After performing the reference point selection algorithm 1, our R and T are sorted and organised class-by-class. When we are picking randomly samples from the X , the R and T will remain unorganised. On the classification phase, the PC-MLM's $\delta(y_n, T)$ can be sorted faster than the MLM's $\delta(y_n, T)$.

The performance of all of the classifiers followed similar trend with each of the HS images. For example, the Indian Pines had the lowest performance (table 4), while the Pavia HS image reached the highest accuracy scores (table 3). One explanation for this trend can be that all of the classifiers were optimised only for the largest Pavia data set.

The MLM and the PC-MLM can classify data with significantly smaller amount of reference points (R) than SVM, ANN and RF classifiers. The MLM and PC-MLM uses the information from the whole training set X on the model $\hat{\mathbf{B}}$ (algorithm 2), but in the prediction phase, the classifier calculates only the distances between the new data and the reference set R . Model utilises the $\hat{\mathbf{B}}$ with the new distances and selects the labels for the new data with nearest neighbour method (algorithm 3). Therefore it is relatively simple and computationally less complex machine learning model compared to the reference models.

The study confirms previous findings that the RF classifier can perform well on the land-cover classifications and remote sensing²³. We trained SVM, ANN and FR models with training sets that had 60% amount of the HS image data points. With RF classifier, it is a good size for avoiding the over fitting²³. Compared to the MLM and PC-MLM classifiers training points, the size difference is remarkable. MLM and PC-MLM had only 675 samples and RF 88 891 samples on the Pavia classification task. The accuracy results were close to each other, but the MLM and PC-MLM could perform results faster than RF classifier.

Since there are similarities with the kNN classifier, it was an interesting to see, how the kNN classifier could perform with the same intentionally selected training points than the MLM and PC-MLM classifiers. The results shows, that the MLM's and PC-MLM's model performed more accurately than the kNN classifier.

As an limitation for the PC-MLM framework, these tests were done only with three HS images and only relatively well optimized classifiers. The PC-MLM framework is now ready for the future improvements. For example, it is an interesting question to solve, how the structure and the size of the data set, and the geometry of each class affects to the accuracy.

The HS images class-by-class principal component visualisations (fig. 1, 2 and 2) reveals the geometry of each class and the selected training points. The labels and amounts of the samples can be seen on the ground truth tables 1, 2. The prediction results are shown in the classification maps shows (figs. 5, 7 and 6).

With Pavia data, the amount of the samples in the ground truth labels had even distribution (table 1), the selected training points and their position can be seen on figure 1. Figure 1 shows that Pavia's training points had less scatter than the classes in the figures 2 (Salinas) and 3 (Indian pines). The reference point selection algorithm 1 worked well with Pavia data. Salinas and Indian Pines were more difficult to predict. Salinas had more classes and samples than Pavia, but the image size was significantly smaller and the amount of spectral bands were twice as big as Pavia's.

Indian Pines was the smallest data set with the biggest variance on the amount of samples per class. The principal component visualisation (fig. 3) shows that the training points were scattered. Figure also shows, how the selected training points are representing the geometry of each class. The classification map 6 reveals, that the mistakes in the classification of the data set covers all of the classes, which differs from, for example, the classification map of Salinas 2, where the mistakes are more clearly focused on few categories.

Based on these observations, it would be interesting to research more the relationship between the geometry of the classes, the structure and size of the data and the intentional reference point selection method (algorithm 1). For example, when the minimum and maximum positions were moved closer to the median position, the optimization of the new locations were done with Pavia data. This movement increased the accuracy rate, and it was necessary because of the noise in the data. It would have been interesting to see, how the accuracy rate results would have behaved, if this optimisation has been done individually to each data set.

We tested MLM and PC-MLM with a large HS image and small number of classes (Pavia Centre), a large HS image with relatively large number of classes (Salinas) and with a small HS-image and relatively large number of classes with wide range on the amount of samples per classes (Indian Pines). It might have been interesting to see how it would have affected on all of the classifiers if we would have performed these tests with a small data set and small number of classes. Some of the previous classification studies (eg.²⁴) has cleaned the Indian Pines data, by limiting the classes to the largest ones, since there are classes that has only limited amount of samples (see from table 2). It would have been interesting to try this experiment again with Indian Pines data and limited classes.

Another idea to improve the PC-MLM is to study the relation between the number of components and number of neighbours. The results indicates that with a low number of principal components it might be good

to use the nearest neighbour in the predicting phase, but when we are optimising the classifier, there is a point with the number of principal components, where the number of neighbours should be increased to achieve better accuracy rate. The way we select the positions of the minimum and maximum values seems also have an affect on the number of neighbours, which means that there might be another place to improve the PC-MLM method. Finding out these optimisation rules might improve and simplify the optimisation process on the future.

The feature extraction (FE) methods are important in hyperspectral data classification. Different FE methods can be used for extraction of geometric structures, shape and texture from the HSI²⁵. The aim of the FE is to get an excellent representation from the original data. On this study, we used the PCA, which is a popular method. Our findings confirms the previous research of the FE methods. With those methods, we can reduce the number of the training samples (R size) and reduce the computing time, without decreasing the accuracy rate. The FE is an important step before using the HS image classification methods^{10, 26}.

5. CONCLUSIONS

The MLM is an effective tool for HS image classification. Comparison against four other spectral classifiers reveals, that MLM is an easy to implement, it provides good accuracy rates and while classifying, it consumes less time than the reference models SVM, ANN, RF and kNN.

MLM's accuracy rate and classification time can be improved by selecting the reference points class-by-class intentionally with principal component analysis to respect the geometry of each class. The proposed framework for improving the performance is called the PC-MLM. However, the improvement is dependent on the size of the data. The results indicates, that MLM's performance can be improved with large HS images, but with smaller HS images, the basic version of the MLM might perform better than the PC-MLM.

ACKNOWLEDGMENTS

This study is partly funded by Jane and Aatos Erkko Foundation (Grant No. 170015) and Academy of Finland (Grant No. 327862).

REFERENCES

- [1] Camps-Valls, G. and Bruzzone, L., “Kernel-based methods for hyperspectral image classification,” *IEEE Transactions on Geoscience and Remote Sensing* **43**(6), 1351–1362 (2005).
- [2] Camps-valls, G., Tuia, D., and Bruzzone, L., “Advances in hyperspectral image classification,” *IEEE Signal Processing Magazine* (January), 45–54 (2013).
- [3] Khan, M. J., Khan, H. S., Yousaf, A., Khurshid, K., and Abbas, A., “Modern Trends in Hyperspectral Image Analysis: A Review,” *IEEE Access* **6**, 14118–14129 (2018).
- [4] Pöölönen, I., Rahkonen, S., Annala, L., and Neittaanmäki, N., “Convolutional neural networks in skin cancer detection using spatial and spectral domain,” *10851*, 10 (2019).
- [5] Tuominen, S., Näsi, R., Honkavaara, E., Balazs, A., Hakala, T., Viljanen, N., Pöölönen, I., Saari, H., and Ojanen, H., “Assessment of classifiers and remote sensing features of hyperspectral imagery and stereophotogrammetric point clouds for recognition of tree species in a forest area of high species diversity,” *Remote Sensing* **10**(5), 1–28 (2018).
- [6] Ghamisi, P., Plaza, J., Chen, Y., Li, J., and Plaza, A. J., “Advanced Spectral Classifiers for Hyperspectral Images: A review,” *IEEE Geoscience and Remote Sensing Magazine* **5**(1), 8–32 (2017).
- [7] He, L., Li, J., Liu, C., and Li, S., “Recent Advances on Spectral-Spatial Hyperspectral Image Classification: An Overview and New Guidelines,” *IEEE Transactions on Geoscience and Remote Sensing* **56**(3), 1579–1597 (2018).
- [8] Hughes, G. F., “Comments “on the Mean Accuracy of Statistical Pattern Recognizers”,” *IEEE Transactions on Information Theory* **IT-15**(3), 420–423 (1969).
- [9] de Souza, A. H., Corona, F., Barreto, G. A., Miche, Y., and Lendasse, A., “Minimal Learning Machine: A novel supervised distance-based approach for regression and classification,” *Neurocomputing* **164**, 34–44 (2015).

- [10] Zhao, B., Ulfarsson, M. O., Sveinsson, J. R., and Chanussot, J., “Unsupervised and Supervised Feature Extraction Methods for Hyperspectral Images Based on Mixtures of Factor Analyzers,” *Remote Sensing* **12**(7), 1179 (2020).
- [11] Yanni, Dong (IEEE), Du, Bo, (IEEE), and Zhang, L. I., “Dimensionality Reduction and Classification of Hyperspectral Images Using Ensemble Discriminative Local Metric Learning,” *Indonesian Journal of Electrical Engineering and Computer Science* **3**(3), 503–511 (2016).
- [12] Plaza, A., Martínez, P., Plaza, J., and Pérez, R., “Dimensionality reduction and classification of hyperspectral image data using sequences of extended morphological transformations,” *IEEE Transactions on Geoscience and Remote Sensing* **43**(3), 466–479 (2005).
- [13] Mesquita, D. P., Gomes, J. P., and Souza Junior, A. H., “Ensemble of Efficient Minimal Learning Machines for Classification and Regression,” *Neural Processing Letters* **46**(3), 751–766 (2017).
- [14] Varoquaux, G., Buitinck, L., Louppe, G., Grisel, O., Pedregosa, F., and Mueller, A., “Scikit-learn,” *Get-Mobile: Mobile Computing and Communications* **19**(1), 29–33 (2015).
- [15] Graña, M., Veganzons, M., and Ayerdi, B., “Grupo de iInteligencia Computational (CIC).” Hyperspectral Remote Sensing Scenes http://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes. (Accessed 9 May 2020).
- [16] Xie, F., Li, F., Lei, C., Yang, J., and Zhang, Y., “Unsupervised band selection based on artificial bee colony algorithm for hyperspectral image classification,” *Applied Soft Computing Journal* **75**, 428–440 (2019).
- [17] Sawant, S. S. and Manoharan, P., “Unsupervised band selection based on weighted information entropy and 3D discrete cosine transform for hyperspectral image classification,” *International Journal of Remote Sensing* **41**(10), 3948–3969 (2020).
- [18] Petropoulos, G. P., Kontoes, C. C., and Keramitsoglou, I., “Land cover mapping with emphasis to burnt area delineation using co-orbital ALI and Landsat TM imagery,” *International Journal of Applied Earth Observation and Geoinformation* **18**(1), 344–355 (2012).
- [19] Breiman, L., “Bagging predictors,” *Machine Learning* **24**(2), 123–140 (1996).
- [20] Breiman, L., “Random forests,” *Machine learning* (45), 5–32 (2001).
- [21] Breiman, L., “Arcing Classifiers,” *Annals of Statistics* (26), 801–824 (1996).
- [22] Raczkowski, E. and Zagajewski, B., “Comparison of support vector machine, random forest and neural network classifiers for tree species classification on airborne hyperspectral APEX images,” *European Journal of Remote Sensing* **50**(1), 144–154 (2017).
- [23] Rodriguez-Galiano, V. F., Ghimire, B., Rogan, J., Chica-Olmo, M., and Rigol-Sánchez, J. P., “An assessment of the effectiveness of a random forest classifier for land-cover classification,” *ISPRS Journal of Photogrammetry and Remote Sensing* **67**(1), 93–104 (2012).
- [24] Zhao, B., Ulfarsson, M. O., Sveinsson, J. R., and Chanussot, J., “Unsupervised and Supervised Feature Extraction Methods for Hyperspectral Images Based on Mixtures of Factor Analyzers,” *Remote Sensing* **12**(7), 1179 (2020).
- [25] Imani, M. and Ghassemian, H., “An overview on spectral and spatial information fusion for hyperspectral image classification: Current trends and challenges,” *Information Fusion* **59**(October 2019), 59–83 (2020).
- [26] Casasent, D. P. and Chen, X.-w., “Waveband selection for hyperspectral data: optimal feature selection,” in [Optical Pattern Recognition XIV], **5106**, 259 (2003).

PII

PIECEWISE ANOMALY DETECTION USING MINIMAL LEARNING MACHINE FOR HYPERSPECTRAL IMAGES

by

Raita-Hakola Anna-Maria and Pölönen Ilkka 2021

ISPRS - Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, <https://doi.org/10.5194/isprs-annals-V-3-2021-89-2021>

PIECEWISE ANOMALY DETECTION USING MINIMAL LEARNING MACHINE FOR HYPERSPECTRAL IMAGES

A-M. Raita-Hakola^{a,*}, I. Pölönen^a,

^a Faculty of Information Technology, University of Jyväskylä, 40100, Jyväskylä, Finland -
(anna.m.hakola, ilkka.polonen)@jyu.fi

Commission III, WG III/4

KEY WORDS: Hyperspectral imaging, Minimal Learning Machine, Piecewise approach, Anomaly detection, Real-time computation, Machine learning

ABSTRACT:

Hyperspectral imaging, with its applications, offers promising tools for remote sensing and Earth observation. Recent development has increased the quality of the sensors. At the same time, the prices of the sensors are lowering. Anomaly detection is one of the popular remote sensing applications, which benefits from real-time solutions. A real-time solution has its limitations, for example, due to a large amount of hyperspectral data, platform's (drones or a cube satellite) constraints on payload and processing capability. Other examples are the limitations of available energy and the complexity of the machine learning models. When anomalies are detected in real-time from the hyperspectral images, one crucial factor is to utilise a computationally efficient method. The Minimal Learning Machine is a distance-based classification algorithm, which can be modified for anomaly detection. Earlier studies confirms that the Minimal learning Machine (MLM) is capable of detecting efficiently global anomalies from the hyperspectral images with a false alarm rate of zero. In this study, we will show that by using a carefully selected lower threshold besides the higher threshold of the variance, it is possible to detect local and global anomalies with the MLM. The downside is that the improved method is highly sensitive with the respect to the noise. Thus, the second aim of this study is to improve the MLM's robustness with respect to noise by introducing a novel approach, the piecewise MLM. With the new approach, the piecewise MLM can detect global and local anomalies, and the method is significantly more robust with respect to noise than the MLM. As a result, we have an interesting, easy to implement and computationally light method which is suitable for remote sensing applications.

1. INTRODUCTION

The Hyperspectral (HS) image typically consists of a stack of frames, where each frame represents the intensity of a different wavelength of light, and each pixel has its spectrum. The HS image anomaly detection is a process where the image is processed pixel-by-pixel. Each pixel spectrum is evaluated, and the aim is to detect pixels whose spectral signature differs from their surroundings. The high-dimensional HS data is suitable for the identification, characterisation and anomaly detection tasks of the targets with high accuracy and robustness (Camps-Valls and Bruzzone, 2005, Bioucas-Dias et al., 2013). The challenges of the spectral anomaly detection methods are usually combinations of the large amounts of data, platform's constraints on payload, processing capability, and restricted available energy with complex machine learning models (Haut et al., 2018, Caba et al., 2020).

The technical evolution of the earth observation instruments on the airborne and satellite platforms have raised the sensor capability of producing an almost continuous high-dimensional data stream (Chen et al., 2018). In remote sensing platforms, the exponentially growing high dimensional data challenges the real-time processing, the data analysis processes and the technical features (Chen et al., 2018, Bioucas-Dias et al., 2013). Despite the challenges, one of the main advantages of real-time processing is improved data quality since it is not compressed and transmitted to the processor (Chen et al., 2018). The higher precision raw data might increase the accuracy of the data analysis. Other advantages are the reduced need for the communication between the ground equipment and the platform, the reduced need for the data

processing on the ground and the possibility to get the real-time responses from the platform (Che et al., 2018).

The Minimal Learning Machine (MLM) is easy to implement, computationally efficient and fast machine learning method (de Souza et al., 2015) which is an effective alternative for detecting global anomalies from the HS images (Pölönen et al., 2020). MLM is a distance-based method, which utilises the mapping between the input and the output distances. Input distance is the distance between the training set and its subset R , representing the selected training points. The output distance is calculated from the label values of the training set X to the subset R 's label values. In this study, we will calculate a linear model between the distances and estimate whether a certain pixel spectrum is an anomaly by using threshold values (Pölönen et al., 2020). The approach we are using is an example of a semi-supervised learning method (Prasad et al., 2009).

This study is an independent continuation of the previous research (Pölönen et al., 2020). The study has been implemented using consistent data with the previous research and using the same methods accurately. The aim is to improve the previous results with a new approach by proving that the MLM can detect local and global anomalies. With the new approach, the method can be more robust for the noisiness of the data. The research differs from the previous with its two test setups. The first setup will concentrate on the local and global anomalies by examining the variance thresholds. The second setup will introduce the piecewise MLM approach and provide robustness for the noise.

Our hypothesis is that by implementing the MLM with a piecewise approach, class-by-class, we can significantly improve results of the previous MLM anomaly detection method (Pölönen

*Corresponding author.

et al., 2020). We will show that the MLM anomaly detection method (Pöllönen et al., 2020) is capable of detecting both local and global anomalies, by using the lower and higher variance threshold values, but the method is still highly sensitive with respect to the noise. The new piecewise MLM is capable of detecting the local and global anomalies being more robust with respect to the noise. The piecewise MLM increases the accuracy of the MLM anomaly detection method, but on the downside, it might not be as fast as the MLM anomaly detecting method.

The paper is organised as follows. Section 2. describes the methods and demonstration materials. The results are introduced in section 3.. The discussion is the fourth section, and the final conclusions can be found in section 5..

2. MATERIAL AND METHODS

2.1 Minimal Learning Machine

The Minimal Learning Machine (MLM) is a computationally cheap distance-based machine learning method (de Souza et al., 2015). With HS images, the MLM offers tools for classification (Hakola and Pöllönen, 2020) and anomaly detection (Pöllönen et al., 2020) applications. The basic idea of the MLM is to utilise linear mapping between the distances of input and output.

When we are implementing the MLM with HS images, the input distances are $d(\mathbf{x}_i, \mathbf{m}_k)$ and output distances $\delta(\mathbf{y}_i, \mathbf{t}_k)$, where $\mathbf{x}_i \in X \subset R^D$ are the training set with D wavebands and $\mathbf{m}_k \in R$ are the randomly sampled subset of X . The labels of the training set Y and its subset $\mathbf{t}_k \in T$ are correspondingly $\mathbf{y}_i \in Y \subset R$ and $\mathbf{t}_k \in T$. The size of the training set X is N samples, and the size of the subset R is K samples.

By defining two matrices, based on these distances $\Delta_y \in R^{N \times K}$ and $\mathbf{D}_x \in R^{N \times K}$, and assuming the linear mapping between these two distance matrices, we have a linear model

$$\Delta_y = \mathbf{D}_x \mathbf{B} + \mathbf{E}, \quad (1)$$

where \mathbf{B} are the coefficients and \mathbf{E} is the residual. Coefficients \mathbf{B} can be approximated using the ordinary least squares estimator

$$\hat{\mathbf{B}} = (\mathbf{D}_x^T \mathbf{D}_x)^{-1} \mathbf{D}_x^T \Delta_y. \quad (2)$$

As a result, the $\hat{\mathbf{B}}$ is a linear model between the distances of $\delta(\mathbf{y}_i, \mathbf{t}_k)$ and $d(\mathbf{x}_i, \mathbf{m}_k)$. Distances between a new spectrum \mathbf{x}_n and its label \mathbf{y}_n is

$$\delta(\mathbf{y}_n, T) = d(\mathbf{x}_n, R) \hat{\mathbf{B}}. \quad (3)$$

Outputs \mathbf{y}_n can be estimated by solving an optimisation problem

$$\min_{\mathbf{y}_n} \sum_{k=1}^K ((\mathbf{y}_n - \mathbf{t}_k)^T (\mathbf{y}_n - \mathbf{t}_k) - \delta^2(\mathbf{y}_n, T))^2. \quad (4)$$

The \mathbf{y}_n is the computationally most expensive part of the classification. On the anomaly detection version, we do not have to estimate it. If the new spectrum \mathbf{x}_n is inside of the training set, the label \mathbf{y}_n is nearby points in the subset T . Then, the distribution of estimated distances $\delta(\mathbf{y}_n, T)$ should be relatively similar to training phase distances in Δ_y . Whether the \mathbf{x}_n is an anomaly or outlier can be detected in $\delta(\mathbf{y}_n, T)$ by studying its behaviour.

In this study, we use the variance

$$\text{Var}(\delta(\mathbf{y}_n, T)) = \frac{1}{K} \sum_{i=1}^K (\delta_i - \bar{\delta})^2 \quad (5)$$

to detect anomalies, where K is number of elements in a vector δ and $\bar{\delta}$ is average value of the δ . For the variance, we set two threshold values that reveals the anomalous spectra from the dataset. The lower threshold is the key to the detection of the local anomalies and the higher threshold exists for the global anomalies.

The threshold values must be selected carefully. Therefore, it is a useful practice to use an optimiser. It can be a parameter search function that loops through different combinations of lower and higher threshold values, calculates and compares the accuracy rates and returns the highest score threshold values.

2.2 Piecewise MLM approach

In this study, we developed a piecewise MLM approach, which aims to improve the accuracy rate of the previous version of the MLM anomaly detection method (Pöllönen et al., 2020) by including the detection of the local anomalies to the results being more robust for the noisiness of the data.

Algorithm 1: Training of the piecewise MLM

Input: X, R, Y, T
 Calculate distance \mathbf{D}_x
for each class i do
 Re-label class i to 0 and other classes to 1
 Calculate distance Δ_y
 Calculate $\hat{\mathbf{B}}_i = (\mathbf{D}_x^T \mathbf{D}_x)^{-1} \mathbf{D}_x^T \Delta_y$
end

Return: Piecewise MLM model $\hat{\mathbf{B}}_{pw} = [\hat{\mathbf{B}}_1, \hat{\mathbf{B}}_2, \dots]$

Algorithm 2: Anomaly detection using the piecewise MLM model

Input: new data X_{new} , R , $\hat{\mathbf{B}}_{pw}$, upper and lower thresholds γ_l and γ_u
 Calculate distance $d(X_{new}, R)$
for each $\hat{\mathbf{B}}_i$ do
 Calculate $\delta(\mathbf{y}_{new}, T)_i = d(X_{new}, R) \hat{\mathbf{B}}_i$
 Calculate $\text{Var}(\delta(\mathbf{y}_{new}, T))_i$
end
 Calculate $score = \sum_i \text{Var}(\delta(\mathbf{y}_{new}, T))_i$
if $\gamma_l < score < \gamma_u$ **then**
 $A_n = 1$
 else
 $A_n = 0$
end
Return: Anomaly detection A_n

The piecewise MLM approach means that instead of one MLM model, we are re-training class-by-class the linear model $\hat{\mathbf{B}}_i$. The training set X and the subset R remain unchanged. The labels Y and its subset T are updated so that every spectrum that does not belong to the class, regardless of whether it belongs to the anomalies or another classes are labelled as anomalies. The algorithm 1 shows the implementation for the piecewise MLM training. Fig. 1 explains the idea of the piecewise MLM implementation by visualising the class-wise re-labelled ground truth data Y .



Figure 1: The piecewise MLM re-trains the MLM model class-by-class. Above is the visualisation of the re-labelled Y , where only the class we are expecting is labelled as zero (0). All the rest of the pixel spectrums, including the anomalies, are re-labelled with one (1), which means an anomaly. Zero (0): Purple, an expected spectrum, One (1): Yellow, an anomaly spectrum. The subset T is labelled similarly.

The detection of the actual anomalies is shown in the algorithm 2. The algorithm uses piecewise trained MLM model $\widehat{\mathbf{B}}_i$ for each class i . Then the Equation 5 is calculated for each class. These variances are summed together and both of the lower and upper thresholds are used to detect anomalies. In our tests, we were using the Python libraries NumPy, scikit-learn and sciPy. The computations on this study were done with a Dell laptop (Intel Core i7-9850H and 16GB memory).

2.3 ColorChecker data

In this study, we used the same datasets with the previous study of the MLM anomaly detection method (Pöllönen et al., 2020). This way, the results are comparable with the previous results. The first data is a HS image, captured using a visible and near-infrared HS imager, which is manufactured by the VTT (Saari et al., 2013). The dataset represents a four colours subset of a X-Rite's ColorChecker and it has 100 wavebands from 450nm to 750nm. The dataset is divided into training and testing portions. On the training set, we had 2500 pixels. Each class is represented with 625 pixels. The training set does not have any anomalous pixels.

The testing set represents the same colours as the training set, but it has a different spatial location. The testing set has a similar size and portions to the training set, but it contains randomly placed 30 anomalous pixels, which are captured from the different parts and colours of the ColorChecker. The first test setup with ColorChecker data was implemented without noise (both training and test data sets), on the second test setup, the measurements were done with different amounts of generated noise in the data. The noise drawn from the uniform distribution was added to each pixel. In the test, the level of the noise varied between 0-0.4.

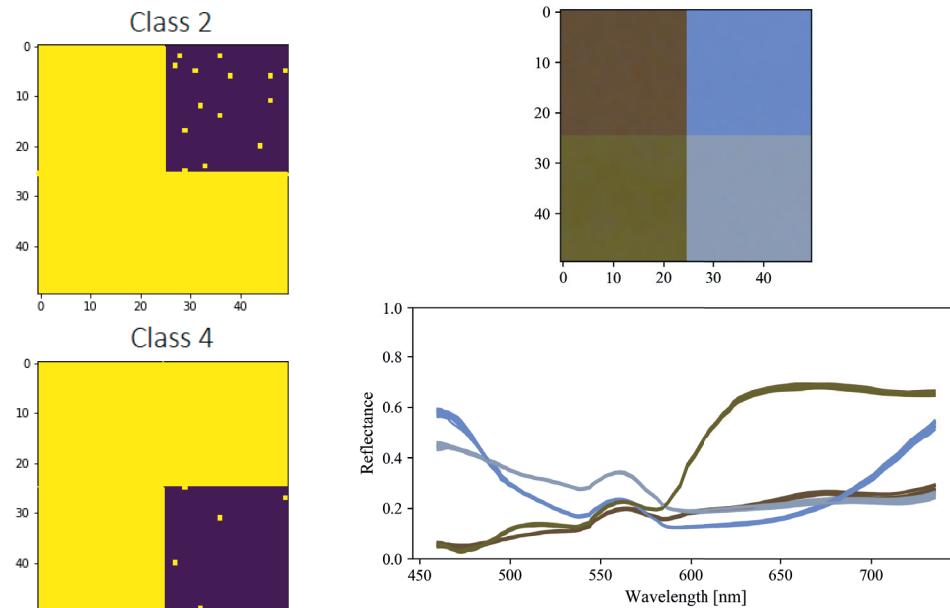


Figure 2: Illustration of the training dataset. Above is an "RGB" presentation of the ColorChecker dataset in the spatial dimension. Below are the samples of the reflectance spectra from each of the classes. Image is originally from (Pöllönen et al., 2020).

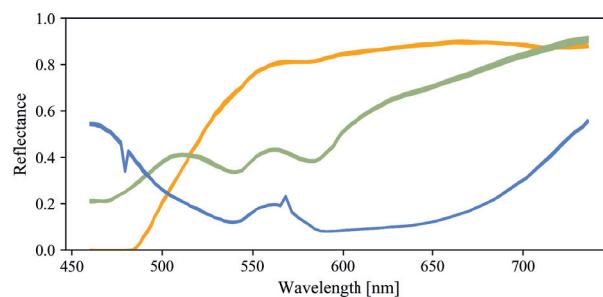


Figure 3: The spectra of the anomalous pixels on the ColorChecker data, which are randomly distributed to the test dataset. The test dataset has the same dimensions as the training set ($50 \times 50 \times 100$). Image is originally from (Pöllönen et al., 2020).

The training data and the ground truth data were randomised on the implementation phase and subset R with its 250-pixel spectra were selected randomly.

2.4 Forest data

The second dataset represents a Finnish forest. The tree species are mainly spruces, pines and birches. The dataset was originally captured for the tree species classification purposes (Nevalainen et al., 2017, Nezami et al., 2020, Polonen et al., 2018). The details of the dataset are described in (Nevalainen et al., 2017). The forest dataset consists of orthophoto mosaic images of the area (Honkavaara et al., 2013). The dataset was captured with a high spatial resolution (9 cm) ground sampling distance (GSD), and it has 38 spectral bands from 507 nm to 820 nm. In this study, we used a $(1500 \times 1400 \times 38)$ subset of the original data for the training. The selected training data is illustrated as a narrowband RGB image in Fig. 4, image A.

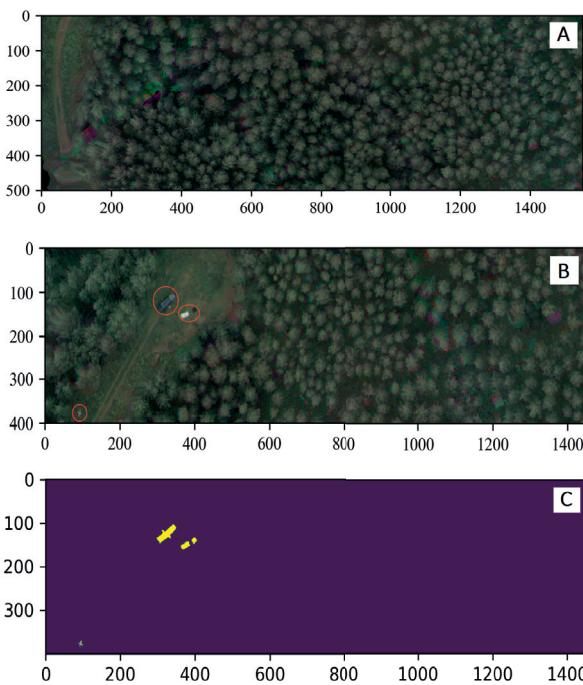


Figure 4: Image A: The training data for the anomaly detection from the forest. Image B: The forest dataset's test data for the anomaly detection. Inside the red circled areas are the anomalous objects (three reflectance panels and one black panel, one blue van, and a cross-shaped georeferencing signal). The third image is the hand-drawn ground truth image, for the test data. Images A and B are originally from (Pöllönen et al., 2020).

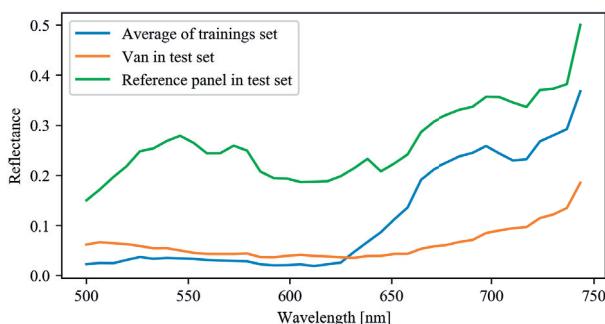


Figure 5: The mean spectra of the training dataset (forest data) and the samples from the van and the reflectance panel (nominal reflectivity of 0.5). Image is originally from (Pöllönen et al., 2020).

The testing data is represented in Fig. 4, image B. It is a subset from the same remote sensing data as the training data, but it has some anomalies that are marked with red circles on the figure. The anomalies are three reflectance panels and one black panel, a blue van and a cross-shaped georeferencing signal. More details of these anomalies can be found from (Pöllönen et al., 2020). Fig. 5 represents the average spectra of the training set compared to the example spectra of the anomalies.

The preprocessing of the forest data performed as follows. At first, the 100 000 pixel spectra samples were selected randomly from the training HS image for the training set X . The subset R

(100 samples) were selected randomly from the training set X . We performed a k-means clustering ($k=3$) (Pedregosa et al., 2011) for testing, which produced the needed ground truth labels for the HS image. The clustering time was excluded from the measured training and detecting times. We created the ground truth labels for the testing data from carefully pixel-by-pixel hand-drawn ground truth image. The ground truth image is visualised in Fig. 4, image C.

3. RESULTS

3.1 Performance comparison

With the ColorChecker datasets, the MLM and the piecewise MLM were implemented using cosine distance metric (De Carvalho and Meneses, 2000, Yuhas et al., 1992) and variance with one (upper) or two (upper and lower) variance thresholds. The tests were repeated 20 times, and the reported results of the ColorChecker data are the mean values from the tests. The computationally interesting measures were the training and detection times. Accuracy was also calculated using the Scikit-learn metrics accuracy score method (Varoquaux et al., 2015)).

At first, we used only the upper threshold with the MLM, which was done similarly with the previous study (Pöllönen et al., 2020). The threshold for the MLM's higher variance was set to > 0.5 . As a result, the method was not able to detect the local anomalies (Fig. 6). In the second phase, we used the MLM with two thresholds. The upper was set to > 0.5 and the lower to < 0.12 . After setting the thresholds, the accuracy improved significantly. With the piecewise MLM, the variance thresholds were set to > 0.7 and < 0.4 , respectively.

	A	B	C
Training time [s]	0.084	0.069	0.290
Testing time [s]	0.047	0.059	0.234
Accuracy [%]	80	98	100

Table 1: Comparison of the computation times for the ColorChecker dataset. A: MLM (cosine, variance, training set size 2500, R size 250), higher threshold: > 0.5 . B: MLM (cosine, variance, training set size 2500, R size 250), lower threshold < 0.12 , higher threshold: > 0.5 . C: Piecewise MLM (cosine, variance, training set size 2500, R size 250), lower threshold > 0.7 , higher threshold < 0.4 .

Table 1 shows that the MLM is faster at training and detecting anomalies than the piecewise MLM. Using two carefully selected thresholds (the threshold values optimisation is described in 2.1), the MLM can detect both global and local anomalies with increased accuracy.

From Fig. 7 and Table 1, we can see that the accuracy rate is significantly improved with the second threshold and with the piecewise MLM approach. The MLM with the upper threshold reached 80% accuracy, leaving local anomalies undetected (red samples in Fig. 6). The MLM with two thresholds and the piecewise MLM could detect both the local and global anomalies. The accuracy of the MLM with two thresholds was 98%, and with the piecewise approach, it was 100%.

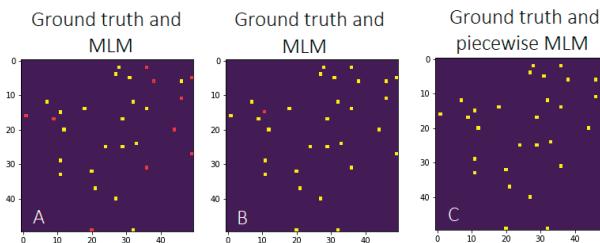


Figure 6: The joint anomaly detection maps of the ground truth and the MLM for the ColorChecker dataset. Yellow represents detected anomalies. Red spots are undetected local anomalies. A: MLM, only higher threshold, B: MLM with two thresholds, C: the piecewise MLM

3.2 Noise sensitivity test

Different amounts of uniformly distributed random noise were added to the training and test sets of the ColorChecker data. The accuracy of the anomaly detection from the data containing different amount of noise was evaluated with an area under curve (AUC) and by drawing the receiver operating characteristic curve (ROC) for the MLM and piecewise MLM setups.

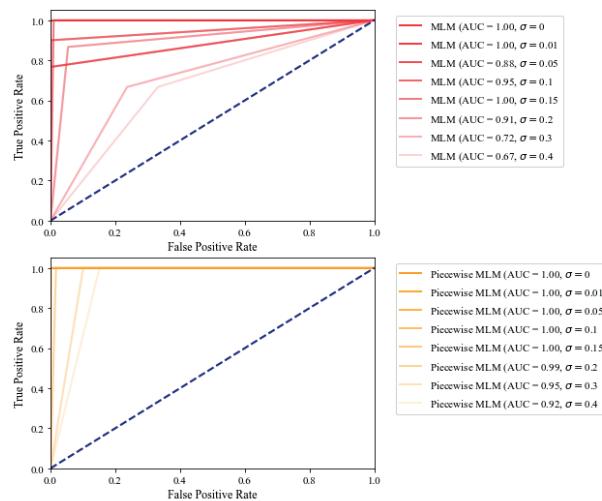


Figure 7: Comparison of the ROC-curves of the MLM and piecewise MLM with a different amount of noise in the data (the ColorChecker dataset). The noise drawn from the uniform distribution was added to each pixel. The level of the noise is denoted with σ . The ROC curves show that both approaches had no false positives with a low noise level. When the noise level increases, the piecewise MLM was significantly more robust with respect to the noise than the MLM.

The Fig. 7 shows that the piecewise approach is significantly more robust with respect to the noise than the MLM. Neither method had any false positives at first, but the number of false positives started to increase after adding noise. The effect of the noise on the performance is much more drastic for the MLM than it is for the piecewise MLM. The effect can be seen in how the increased noise affects the AUC. It ranges from 0.67 to 1.0 with the MLM and accordingly from 0.92 to 1.0 with the piecewise MLM.

The visualisation of the anomaly detection maps with the different noise levels in the data is shown in Fig. 8. Each pair is marked

with similar alphabets. The noise levels and the ground truth are shown in the figure. The effect of the decreasing accuracy, caused by the noise, can be seen from the MLM maps D-G, where the level of the noise was between 0.1 and 0.4 respectively. A similar effect can be seen with the piecewise MLM approach on F and G images (noise levels of 0.3 and 0.4).

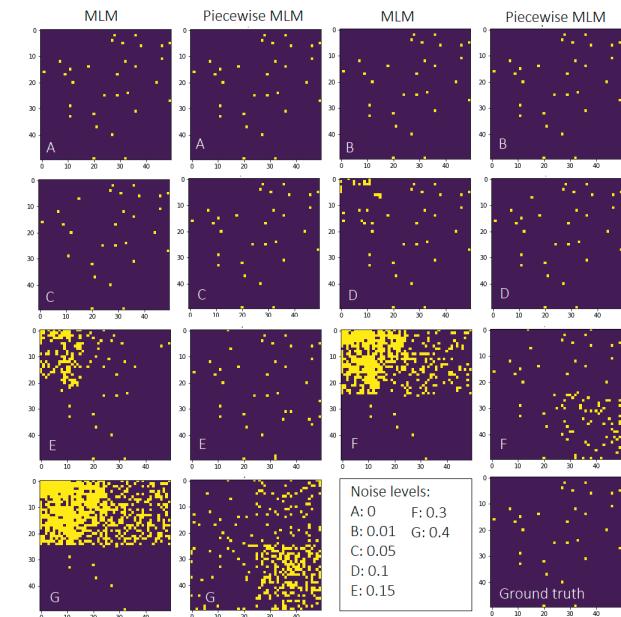


Figure 8: The ColorChecker dataset, the pairwise comparison of the MLM anomaly detection maps (on the left) and the piecewise MLM (on the right) with different amount of noise in the data. The level of the noise is marked with the alphabets.

3.3 Forest data

The visual evaluation of the forest data (Fig. 9) shows that both approaches can detect anomalies with a low false alarm rate. The most difficult anomaly to detect was the cross-shaped georeferencing signal on the left corner of the Fig. 4, image B. We tested the forest data with the MLM using different thresholds.

The MLM outperforms the piecewise approach in the forest data computation time comparison. It was the fastest method for both training and detecting tasks (Table 2).

Image on Fig.9	A-D	E-F
Training time [s]	0.390	1.238
Detecting time [s]	2.189	6.682

Table 2: The forest data, the comparison of the computation times for images on Fig. 9. A-D: MLM (cosine, variance, training set size 2500, R size 250). E-F: Piecewise MLM (cosine, variance, training set size 2500, R size 250).

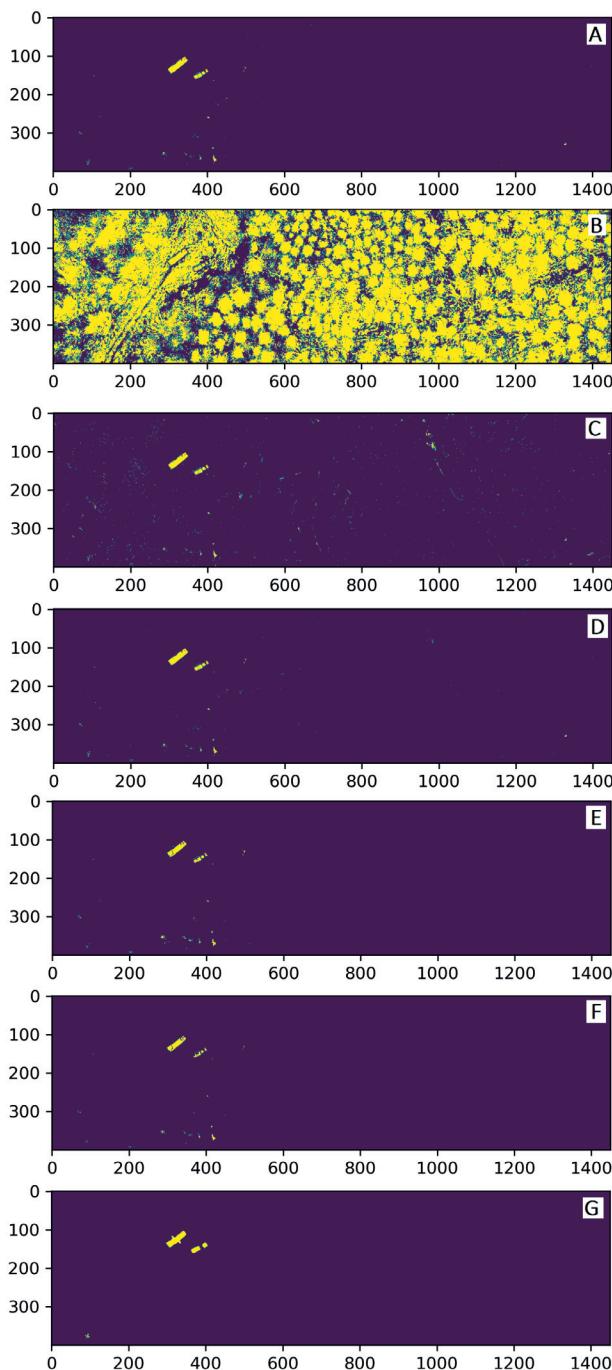


Figure 9: Comparison of the MLM (A-D) and the piecewise MLM (E-F) with the Ground truth (G) with the forest data. Thresholds and accuracy rates of the images are shown on Table 3.

Image on Fig.9	A	B	C
Higher threshold	0.1	0.1	0.1
Lower threshold	-	0.01	0.001
Accuracy rate [%]	99.86	30.10	99.57
Image on Fig.9	D	E	F
Higher threshold	0.1	1.2	1.0
Lower threshold	0.0001	0.07	0.07
Accuracy rate [%]	99.85	99.85	99.85

Table 3: The forest data, thresholds and accuracy rates for images on Fig. 9. A-C MLM, E-F Piecewise MLM.

The accuracy rate results show (Table 3) that if there is a need to use the lower limit, it must be chosen carefully. We can see from image B (Fig. 9) that if the lower limit is set too high, the false alarm rate will increase. Table 3 confirms that the range of the threshold is different between the methods, and the MLM lower threshold's sensitiveness is obvious.

The MLM performed the most accurately with only one threshold (image A in Fig. 9, Table 3), but images C, D and E show that the MLM can be implemented with two thresholds without compromising the accuracy rate significantly.

4. DISCUSSION

This study confirms the previous observations (Pöölönen et al., 2020) that the MLM is a fast and efficient method for detecting anomalies from the HS images. However, the method's weaknesses were the inability to detect local anomalies and the high sensitivity with respect to noise. This study proved that by using a carefully selected variance lower threshold value with the higher variance threshold value, it is possible to detect both local and global anomalies. The other main result of this study was the piecewise MLM, which seems to be a significantly more robust with respect to noise than the MLM. The ColorChecker noise comparison shows in Fig. 7 that the MLM's lowest AUC was 0.67, and with a similar amount of noise (0.4), the piecewise MLM approach reached 0.92 AUC.

With the MLM method, usually, size of the training data's subset R influences on the accuracy rate. By increasing the size of R , we can increase the accuracy of the results (de Souza et al., 2015, Hakola and Pöölönen, 2020). In this study, we increased the accuracy without increasing the size of R , which was 250 samples with the ColorChecker data, containing randomly picked samples from all of the classes. The piecewise MLM was implemented by training the MLM model class-by-class and re-labelling Y and T so that there were only two classes. On the implementation, zero (0) represented the expected spectrum and one (1) represented anomalies, containing all of the rest of the ground truth classes and anomalies (Fig. 1). With the ColorChecker data and the piecewise MLM approach, we trained the MLM four times, which was the number of the ground truth classes. The class-by-class models seemed to be more accurate, even though the actual size of R remained unchanged. One of the reasons why the piecewise MLM is outperforming with the noisy data is that the approach provides more accurately trained models since there are stricter rules on the deviations and expected values.

The piecewise MLM is more robust with respect to noise than the MLM, and therefore it is more sensitive to detect the local anomalies, even if they are close to the classes of the expected spectrum. With the MLM, the lower threshold value must be set carefully, and only the small enough values will work, but with the piecewise MLM, the range of the possible threshold values seems to be wider.

The downside of the piecewise MLM approach can be seen in Table 2. The average training times of the MLM ranged from 0.069 to 0.084 seconds, where the piecewise approach took 0.290 seconds. The detecting times were similarly from 0.047 to 0.049 and 0.234 seconds.

The MLM has been performing well against other anomaly detection methods (Pöölönen et al., 2020). Previous studies confirms that a one-class support vector machine (OC-SVM) is faster in training and anomaly detection than the MLM, but an Isolation forest is slower than the MLM. Global and local RX are slower

in the detection, but the training phase of the RX is faster than the MLM's training phase. The piecewise MLM is slower than the MLM in the training and anomaly detection, and it will probably lose against some of the mentioned reference methods. On the other hand, Fig. 10 shows promising results on the comparison of the anomaly maps. With the ColorChecker data, Fig. 10 shows that the piecewise MLM and the MLM are the most accurate methods and they performed without false positives.

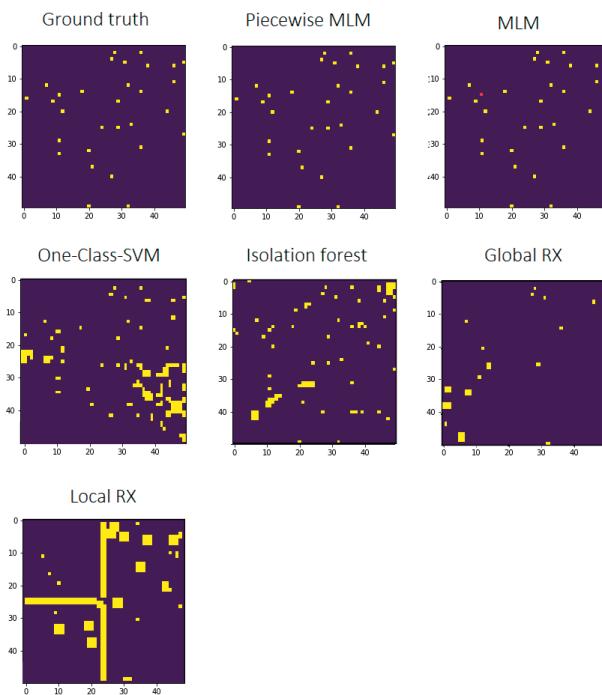


Figure 10: The anomaly detection maps of the ColorChecker dataset. The ground truth, the Piecewise MLM and the MLM: implemented with cosine distance metric, using variance and two thresholds (explained in subsection 3.1). The size of the X was 2500, and R was 250. The anomaly maps of the reference methods are originally from the (Pöllönen et al., 2020).

The forest data comparison and the visual evaluation (Fig. 9) shows that there was no significant improvement on the anomaly maps between the different approaches of the MLM. The MLM performed faster than the piecewise MLM approach. There were no improvements on the piecewise MLM's results towards the MLM because the data does not necessarily contain enough local anomalies or noisiness for the results to be improved.

We used the k-means clustering method for creating the labels for the forest training data (chapter 2.4). It occurs that the initialization of the clustering might affect the final results and accuracy. If the randomly selected locations of the initial clusters are, for example, somehow overlapping or close to the other final classes, some of the labels might not be in the right classes after the clustering is performed. Another note for the results is that the accuracy rates of the forest comparison (Table 3) were based on the hand-drawn ground truth image, which affects accuracy. Especially the borders of the anomalous objects were slightly unclear when we were drawing the ground truth image.

The previous studies confirms that the MLM was outperforming against the reference methods with the forest data (Pöllönen et al., 2020). Based on the visual evaluation (Fig. 11), this study confirms those findings. The MLM Forest data comparison (Table 3,

Fig. 9) shows that the threshold levels seem to be more sensitive with MLM than with the piecewise MLM. If the values are set too high or low, the accuracy will decrease. The range of the acceptable values was narrower on the MLM than in the piecewise MLM approach.

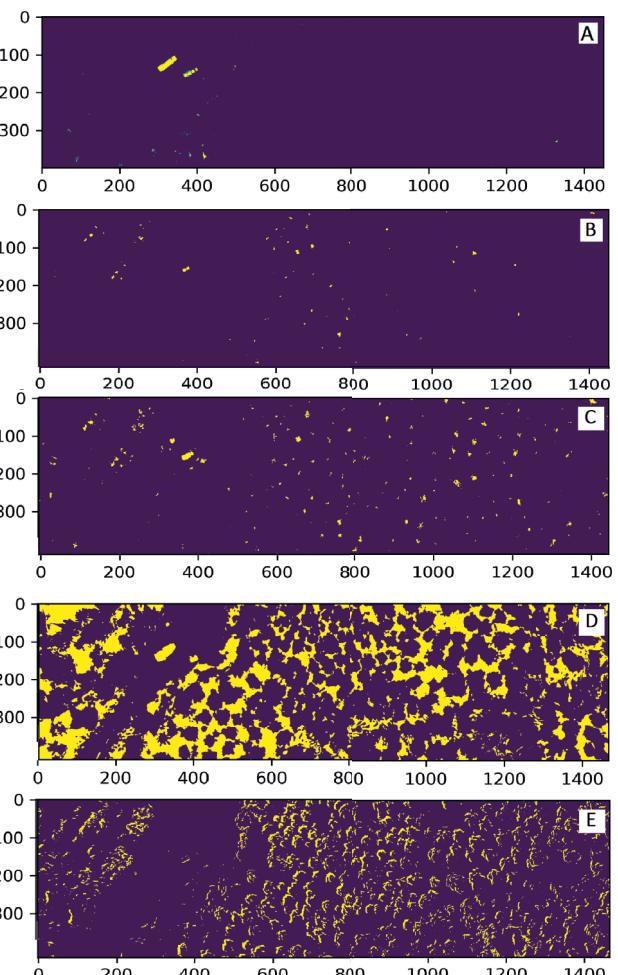


Figure 11: The anomaly detection maps of the forest data. A: MLM, implemented with cosine distance metric, using variance and one threshold (Table 3, image A). The size of the X was 2500, and R was 250. B: One-Class-SVM, C: Isolation forest, D: Global RX, E: Local RX. The anomaly maps of the reference methods are originally from the (Pöllönen et al., 2020).

In this study, estimating whether the pixel is anomalous was carried out at individual pixel level. However, for the future, it is possible to improve these methods to include the pixel neighbours for the estimation. Another interesting idea to explore is the sensitivity analysis for anomalies that are smaller than one pixel. The subpixel level sensitivity studies could reveal the detection rates for anomalies that, for example, can be seen among the trees in Fig. 9.

Since the MLM seems to perform well in the HS image anomaly detection, the answer to the question, which MLM version to select for the anomaly detection, depends on the data. If the HS image is large, moreover, if it contains both local and global anomalies, and the noise levels are unknown or high; the piecewise MLM would be the method to use. If the data is less noisy, the selection would be the MLM. The MLM can detect both local and global anomalies, and it is faster than the piecewise MLM,

but the piecewise MLM might be an easier method to use since it is not so sensitive with the threshold values and it is more robust with respect to noise.

5. CONCLUSIONS

The MLM anomaly detection method can be extended to detect both local and global anomalies from the hyperspectral images, but the method's accuracy is highly sensitive with respect to noise. The piecewise MLM is a new approach, which can perform similar anomaly detection tasks from the hyperspectral data, being significantly more robust with respect to the noise. The MLM or piecewise MLM approach should be selected depending on whether the data contains low or high noise levels. If the data is less noisy, the MLM would be a computationally effective solution for anomaly detection. If the data contains higher levels of noise, the choice would be the piecewise MLM. Both of these methods are easy to implement, and they can be used in real-time hyperspectral anomaly detection applications. The methods can be implemented with small, single-board computers.

ACKNOWLEDGEMENTS

This study is partly funded by the Academy of Finland (Grant No. 327862). We thank our colleague Kimmo Riihiaho for assistance and comments that improved the manuscript.

REFERENCES

- Bioucas-Dias, J. M., Plaza, A., Camps-Valls, G., Scheunders, P., Nasrabadi, N. M. and Chanussot, J., 2013. Hyperspectral remote sensing data analysis and future challenges. *IEEE Geoscience and Remote Sensing Magazine* 1(2), pp. 6–36.
- Caba, J., Díaz, M., Barba, J., Guerra, R., de la Torre, J. A. and López, S., 2020. Fpga-based on-board hyperspectral imaging compression: Benchmarking performance and energy efficiency against gpu implementations. *Remote Sensing* 12(22), pp. 1–37.
- Camps-Valls, G. and Bruzzone, L., 2005. Kernel-based methods for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing* 43(6), pp. 1351–1362.
- Che, W., Sun, L., Zhang, Q., Tan, W., Ye, D., Zhang, D. and Liu, Y., 2018. Pixel based bruise region extraction of apple using Vis-NIR hyperspectral imaging. *Computers and Electronics in Agriculture* 146(July 2017), pp. 12–21.
- Chen, C., Li, W., Gao, L., Li, H. and Plaza, J., 2018. Special issue on advances in real-time image processing for remote sensing. *Journal of Real-Time Image Processing* 15(3), pp. 435–438.
- De Carvalho, O. A. and Meneses, P. R., 2000. Spectral correlation mapper (scm): an improvement on the spectral angle mapper (sam). In: Summaries of the 9th JPL Airborne Earth Science Workshop, JPL Publication 00-18, Vol. 9, JPL Publication Pasadena, CA.
- de Souza, A. H., Corona, F., Barreto, G. A., Miche, Y. and Lendasse, A., 2015. Minimal Learning Machine: A novel supervised distance-based approach for regression and classification. *Neurocomputing* 164(January), pp. 34–44.
- Hakola, A.-M. and Pölönen, I., 2020. Minimal learning machine in hyperspectral imaging classification. In: L. B. Santi, F. Bovolo and Emanuele (eds), *Image and Signal Processing for Remote Sensing XXVI*, Vol. 11533, p. 26.
- Haut, J. M., Paoletti, M. E., Plaza, J. and Plaza, A., 2018. Fast dimensionality reduction and classification of hyperspectral images with extreme learning machines. *Journal of Real-Time Image Processing* 15(3), pp. 439–462.
- Honkavaara, E., Saari, H., Kaivosoja, J., Pölönen, I., Hakala, T., Litkey, P., Mäkinen, J. and Pesonen, L., 2013. Processing and assessment of spectrometric, stereoscopic imagery collected using a lightweight UAV spectral camera for precision agriculture. *Remote Sensing* 5(10), pp. 5006–5039.
- Nevalainen, O., Honkavaara, E., Tuominen, S., Viljanen, N., Hakala, T., Yu, X., Hyypä, J., Saari, H., Pölönen, I., Imai, N. N. and Tommaselli, A. M., 2017. Individual tree detection and classification with UAV-Based photogrammetric point clouds and hyperspectral imaging. *Remote Sensing*.
- Nezami, S., Khoramshahi, E., Nevalainen, O., Pölönen, I. and Honkavaara, E., 2020. Tree species classification of drone hyperspectral and RGB imagery with deep learning convolutional neural networks. *Remote Sensing*.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M. and Duchesnay, E., 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* 12, pp. 2825–2830.
- Polonen, I., Annala, L., Rahkonen, S., Nevalainen, O., Honkavaara, E., Tuominen, S., Viljanen, N. and Hakala, T., 2018. Tree Species Identification Using 3D Spectral Data and 3D Convolutional Neural Network. *Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing*.
- Pölönen, I., Riihiaho, K., Hakola, A.-M. and Annala, L., 2020. Minimal learning machine in anomaly detection from hyperspectral images. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives* 43(B3), pp. 467–472.
- Prasad, N. R., Almanza-Garcia, S. and Lu, T. T., 2009. Anomaly detection. *Computers, Materials and Continua* 14(1), pp. 1–22.
- Saari, H., Pölönen, I., Salo, H., Honkavaara, E., Hakala, T., Holmlund, C., Mäkinen, J., Mannila, R., Antila, T. and Akujärvi, A., 2013. Miniaturized hyperspectral imager calibration and UAV flight campaigns. *Sensors, Systems, and Next-Generation Satellites XVII* 8889(October 2013), pp. 88891O.
- Varoquaux, G., Buitinck, L., Louppe, G., Grisel, O., Pedregosa, F. and Mueller, A., 2015. Scikit-learn: GetMobile: Mobile Computing and Communications 19(1), pp. 29–33.
- Yuhas, R. H., Goetz, A. F. H. and Boardman, J. W., 1992. Discrimination among semi-arid landscape endmembers using the spectral angle mapper (sam) algorithm. In: *Summaries of the Third Annual JPL Airborne Geoscience Workshop*, Vol. 1, pp. 147–149.

PIII

UPDATING STRATEGIES FOR DISTANCE BASED CLASSIFICATION MODEL WITH RECURSIVE LEAST SQUARES

by

Raita-Hakola Anna-Maria and Pölönen Ilkka 2022

ISPRS - Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, <https://doi.org/10.5194/isprs-annals-V-3-2022-163-2022>

Reproduced with kind permission of ISPRS.

UPDATING STRATEGIES FOR DISTANCE BASED CLASSIFICATION MODEL WITH RECURSIVE LEAST SQUARES

Anna-Maria Raita-Hakola^{a,*}, Ilkka Pöölönen^a,

^a Faculty of Information Technology, University of Jyväskylä, 40100, Jyväskylä, Finland -
(anna.m.hakola, ilkka.polonen)@jyu.fi

Commission III, WG III/4

KEY WORDS: Hyperspectral imaging, Minimal Learning Machine, Recursive Least Squares, Classification, Real-time computation, Machine learning

ABSTRACT:

The idea is to create a self-learning Minimal Learning Machine (MLM) model that is computationally efficient, easy to implement and performs with high accuracy. The study has two hypotheses. Experiment A examines the possibilities of introducing new classes with Recursive Least Squares (RLS) updates for the pre-trained self learning-MLM model. The idea of experiment B is to simulate the push broom spectral imagers working principles, update and test the model based on a stream of pixel spectrum lines on a continuous scanning process. Experiment B aims to train the model with a significantly small amount of labelled reference points and update it continuously with (RLS) to reach maximum classification accuracy quickly.

The results show that the new self-learning MLM method can classify new classes with RLS update but with a cost of decreasing accuracy. With a larger amount of reference points, one class can be introduced with reasonable accuracy. The results of experiment B indicate that self-learning MLM can be trained with a few reference points, and the self-learning model quickly reaches accuracy results comparable with nearest-neighbour NN-MLM. It seems that the self-learning MLM could be a comparable machine learning method for the application of hyperspectral imaging and remote sensing.

1. INTRODUCTION

For remote sensing and Earth observation, hyperspectral imaging (HS Imaging) provides powerful tools for a variety of interesting applications, such as monitoring vegetation, environmental parameters, agricultural and forestry phenomena (Adão et al., 2017, Tuominen et al., 2018, Honkavaara et al., 2013, Nevalainen et al., 2017, Nezami et al., 2020). Due to the continuous HS imaging technology development, the sizes of the HS imagers has been decreasing, and the HS imaging sensors have become more affordable. So far, small and lightweight HS imagers can be applied to unmanned aerial systems (UAS), which are popular and cost-effective platforms in remote sensing (Adão et al., 2017).

The HS data is highly dimensional, offering high accuracy and robustness for identification and characterisation tasks (Camps-Valls and Bruzzone, 2005, Bioucas-Dias et al., 2013). The strengths and weaknesses of HS imaging in remote sensing applications arise from the essence of the data. Each (HS) image can be considered as a stack of frames, each representing an intensity of a different wavelength of light. The data contains spatial and spectral information, and each pixel has its spectrum, which is a strength compared to other less informative imaging methods. The downside of remote sensing HS imaging applications is a large amount of data due to the spectral channels. Other challenges can be the platform's payload and processing constraints, restricted available energy and the complexity of the machine learning methods (Haut et al., 2018, Caba et al., 2020). Thus, computationally efficient, easy to implement and adaptive machine learning methods are needed.

Minimal Learning Machine (MLM) is a distance-based supervised method for classification, regression and anomaly detection, which performance is comparable to most state-of-the-art

methods (Mesquita et al., 2017, Pöölönen et al., 2020, Raita-Hakola and Pöölönen, 2021, Hakola and Pöölönen, 2020). MLM is easy to implement. Its learning phase consists of building a linear mapping between input and output distance matrices. MLM uses this model in the generalisation phase for estimating the distances from the new output reference points to the target output values. In classification, the output estimation can be seen as an optimisation problem or, we can use the MLM Nearest neighbour (NN-MLM) variation, a computationally efficient alternative. The efficiency is reached by using a simple linear search over a set of distance estimations (Mesquita et al., 2017).

Since the core of this method is the linear mapping, a model that can be solved with the Ordinary Least Squares estimator (OLS), it is an interesting idea to study if it is possible to create a new variation from the MLM, which updates itself with Recursive Least Squares (RLS) algorithm. This study examines the MLM with RLS, aiming to test two hypotheses.

Hypothesis A is that we can train new classes to the MLM model with new data and RLS algorithm without re-training the whole model. Since the MLM needs the whole training set X only at the beginning and operates with the significantly smaller subset R . Thus, it is interesting to increase the amounts of classes by introducing the new classes to the model with a new set of training points. The new training points are only a portion of the subset R , which ensures the model is computationally efficient.

Hypothesis B: it is possible to increase the accuracy of the MLM classifier by updating the pre-trained model with new information and a recursive least squares estimator. We can use only a few reference points for the first-round classifier and increase the accuracy with the model updates. This way, we could create a model that could learn from the new data and adapt the classifier better for changing targets, such as forest types or land covers.

*Corresponding author.

The hypothesis is that the MLM model can be updated without re-training the whole model, and the new self-learning MLM can adapt to its new data for increased accuracy. This kind of machine learning method could be suitable, for example, for real-time push broom HS imaging applications. In this study, we call this new approach to self-learning MLM.

The paper is organised as follows. Section 2. describes the methods and the selected hyperspectral data. The results are introduced in section 3.. The discussion is the fourth section, and the final conclusions are in the section 5..

2. MATERIAL AND METHODS

2.1 Minimal Learning Machine

The Minimal Learning Machine offers computationally efficient solutions for classification and anomaly detection (Pöllönen et al., 2020, Raita-Hakola and Pöllönen, 2021) applications. It is a computationally efficient and fast machine learning method with the basic idea of utilising the mapping between the input and output distances (de Souza et al., 2015).

Let $X \subset \mathbb{R}^d$ be a training set consisting from spectra and $\mathbf{x}_i \in X$ a single spectrum i . Also, let $R \subset X$ be a set of reference points and $\mathbf{m}_k \in R$ a single spectrum k . Corresponding, the training set ground truth labels are $\mathbf{y}_i \in Y \subset \mathbb{R}$ and for the subset of Y , labels $\mathbf{t}_k \in T$. The training set X has N samples, and the subset R size is K . Now, the distances for the linear mapping are $d(\mathbf{x}_i, \mathbf{m}_k)$ and $\delta(\mathbf{y}_i, \mathbf{t}_k)$.

When T and R are selected, we can define two distance matrices $\Delta_y \in \mathbb{R}^{N \times K}$ and $\mathbf{D}_x \in \mathbb{R}^{N \times K}$ using previous mappings d and δ . By assuming the linear mapping between these two distance matrices, we have a linear model:

$$\Delta_y = \mathbf{D}_x \mathbf{B} + \mathbf{E}, \quad (1)$$

where \mathbf{B} is coefficients and \mathbf{E} is the residual. We can approximate the coefficients \mathbf{B} with the Ordinary Least Squares estimator (OLS) (Mesquita et al., 2017)

$$\hat{\mathbf{B}} = (\mathbf{D}_x^T \mathbf{D}_x)^{-1} \mathbf{D}_x^T \Delta_y. \quad (2)$$

As a result, we have a $\hat{\mathbf{B}}$, which is the linear model between the input distances $d(\mathbf{x}_i, \mathbf{m}_k)$ and output distances $\delta(\mathbf{y}_i, \mathbf{t}_k)$.

Now, for the new spectrum \mathbf{x}_n the distance between its label y_n and set T is

$$\delta(y_n, T) = d(\mathbf{x}_n, R)\hat{\mathbf{B}}. \quad (3)$$

Label y_n can be estimated by solving an quadratic optimisation problem

$$\min_{y_n} \sum_{k=1}^K \left(y_n - \mathbf{t}_k \right)^T \left(y_n - \mathbf{t}_k \right) - \delta^2(y_n, T). \quad (4)$$

The classification (equation 4) can be seen as multilateration problem, where quadratic optimisation can be used (de Souza et al., 2015). It is also possible to use Tikhonov regularisation to produce more robust models (Kärkkäinen, 2019). Besides the optimisation and regularisation possibilities, we can utilise the distances and consider the model $\hat{\mathbf{B}}$ to be a generalisation of the nearest neighbour classifier (NN-MLM) (Mesquita et al., 2017). Equation 3 gives us distances, and for classifying the \mathbf{x}_n , we perform an argument sorting for $\delta(y_n, T)$ and assign the N-closest label value from T . The detailed implementation of the nearest neighbour generalisation is in (Hakola and Pöllönen, 2020).

2.2 Recursive least squares approach

Ordinary Least Squares (OLS), used in the MLM (equation 2), is meant for static situations. However, the OLS can be updated continuously with new data, using a Recursive Least Squares (RLS) algorithm. The updating is computationally efficient because it does not need the original data. The RLS is based on a matrix inversion lemma called the Sherman-Morrison-Woodbury equation (Meyer, 2000). According to this lemma, it is possible to update a linear equation system solution efficiently. Detailed proof and equations can be found from (Romberg, 2016, Haykin, 2008). Algorithm 1 describes the RLS updating phases, which are implemented and utilised in our following self-learning MLM model experiments.

Algorithm 1: Recursive Least Squares (Romberg, 2016)

Input: X, R, Y, T
Initialize:
 Calculate distance matrix $\mathbf{D}_{x,0}$
 Calculate distance matrix $\Delta_{y,0}$
 Calculate $\mathbf{P}_0 = (\mathbf{D}_{x,0}^T \mathbf{D}_{x,0})^{-1}$
 Calculate model $\hat{\mathbf{B}}_{x,0} = \mathbf{P}_0 \mathbf{D}_{x,0}^T \Delta_{y,0}$
for $i = 1, 2, 3, \dots$ **do**
 New data X_i, Y_i appears
 Calculate distance matrix $\mathbf{D}_{x,i}$
 Calculate distance matrix $\Delta_{y,i}$
 Calculate $\mathbf{P}_{x,i} = \mathbf{P}_{i-1} \mathbf{D}_{x,i}$
 Calculate $\mathbf{P}_i =$

$$\mathbf{P}_{i-1} - \mathbf{P}_{x,i} (\mathbf{I} + \mathbf{D}_{x,i} \mathbf{P}_{x,i})^{-1} \mathbf{D}_{x,i} \mathbf{P}_{i-1}$$
 Calculate $\mathbf{K}_i = \mathbf{P}_i \mathbf{D}_{x,i}^T$
 Update model $\hat{\mathbf{B}}_{x,i} =$

$$\hat{\mathbf{B}}_{x,i-1} + \mathbf{K}_i (\Delta_{y,i} - \mathbf{D}_{x,i} \hat{\mathbf{B}}_{x,i-1})$$
end for

2.2.1 Self-learning MLM: Experiment A In experiment A, we trained and updated the model in phases. The aim was to train the model only with a few classes and then use RLS updates to introduce new classes.

At first, we selected an equal amount (*class-wise R size*) of data points to R and their labels T from each of the first four classes. The first model was trained on the initialization phase in Algorithm 1.

After the initialization, the Algorithm 1 got new data X_i, Y_i . The size of the new data was the *class-wise R*, and it contained samples only from a new class. The model was updated and tested with test data containing the previous and new data class samples. The updated model provided the new distances $\delta(y_n, T)$. Since the T contains only samples from classes one to four, we needed to alter how we selected new classes. At first, we classified the data with T , and nearest neighbour method (Hakola and Pöllönen, 2020). Classes from one to four got their labels.

Then we used classification rules shown in Fig. 1. As we know the new datapoint distances to the reference data points $\delta(y_n, T)$, we can re-check the predictions that had the biggest class label four (4). The distance from a new label to the first label is longer than the longest distance of the last known biggest label four (4). We can see from Fig. 1 that the Euclidean distance from label four (4) to label four (4) is 0. The longest known distance from label four (4) to label one (1) is 3. Therefore, the distance from a new label five (5) to label one (1) is more than 3.5.

Based on the known distances, the halfway limit was set to 0.5. With this limit, we used a majority voting rule: if more than

80% (calculated from the *class-wise R size*) of the distances from new data point to reference points are longer than the last round's *biggest class* -0.5, the point is classified to a new class, which is the last round's *biggest class* + 1. For every new class, we computed the classes similarly, updating the *biggest class* in a loop, until all of the new classes were re-classified.

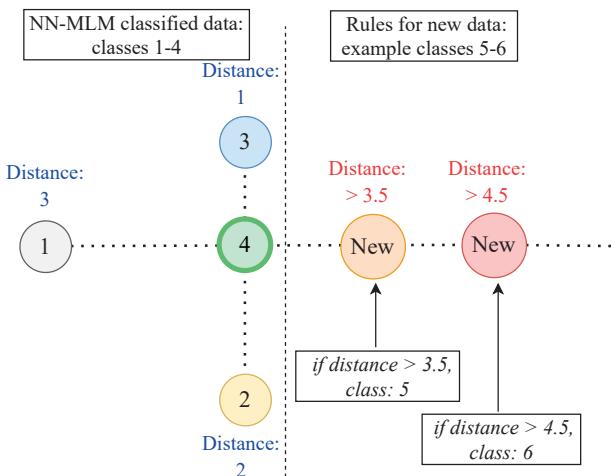


Figure 1: Classification rules for the new classes. Classes 1-4 were classified using the NN-MLM. The distances from the biggest known class four (4) are written in blue. The image show the first updating rules. The longest distance of the class four (4) is 3, all distances longer than 3.5 were re-classified to a bigger class. The biggest class was updated and again, the longest distance was set to 4.5 (5 - 0.5), and the data was re-checked. Each time a new class appeared, all of the biggest class datapoints were re-checked with this halfway limit as many times as we had new classes.

2.2.2 Self-learning MLM: Experiment B This experiment *B* aims to simulate the push broom spectral imagers working principles, update and test the model based on a stream of pixel spectrum lines on a continuous scanning process. We will limit the amount of the reference points *R* and the sizes of the distance matrices \mathbf{D}_x and Δ_y , train the first model and update it with new data and its new ground truth, labelled with the previous model, row-by-row.

Usually, machine learning methods need a lot of training data. MLM differs, it is a method that uses only a subset (*R*) of the original data *X*, and therefore it is computationally efficient. The first model can be trained with only a few known labelled pixel spectra in this experiment. Instead of using the *X* and *Y* in distance calculations, the first model was trained (Algorithm 1, initialize) using only the distances between *R* to *R* and *T* to *T*. This reduces the number of distance calculations and decrease the distance matrices sizes.

The first model was given to the RLS update loop. There were 41 rows in the training and testing datasets. For each row (image A, Fig. 4), the model was updated and tested. Each time the given new data \mathbb{X}_i got an updated ground truth \mathbb{Y}_i . For the updated ground truth, we classified the new training points \mathbb{X}_i with a previous model $\hat{\mathbf{B}}_{i-1}$. As a result, the model's behaviour is self-learning since each time it updates itself, it specifies and updates its ground truth knowledge based on the current update and uses it for the next model update, reaching for increasing accuracy.

We run tests for each row. In row-by-row testing (RBR), the test data $\mathbb{X}_{test,i}$, $\mathbb{Y}_{test,i}$ was a new row, top-down from the visu-

alised image B Fig. 4. Besides the RBR testing, each updated model was tested with whole test data (TDR testing). The average number of pixels (each with 103 spectral channels) in an RBR's training and test row was 67 since the unlabelled pixels were removed from the data.

Experiments were implemented using Scikit-Learn (Pedregosa et al., 2011) and SciPy (Virtanen et al., 2020) Python libraries. The computing was performed with 28 core Linux server for non-parallel computing, x86_64.

2.3 Hyperspectral data

The experiments were done with two known hyperspectral images, Pavia University and Salinas A scene. The data is provided by the Grupo De Intelligencia Computational (GIC)(Graña et al., 2022). The Pavia Centre HS image is acquired by the ROSIS sensor with 103 spectral channels with the spatial resolution of 1.3 meters and with a spectrum coverage ranging from 430 to 860 nm (Xie et al., 2019). The image size is 1096 x 1096 pixels. The image ground truth (Fig. 2) is divided into 9 classes (Graña et al., 2022).

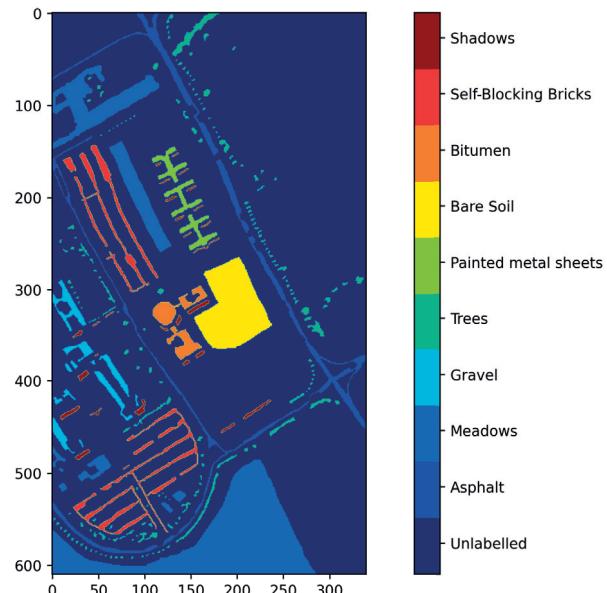


Figure 2: Visualization of the Pavia University ground truth. The image size is 610 x 340 pixels with 103 spectral channels.

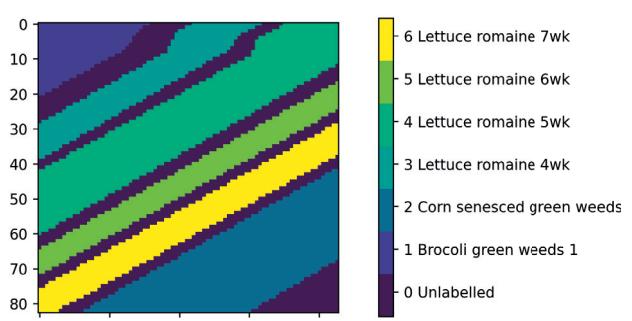


Figure 3: Visualization of the Salinas A scene ground truth. The image size is 83 x 86 pixels with 204 spectral channels.

We used the Salinas A scene collected with an AVIRIS sensor in the self-learning MLM *B* experiment. The image size is 83 x 86 pixels, it has 204 spectral bands, and the labels are divided into six classes (Fig. 3). The spatial resolution is 3.7 m over the range of 400–2500 nm (Sawant and Manoharan, 2020).

Both datasets were normalised between 0 and 1, and the unlabelled pixels were removed from the training and testing data. After the preprocessing, the datasets were divided into training and testing portions as follows. The detailed information of these datasets can be found from (Graña et al., 2022).

2.3.1 Pavia University, experiment A The Pavia dataset was divided class-wise with 50:50 portions to training and test datasets. After the split, the data was randomised. The idea of this dividing-and pre-processing logic enabled was the introduction of the new classes for the self-learning MLM model. This way, we could easily train the first model with four first classes and test it with four class customised test data. During the next rounds, we could introduce only new class data points from the training data for the model and test the updated model with a new customised test set, including the old and new classes. After removing the unlabelled data, the training data size was 21391 and the test data 21385 pixels with 103 spectral channels.

2.3.2 Salinas A scene, experiment B The Salinas dataset was divided horizontally top-down with portions of 50:50. Every other row to test data, every other to training data. Fig. 4 visualises the training (A) and test data (B) on frame 144, representing wavelength 1890 nm. After removing the unlabelled data, the training data size was 2707 and the test data 2641 pixels with 204 spectral channels.

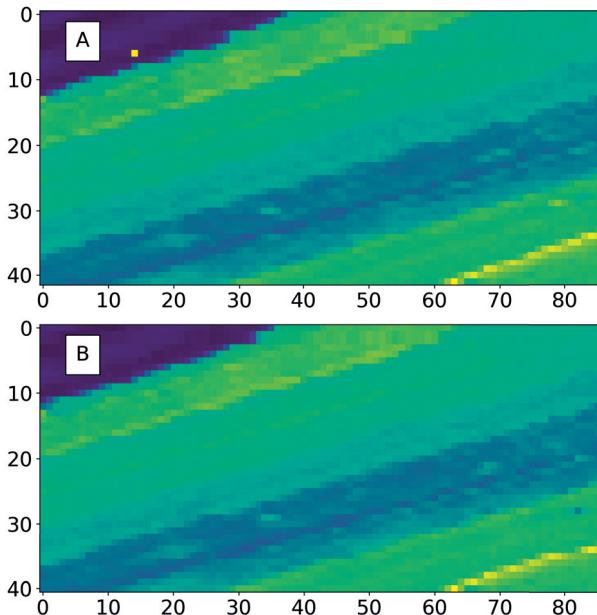


Figure 4: Salinas test and training data. Images A and B represent Salina's frame number 144 (wavelength 1890 nm) after the slicing. The dataset was divided horizontally, top-down. Every other row to training data (A), every other to test data (B).

After selecting the equal amount of the reference points from the randomised test data class-wise, the self-learning MLM model utilised the training data row by row, top-down. The accuracy of the model was tested row-by-row (RBR) with test row (next index from the original data) and with the whole test data (TDR).

3. RESULTS

3.1 Experiment A

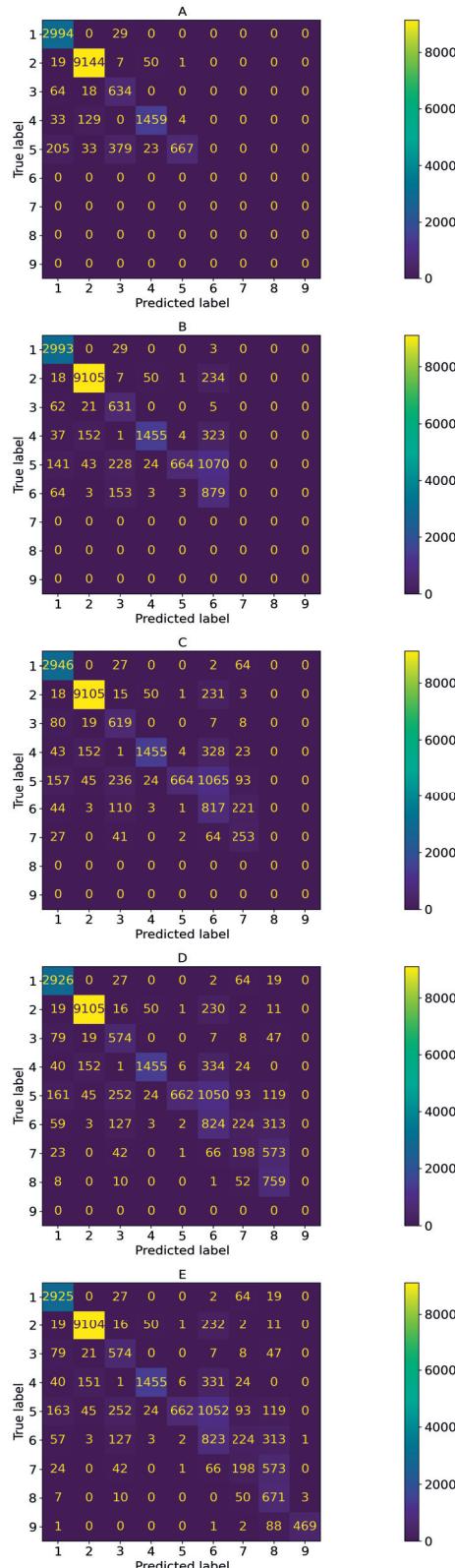


Figure 5: Experiment A confusion matrices. The images visualise the classification results after the new classes are introduced to the self-learning MLM. Image A new class: 5, B new class: 6, C new class: 7, D: new class 8, E: new class 9.

In experiment A, we introduced new classes to the self-learning MLM. Fig. 5 visualises the confusion matrices of the classification results. Images A-E show (Fig. 5) how the self-learning MLM classifies the first model's classes (1-4) and the rest of classes 5-9. Image A has classes 1-5, B 1-6 until finally, image E has all of the Pavia dataset's nine classes.

The self-learning MLM classifier can recognise and classify new classes with RLS updates and classification rules from Fig. 1, but with a cost of decreasing accuracy. Table 1 show that the, and the R size affects the results, increasing the accuracy, but the newly introduced classes are decreasing it.

Class-wise R size	5	6	7	8	9
100	0.90	0.80	0.78	0.74	0.73
200	0.92	0.82	0.80	0.76	0.76
500	0.94	0.85	0.83	0.79	0.79
800	0.94	0.86	0.84	0.80	0.80
1000	0.94	0.87	0.85	0.80	0.80

Table 1: Experiment A accuracy results for each new class 5-9. Class R size: number of training points selected per class.

3.2 Experiment B

Experiment B results in table 2 show that the self-learning MLM is reaching comparable accuracy against the once trained MLM (OTM).

Model	Class R, N.N.	Max	Min	Mean	Mode
RBR	10,3	1.00	0.86	0.97	1.00
TDR	10,3	0.97	0.95	0.96	0.96
OTM	10,3	0.98	0.98	0.98	0.98
RBR	10,5	1.00	0.86	0.97	1.00
TDR	10,5	0.97	0.95	0.96	0.96
OTM	10,5	0.98	0.98	0.98	0.98
RBR	10,10	1.0	0.86	0.97	1.00
TDR	10,10	0.97	0.95	0.96	0.96
OTM	10,10	0.98	0.98	0.98	0.98
RBR	20,3	1.00	0.89	0.98	1.00
TDR	20,3	0.98	0.97	0.98	0.98
OTM	20,3	0.99	0.99	0.99	0.99
RBR	20,5	1.00	0.89	0.98	1.00
TDR	20,5	0.98	0.97	0.98	0.98
OTM	20,5	0.99	0.99	0.99	0.99
RBR	20,10	1.0	0.89	0.98	1.00
TDR	20,10	0.98	0.97	0.98	0.98
OTM	20,10	0.99	0.99	0.99	0.99
RBR	100,3	1.00	0.94	0.99	1.00
TDR	100,3	0.99	0.99	0.99	0.99
OTM	100,3	0.99	0.99	0.99	0.99
RBR	100,5	1.00	0.94	0.99	1.00
TDR	100,5	0.99	0.99	0.99	0.99
OTM	100,5	0.99	0.99	0.99	0.99
RBR	100,10	1.0	0.94	0.99	1.00
TDR	100,10	0.99	0.99	0.99	0.99
OTM	100,10	0.99	0.99	0.99	0.99

Table 2: Maximum, minimum, mean and mode values of the accuracy. RBR: row-by-row updated model, one-row test data accuracy, TDR: row-by-row updated model, whole test data row-by-row model accuracy, OTM: once trained MLM model. Class R: number of selected training reference points per class, N.N.: number of neighbours.

The row-by-row results (RBR) in Table 2 and Fig. 6 has the most variation. The single RBR row accuracy result might be weak, but

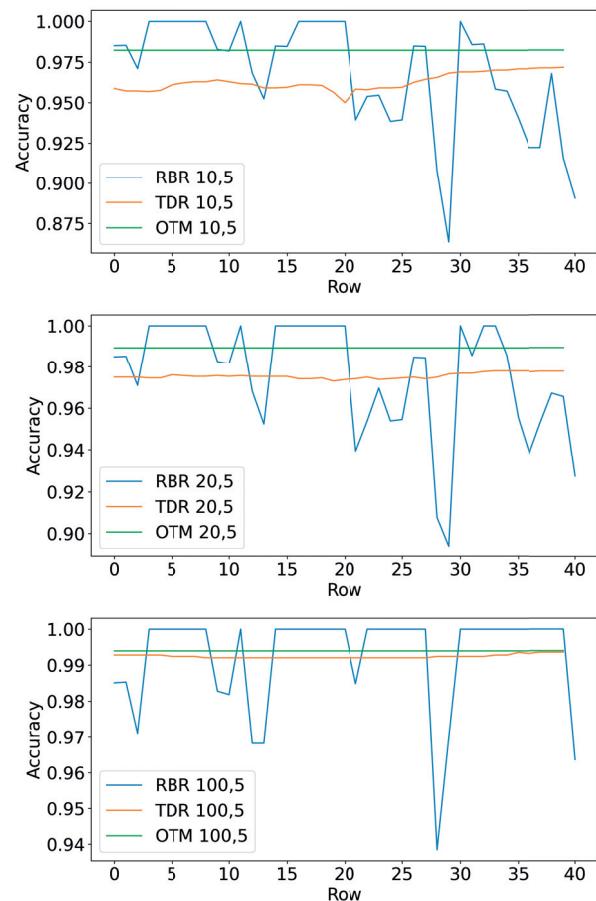


Figure 6: Visualisation of the self-learning MLM accuracy results. RBR: self-learning MLM, one-row accuracy, test data is original image's next row from the current update row. TDR: self-learning MLM, the accuracy of the whole classified test data, measured after very row-by-row model updates. OTM: once trained MLM model, tested with the whole test data. First image: *classwise R size*: 10, number of neighbours: 5, second image *classwise R size*: 20, number of neighbours: 5, third image *classwise R size*: 100, number of neighbours: 5.

the overall effect of the single row low performance is reasonable for the whole model. For example, the self-learning MLM model that was trained with 10 as a *classwise R size* had the lowest single row accuracy of 86%. The same model had a mean accuracy of 97%, and the mode of the one-row pixel-wise classification accuracy results was 100%. The model was also tested row-by-row with the whole test data (TDR), reaching the mean accuracy of 96%.

In Fig. 6, we can see the accuracy results row by row. RBR, TDR (self-learning model, tested after every row with whole test data), and OTM (once trained MLM) models had the same *classwise R size* and the number of neighbours. The effect of the increasing accuracy of the *classwise R size* for the self-learning MLM can be seen in Fig. 8, which visualises the accuracy results calculated row-by-row from the classified whole test data. The number of neighbours seemed to be a parameter that did not affect the results of this experiment.

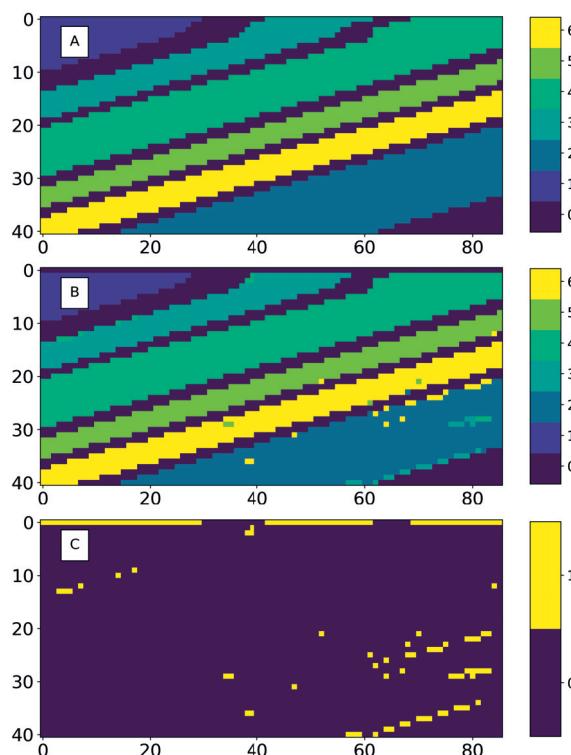


Figure 7: A: Salinas test data ground truth, B: Self-learning MLM's top-down, row by row (RBR) classification map, C: Row-by-row classification error map.

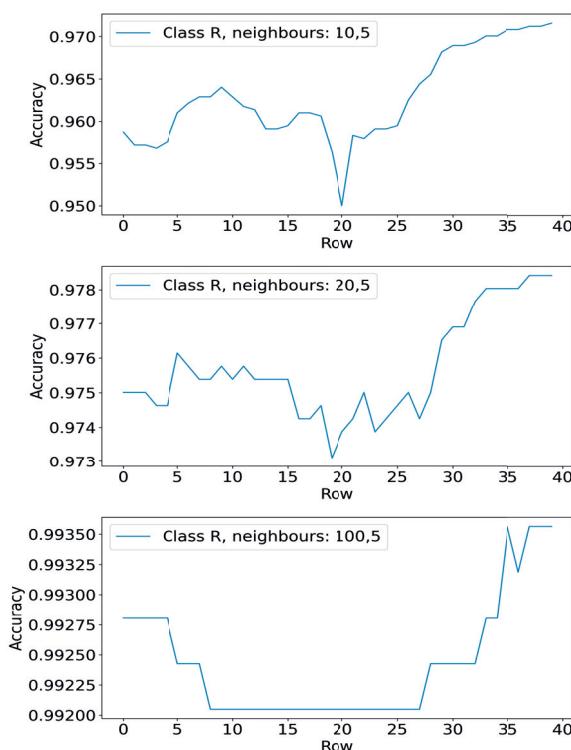


Figure 8: Self-learning MLM accuracy results row-by-row, using each time the whole test data.

Self-learning MLM's row-by-row classification results can be seen

on the classification map (Fig. 7). Image A is the ground truth of the test data. B visualises the results that are classified row-by-row from top to down. Between every row shown in the image, the model is updated with one row of new training data (training and test data are visualised in Fig. 4. The labels for the new training data are produced with the previous updated model. Image C in Fig. 7 shows the classification errors of the RBR results. Yellow represents miss-classification; purple is correct.

4. DISCUSSION

The results of the self-learning MLM experiment A show that the model can handle new classes after the RLS updates, but the accuracy decreases with every appearing new class. Similarly, with the previous research, when the size of R approaches the size of X , the accuracy of the results improves (de Souza et al., 2015). The number of samples in the Pavia dataset was not classwise equal; for example, class seven (7) had only 476 samples, which might have affected the results since the maximum of the updating phase's *classwise R size* in that class was 238.

In our experiment, the number of the training samples in the reference set R had a positive effect on the results, which means that with a larger amount of training data, the first new class can be updated to the model with reasonable accuracy (Table 1, Fig. 5).

However, for introducing several new classes, we would recommend either improving the classification method 1, using the optimization methods (de Souza et al., 2015, Kärkkäinen, 2019) or considering training a new model with accurate ground truth after every n :th update rounds. Further research is needed for increasing the accuracy of introducing multiple new classes to the self-learning MLM.

Self-learning MLM experiment B seemed to reach our aim of creating a method for a model that can be trained with a low number of samples, and it will adapt itself to the data stream. Originally, the MLM training distances were the distances between the training set X , subset R , and the corresponding label distances Y and T (de Souza et al., 2015). Our once trained model (OTM) calculated distances from each point of the R to each point of the X . The self learning-MLM calculated the input distances from R to R and output distances from T to T .

As an example, we had six (6) classes, and the *classwise R size* was 20. OTM calculated the distances from 120 reference points to all 2707 data points (and similarly the label distances). Our approach calculated the distances using only the reference points R and their labels T , which means distances from the 120 points to 120 data points and similarly with the labels. As a starting point, the first model can be trained with 20 or 100 pixel spectra per class, and it can reach the accuracy of the OTM model (98–99%) by updating and learning from the incoming data stream. From the computational performance perspective, since we calculate the distances row-by-row, the self-learning MLM is close to the NN-MLM classifier even though the starting point is less demanding.

One of the MLM's benefits against many other machine learning models is the low number of hyperparameters. A comparable NN-MLM can be trained, depending on the classification task, with three hyperparameters: the number of neighbours, the R size and distance calculation method (Hakola and Pöllönen, 2020). We selected the Euclidian distance method based on the previous study of MLM classifier, and hyperspectral data (Hakola and Pöllönen, 2020). In that study, the number of neighbours affected the accuracy. Self-learning MLM seems to be more ignorant on

the number of neighbours. The most effective hyperparameter was the *classwise R size*. In experiment A, the *R* size had a more significant role on the results, where the experiment B (Table 2) show that the self-learning MLM can reach high accuracy without a large number of training data, using only few class-wise spectra as a reference data.

The row-by-row approach was semi-sensitive to the quality of the incoming data. Fig. 6 shows how the RBR accuracy varies depending on the single training or testing row of pixel spectra. This variation seems to be a phenomenon with only a small effect on the results conducted using the same row-by-row models with the whole test data between the updating and learning processes.

RBR classification error map (Fig. 7, image C) reveals the miss-classified pixels. Visual evaluation over the spectral channels of the training and test data shows that there might be some anomalous pixel spectra that differ from the other similarly annotated data around them. For example, the yellow line on the right bottom corner and blue dots on the green area on top of it in image B (Fig. 4) can be seen on spectral channel 1890 nm. Similar features can be seen on the classification error map but not in the ground truth image (Fig. 3). That is an interesting point of view for further studies. The question could be how sensitive the self-learning MLM is towards the possible anomalous pixels. Could the implementation in B be extended for anomaly detection, using the variance and thresholds method with the output distances $\delta(y_i, t_k)$, as seen in (Raita-Hakola and Pölönen, 2021).

5. CONCLUSIONS

There is a need for fast, accurate and computationally efficient machine learning methods in hyperspectral imaging remote sensing applications. In this study, we implemented and tested the nearest neighbour Minimal Learning Machine (NN-MLM) with Recursive Least Squares (RLS) updates, and as a result, we created a new method, the self-learning MLM.

The self-learning method was tested in two experiments. Experiment A confirms that the updated model can recognise and classify new classes. The classification accuracy increase when the training set reference subset's size increases. The limitation to this approach is the number of new classes. The updated model can handle one class relatively well, but the accuracy decreases after every new class and update. Further research is needed for achieving better overall accuracy with multiple new classes.

The second experiment was more successful than experiment A. Experiment B simulated the push broom spectral imagers working principles. The model was updated and tested based on a stream of pixel spectrum lines and a continuous scanning process. The results show that it is possible to train the model with a significantly small amount of labelled reference points and update it continuously with (RLS) to reach quickly high classification accuracy. The accuracy results are comparable with NN-MLM, and the benefit is that the new self-learning model can be first trained even with 20 or 100 reference pixel spectra instead of the whole test data.

ACKNOWLEDGEMENTS

This study is partly funded by the Academy of Finland (Grant No. 327862). We thank our colleague Kimmo Riihiaho for proofreading assistance and comments that improved the manuscript.

REFERENCES

- Adão, T., Hruška, J., Pádua, L., Bessa, J., Peres, E., Morais, R. and Sousa, J. J., 2017. Hyperspectral imaging: A review on uav-based sensors, data processing and applications for agriculture and forestry. *Remote Sensing*.
- Bioucas-Dias, J. M., Plaza, A., Camps-Valls, G., Scheunders, P., Nasrabadi, N. M. and Chanussot, J., 2013. Hyperspectral remote sensing data analysis and future challenges. *IEEE Geoscience and Remote Sensing Magazine* 1(2), pp. 6–36.
- Caba, J., Díaz, M., Barba, J., Guerra, R., de la Torre, J. A. and López, S., 2020. Fpga-based on-board hyperspectral imaging compression: Benchmarking performance and energy efficiency against gpu implementations. *Remote Sensing* 12(22), pp. 1–37.
- Camps-Valls, G. and Bruzzone, L., 2005. Kernel-based methods for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing* 43(6), pp. 1351–1362.
- de Souza, A. H., Corona, F., Barreto, G. A., Miche, Y. and Lendasse, A., 2015. Minimal Learning Machine: A novel supervised distance-based approach for regression and classification. *Neurocomputing* 164(January), pp. 34–44.
- Graña, M., Veganzons, M. and Ayerdi, B., 2022. Hyperspectral Remote Sensing Scenes. Data available in http://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes.
- Hakola, A.-M. and Pölönen, I., 2020. Minimal learning machine in hyperspectral imaging classification. In: L. B. Santi, F. Bovolo and Emanuele (eds), *Image and Signal Processing for Remote Sensing XXVI*, Vol. 11533, p. 26.
- Haut, J. M., Paoletti, M. E., Plaza, J. and Plaza, A., 2018. Fast dimensionality reduction and classification of hyperspectral images with extreme learning machines. *Journal of Real-Time Image Processing* 15(3), pp. 439–462.
- Haykin, S. S., 2008. Adaptive filter theory. Pearson Education India.
- Honkavaara, E., Saari, H., Kaivosa, J., Pölönen, I., Hakala, T., Litkey, P., Mäkinen, J. and Pesonen, L., 2013. Processing and assessment of spectrometric, stereoscopic imagery collected using a lightweight UAV spectral camera for precision agriculture. *Remote Sensing* 5(10), pp. 5006–5039.
- Kärkkäinen, T., 2019. Extreme minimal learning machine: Ridge regression with distance-based basis. *Neurocomputing* 342, pp. 33–48.
- Mesquita, D. P., Gomes, J. P. and Souza Junior, A. H., 2017. Ensemble of Efficient Minimal Learning Machines for Classification and Regression. *Neural Processing Letters* 46(3), pp. 751–766.
- Meyer, C. D., 2000. Matrix analysis and applied linear algebra. Vol. 71, Siam.
- Nevalainen, O., Honkavaara, E., Tuominen, S., Viljanen, N., Hakala, T., Yu, X., Hyppä, J., Saari, H., Pölönen, I., Imai, N. N. and Tommaselli, A. M., 2017. Individual tree detection and classification with UAV-Based photogrammetric point clouds and hyperspectral imaging. *Remote Sensing*.
- Nezami, S., Khoramshahi, E., Nevalainen, O., Pölönen, I. and Honkavaara, E., 2020. Tree species classification of drone hyperspectral and RGB imagery with deep learning convolutional neural networks. *Remote Sensing*.

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

Pöllönen, I., Riihiaho, K., Hakola, A.-M. and Annala, L., 2020. Minimal learning machine in anomaly detection from hyperspectral images. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives* 43(B3), pp. 467–472.

Raita-Hakola, A.-M. and Pöllönen, I., 2021. Piecewise anomaly detection using minimal learning machine for hyperspectral images. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* V-3-2021, pp. 89–96.

Romberg, J., 2016. Ece 6250 fall 2016 lecture notes. Lecture notes available from Georgia Tech University.

Sawant, S. S. and Manoharan, P., 2020. Unsupervised band selection based on weighted information entropy and 3D discrete cosine transform for hyperspectral image classification. *International Journal of Remote Sensing* 41(10), pp. 3948–3969.

Tuominen, S., Näsi, R., Honkavaara, E., Balazs, A., Hakala, T., Viljanen, N., Pöllönen, I., Saari, H. and Ojanen, H., 2018. Assessment of classifiers and remote sensing features of hyperspectral imagery and stereo-photogrammetric point clouds for recognition of tree species in a forest area of high species diversity. *Remote Sensing* 10(5), pp. 1–28.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C. J., Polat, İ., Feng, Y., Moore, E. W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., van Mulbregt, P. and SciPy 1.0 Contributors, 2020. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods* 17, pp. 261–272.

Xie, F., Li, F., Lei, C., Yang, J. and Zhang, Y., 2019. Unsupervised band selection based on artificial bee colony algorithm for hyperspectral image classification. *Applied Soft Computing* Journal 75, pp. 428–440.

PIV

FPI BASED HYPERSPECTRAL IMAGER FOR COMPLEX SKIN SURFACES - CALIBRATION, ILLUMINATION AND APPLICATIONS

by

Raita-Hakola Anna-Maria, Annala Leevi, Lindholm Vivian, Trops Roberts,
Näsilä Antti, Saari Heikki, Ranki Annamari and Pölönen Ilkka 2022

MDPI Sensors, Volume 22, Number 9, Pages 3420,
<https://doi.org/10.3390/s22093420>

Reproduced with kind permission of MDPI.

Article

FPI Based Hyperspectral Imager for the Complex Surfaces—Calibration, Illumination and Applications

Anna-Maria Raita-Hakola ^{1,*}, Leevi Annala ¹, Vivian Lindholm ², Roberts Trops ³, Antti Näsilä ³, Heikki Saari ³, Annamari Ranki ² and Ilkka Pölönen ¹

¹ Faculty of Information Technology, University of Jyväskylä, 40100 Jyväskylä, Finland; leevi.a.annala@jyu.fi (L.A.); ilkka.polonen@jyu.fi (I.P.)

² Department of Dermatology and Allergology, University of Helsinki and Helsinki University Hospital, 00290 Helsinki, Finland; vivian.lindholm@hus.fi (V.L.); annamari.ranki@hus.fi (A.R.)

³ VTT Technical Research Centre of Finland Ltd., 02150 Espoo, Finland; roberts.trops@vtt.fi (R.T.); antti.nasila@vtt.fi (A.N.); heikki.saari@vtt.fi (H.S.)

* Correspondence: anna.m.hakola@jyu.fi

Abstract: Hyperspectral imaging (HSI) applications for biomedical imaging and dermatological applications have been recently under research interest. Medical HSI applications are non-invasive methods with high spatial and spectral resolution. HS imaging can be used to delineate malignant tumours, detect invasions, and classify lesion types. Typical challenges of these applications relate to complex skin surfaces, leaving some skin areas unreachable. In this study, we introduce a novel spectral imaging concept and conduct a clinical pre-test, the findings of which can be used to develop the concept towards a clinical application. The SICSURFIS spectral imager concept combines a piezo-actuated Fabry–Pérot interferometer (FPI) based hyperspectral imager, a specially designed LED module and several sizes of stray light protection cones for reaching and adapting to the complex skin surfaces. The imager is designed for the needs of photometric stereo imaging for providing the skin surface models (3D) for each captured wavelength. The captured HS images contained 33 selected wavelengths (ranging from 477 nm to 891 nm), which were captured simultaneously with accordingly selected LEDs and three specific angles of light. The pre-test results show that the data collected with the new SICSURFIS imager enable the use of the spectral and spatial domains with surface model information. The imager can reach complex skin surfaces. Healthy skin, basal cell carcinomas and intradermal nevi lesions were classified and delineated pixel-wise with promising results, but further studies are needed. The results were obtained with a convolutional neural network.



Citation: Raita-Hakola, A.-M.; Annala, L.; Lindholm, V.; Trops, R.; Näsilä, A.; Saari, H.; Ranki, A.; Pölönen, I. FPI Based Hyperspectral Imager for the Complex Surfaces—Calibration, Illumination and Applications. *Sensors* **2022**, *22*, 3420. <https://doi.org/10.3390/s22093420>

Academic Editors: Robert Martí and Joan Martí Bonmatí

Received: 7 February 2022

Accepted: 23 April 2022

Published: 29 April 2022

Publisher’s Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: hyperspectral; FPI; calibration; interferometry; optical modelling; convolutional neural network; LED illumination; photometric stereo; skin surface model; biomedical imaging; dermatological application; optical biopsy

1. Introduction

Hyperspectral imaging (HSI) systems can be utilized in various sensing applications. The highly dimensional hyperspectral (HS) data offers high accuracy and robustness for identification and characterisation tasks [1,2]. A HS image can be considered a stack of frames, each representing the intensity of a different wavelength of light. Since each spatial pixel has a spectrum, the HS image contains spatial and spectral domains, which enables the accurate pixel-wise classification [3].

Spectral imaging systems were originally applied in remote sensing applications, where systems are robust to rough or irregular topographies, because imaging is performed from the satellites or airplanes. When imaging is performed from a closer range, surface topography and tomography start to affect the image’s spectral quality and homogeneity. Recent advance in hyperspectral sensor imaging has made sensors smaller [4]. Using the

Fabry–Pérot interferometer (FPI) as a spectral separator, the imager size can be reduced without bargaining from spatial or spectral resolution.

HS imaging is proven to be a powerful tool for detecting and identifying diseases in medical research [5–7]. In biomedical imaging, one potential area is dermatological applications. With hyperspectral imaging systems it is possible to delineate malignant tumors [8–10], detect invasions [11–14] and even classify lesion types [15]. These studies pointed out that complex surface topography and tomography is one major challenge when shoulders, nose, chin or other facial skin areas are imaged with hyperspectral cameras. Hyperspectral imaging of such complex areas requires a small-sized hand-held spectral imager. Additionally, the illumination needs to be handled similarly, as the illumination of the hyperspectral images needs to be stable and constant in order to the captured data to be easily analysed and processed.

This article is the first stage of a pilot study completed in three stages from 2020 to 2022. The aim is to introduce the working principles and clinically pre-test the concept of the new, compact hand-held SICSURFIS Piezo-actuated metallic mirror FPI hyperspectral imager (SICSURFIS HSI) for complex skin surfaces. This pre-test enables the system engineering, imaging, and analysis aspects of further system development toward a clinical application.

The SICSURFIS HSI is designed to face the mentioned challenges of the previous studies. For illumination and challenging skin surfaces, there is a controllable built-in light-emitting diode (LED) based illumination module, which is designed for photometric stereo imaging. The special stray light protection cones will block the unwanted light and enable the imager to reach and adapt to complex skin surfaces. The imager produces spectral, spatial and surface topography information.

In the data processing, the raw measurement data were processed into the surface albedo, normals and surface model. The processing was performed with common hyperspectral reflectance calculations and a photometric stereo method for the surface model calculation [16]. Training and testing data sets were composed of windowed pixels (HS-sub-cubes) from the lesions that were histologically confirmed by an experienced dermatopathologist. This pixel-wise classification approach provides large training data (31,168 HS-sub-cubes after augmentation) for the machine learning model.

Our hypothesis is that these surface models can improve the hyperspectral imaging machine learning model's performance in clinical decision making and the new imaging concept can adapt to the complex skin surfaces. The skin surface models combined with the spectral and spatial domains will benefit the classification and delineation results of the convolutional neural network (CNN). In this first-stage pilot study, the aim is to compare the capacity of the HS in differentiating malignant basal cell carcinomas (BCCs) from clinically similar-appearing but benign intradermal nevi (ID) by comparing the surface models and albedo maps of the measured BCCs and IDs with CNN. So far, this is the pre-study for testing the concept, and evaluating the technical and instrumental aspects from the analysis point of views. The prototype imager with its application is not ready for a clinical use, but the first results and future potential can be evaluated. The mentioned second and third study stages will be independent continuation clinical pre-studies, with more lesions and lesion types. The imaging procedure, methods, and machine learning models will be improved based on these first steps.

The paper is organised as follows. Section 2 describes the overall and mechanical design and system validation of a novel skin cancer HS imager. Section 3 describes the steps of a clinical pre-test from instrumentation (Section 3.2), materials and methods (Section 3.3) to the results (Section 3.4) of the photometric stereo and the spectral 3D classification. The discussion (Section 4) and conclusions (Section 5) finalise the study.

2. The Sicsurvis Hyperspectral Imager Concept

This section introduces the SICSURFIS HSI's concept in six subsections; the design and operating principles, spectral responses and imaging calibration are presented in Sections 2.1 and 2.2. The following topics are LED illumination module (Section 2.3),

optical design (Section 2.4) and system validation (Section 2.5). The photometric stereo imaging methods (Section 2.6) will finalise the concept presentation.

2.1. The Design and Operating Principles of the Spectral Imager

The SICSRIFIS Spectral Imager (SICSRIFIS HSI) was designed to provide spectral images enlightened from different angles of light for the photometric stereo algorithm. The stray light protection cones (Figure 1) block the unwanted background light and ensure the correct distance for focusing the images. The soft silicone collars adapt the stray light protection cones softly to complex skin surfaces and provide comfort for the patient, since the imager must be held relatively tightly against the skin. The silicone collar is resistant to strong surgical disinfectants. The diameters of the light protection cones are 2.0, 3.6, 4.9 and 8.5 cm.



Figure 1. SICSRIFIS spectral imager with integrated LED module and stray light protection cones.

SICSRIFIS HSI's modules, sensor, spectral separator and LEDs are independently controllable, which enables effective configuration via software. Imagers spectral range is 475–975 nm and the spectral resolution (full-width at half maximum, FWHM) 8–18 nm. The average FWHM in selected spectral channels in this study was ~10 nm. In this study, we utilised 33 selected spectral channels, but the imager is, depending on the calibration, capable of capturing hundreds or even thousands of spectral channels. The imager's pixel resolution is approximately $24 \mu\text{m} \times 24 \mu\text{m}$. In spectral imaging, the tissue penetration depth depends on the wavelength. In this study, the used wavelengths had penetration depth from 0 to 6 mm [17].

The SICSRIFIS HSI with its LED module and stray light protection cones is a small-sized hand-held device, the overall weight of the imager is 880 g. The imager is easy to apply to patients' skin and it reaches the challenging places. The patient does not have to move, so imaging is also possible in patients with reduced mobility. The imager is a spectral scanner, but it is almost as fast as a snapshot. Nor does it describe hundreds of wavelength ranges and thus add extra dimensions to the data. The number of wavelengths and the wavelengths of interest can be selected on a case-by-case basis, just as the LEDs can be individually controlled to match the current wavelength. This study used the wavelengths

that had spectral absorption peaks in tissue chromophores such as melanin, haemoglobin, water, beta-carotene, collagen, and bilirubin [18].

The size and adaptivity are advances compared to devices used in previous studies. Example images of previous and current HSI systems can be seen in these articles [13,14,19–22], (Figure 1 in all of them). The SICSURFIS HSI’s imaging setup can be seen in Section 3.2, Figure 11a.

The imager consists of a Piezo-actuated metallic mirror Fabry–Pérot Interferometer (FPI), an RGB sensor and an LED light source. The FPI controls the light’s transmission to the RGB sensor, and the role of the separate long and short pass filters, shown in Figure 2, is to cut the unwanted transmission at not selected orders of the FPI. The sensor’s basic principle is to provide different spectral layers by changing the FPI air gap [4]. Typically the FPI air gap can be changed to a new value in less than 15 ms, i.e., the settling time of the air gap is 15 ms.

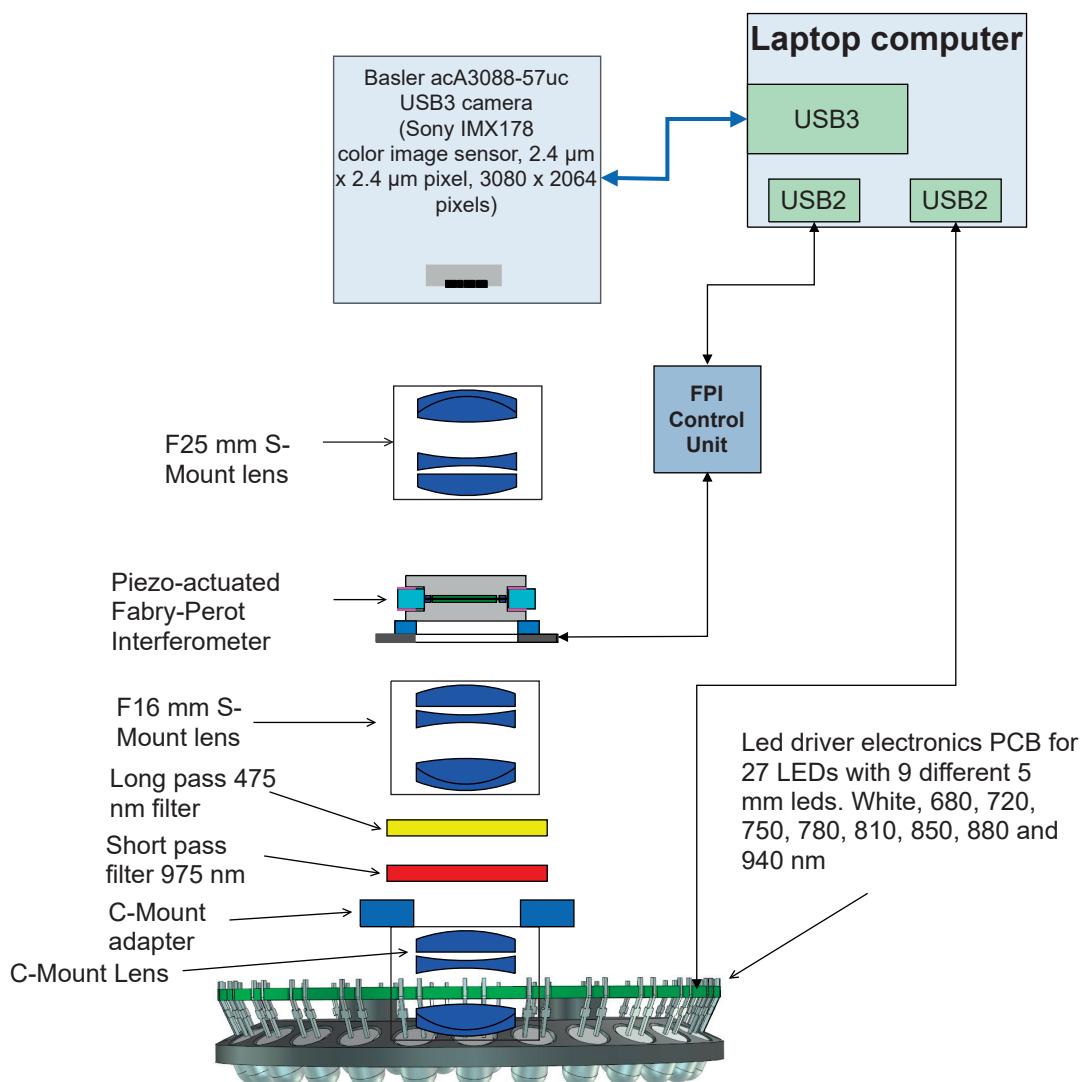


Figure 2. Block diagram of the SICSURFIS spectral imager.

Those FPI orders are matched with different sensitivities of the image sensor channels. There are three wavelength channels with different pixels in the RGB sensor’s Bayer pattern. If we carefully select the FPI air gap range, the RGB sensor will receive one to three transmission peaks. After recording the transmission peaks, the different spectral

responses of the red, green and blue pixels can be seen throughout the selected spectral range [4,23].

2.2. Pixel Spectral Response and Imager Calibration

Before spectral calibration, we demosaic each Bayer pattern frame using bilinear interpolation, so that response in each pixel is now vector $\mathbf{s} \in \mathbb{R}^3$. We can now arrange these pixels to a response matrix \mathbf{S} , which is $[3 \times n]$ matrix, where n is the number of pixels. We are interested in reconstructing radiance \mathbf{R} which is $[n \times 3]$ matrix. Described FPI system makes it possible to have one to three wavebands with one FPI gap. We need to determine the function between these two values to achieve this. The separate spectral calibration process in Section 2.5.1 will give use FPI gap wise coefficient matrix \mathbf{C} ($[3 \times 3]$ matrix) using radiance \mathbf{R}_λ information from the reference spectrometer so that

$$\mathbf{S} = \mathbf{CR}_\lambda. \quad (1)$$

From Equation (1) we solve coefficient matrix C . Now, the radiance is:

$$\mathbf{R} = \mathbf{C}^{-1}\mathbf{S}. \quad (2)$$

2.3. Led Illumination System

The LED illumination system of the SICSURFIS HSI is designed for the photometric stereo imaging setting. The LED light source module is a driver electronics PCB with 27 LEDs. The inner radius of the ring is 33 mm. The LED light source module has 27 pieces of intentionally selected 5 mm LEDs: white, 680, 720, 750, 780, 810, 850, 880 and 940 nm. Three series of these 9 LEDs are tilted by an angle of 30 degrees relative to the system optical axis. LEDs and FPI positions are individually controllable. The captured wavelengths, selected with FPI, can be combined with according LED illumination via software. Figure 3 visualizes the LED light source module's technical details. The temporal stability of the LED light source is taken into account by keeping the system on for several minutes before the recording of spectra. The other way to control the intensity stability is to record white reference images frequently.

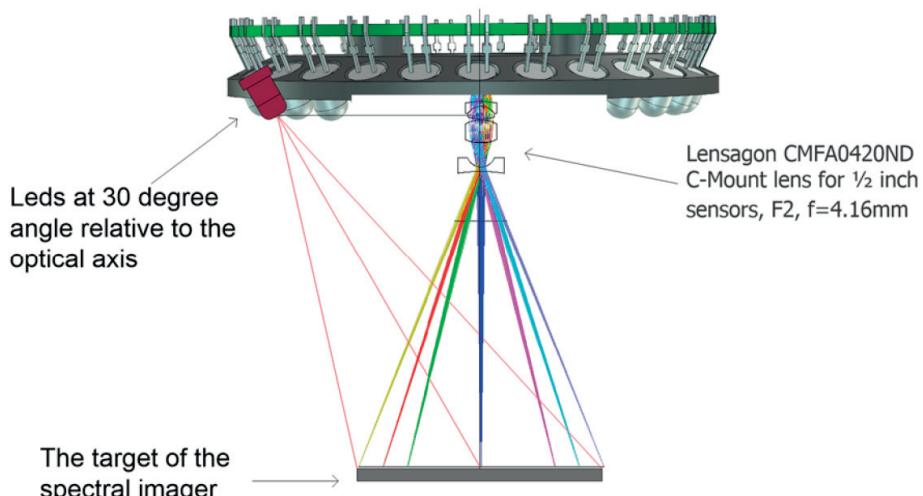


Figure 3. Three sets of nine LEDs are at a 30-degree angle relative to the system's optical axis. These three same wavelength LEDs sets are located at 120-degree intervals on the Led PCB.

2.4. Optical Design

The optics of the SICSRFIS HSI is designed with commercial S-mount and C-mount lenses, which provide collimated light beam through the Piezo-actuated FPI tunable filter. The selected optical design is described in Figure 4. The Lensagon CMFA0420ND C-Mount lens (F-number 2.0, focal length = 4.16 mm) forms an intermediate image at the focal plane of the Lensagon B5M16020V2 5 Megapixel S-Mount lens (F-number = 2.0, focal length = 16 mm). This lens collimates the light coming from the intermediate image. The collimated light goes through the Piezo-actuated FPI tunable filter to the Lensagon B5M25024V2 5 Megapixel S-Mount lens (F-number = 2.4, focal length = 25mm), which focuses the image of the target on the image sensor.

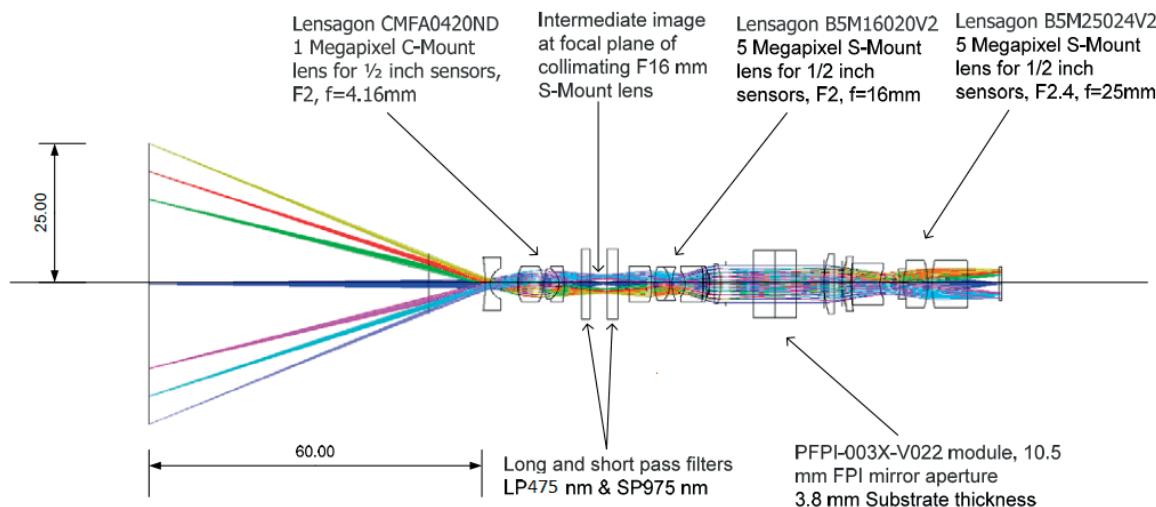


Figure 4. Optical concept of the imaging system of the SICSRFIS Spectral Imager.

2.5. System Validation

2.5.1. The Monochromator Calibration Setup

Before the monochromator calibration, the transmission spectra characterization of the selected FPI module was carried out. The spectra were measured with the Ocean Optics HR4000 spectrometer at the center of the FPI. The monochromator calibration of the SICSRFIS HSI camera was performed using the setup shown in Figure 5.

The combined sensitivities of the FPI and red, green and blue pixels were determined for over 200 FPI air gaps. In the calibration data analysis, the coefficients for the combined sensitivities of the FPI and red, green and blue pixels are retrieved in such a way that the linear combination of the R-, G- and B-pixel sensitivity signals contains none zero signal only at one spectral band.

In the monochromator calibration, the signals of the calibrated optical power meter (Thorlabs PM16-120 USB Power meter) and the SICSRFIS HSI were recorded for the wavelength range 475–975 nm at 2 nm intervals and the Full-Width Half Maximum (FWHM) resolution of 5 nm.

The results of the system calibration for SICSRFIS HSI are the raw measurement data of the monochromator calibration, the spectral response functions of R-, G- and B pixels for Piezo-actuated FPI (PFPI) setpoint voltages used in the calibration, and the system calibration matrices that can be used to calculate the spectral radiance or photons at the input aperture of the hyperspectral camera optics from the raw pixel signals. The exposure time and the camera gain for pixels must be known relative to the exposure time and pixel gains used in the monochromator calibration in order for absolute calibration to be valid. Examples of the measured combined sensitivities of the FPI and red, green and blue pixels and linear combinations of these sensitivities are presented in the next section.

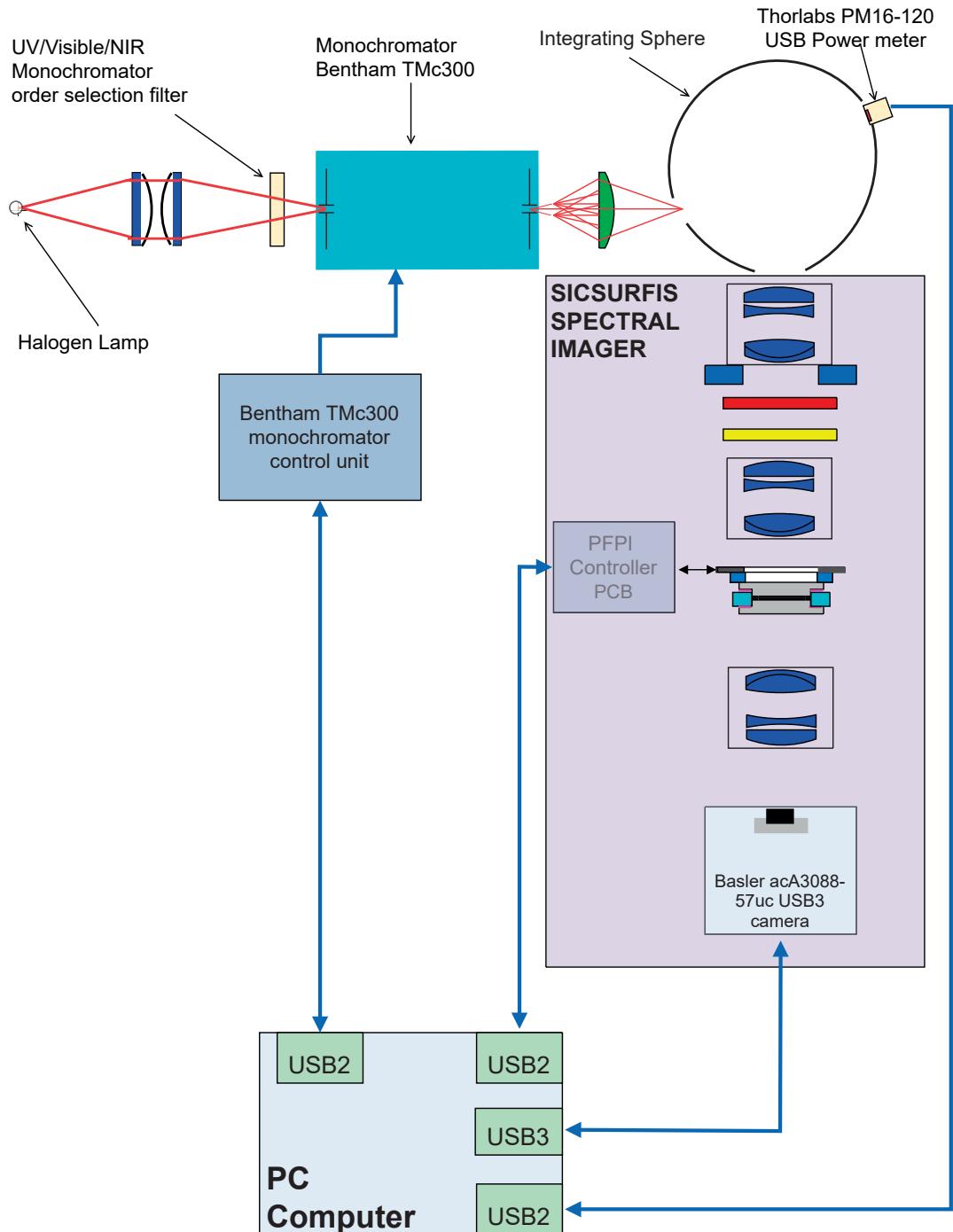


Figure 5. The SICSURFIS hyperspectral imager calibration measurement setup.

2.5.2. Led Illumination System

The LED light source module's spectral radiances were measured using calibrated fiber, HR4000 spectrometer and white balance reflectance target, using the setup shown in Figure 6. The LEDs were selected to cover the wavelength range of the used FPI. The results of the single LED spectral radiance measurements can be seen in Figure 7.

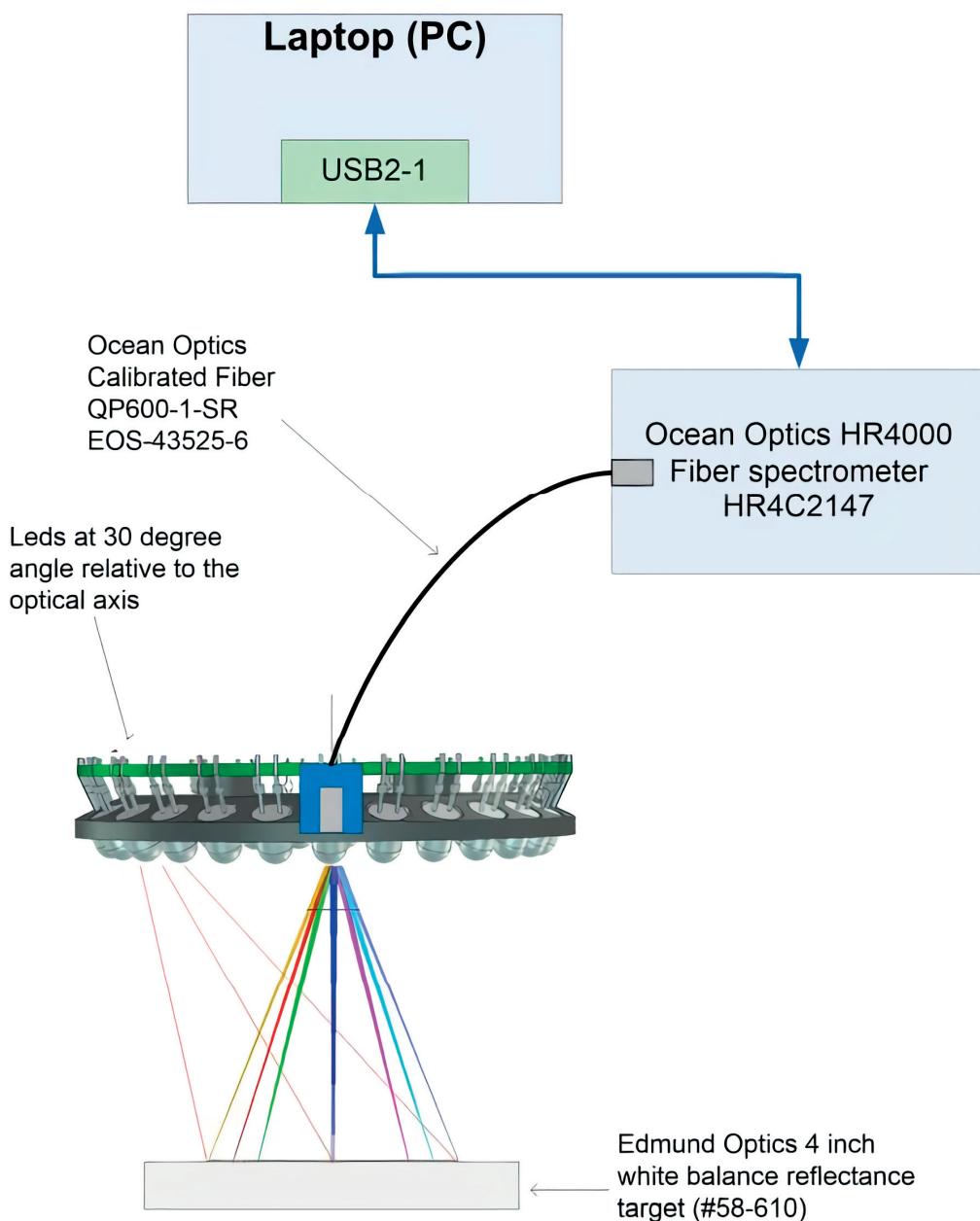


Figure 6. The setup was used in the spectral radiance measurements of LEDs and LED groups using the Ocean Optics Calibrated Fiber QP600-1-SR EOS-43525-6 and HR4000 spectrometer. Edmund Optics 4 inch white balance reflectance target (#58-610) was used in the measurements.

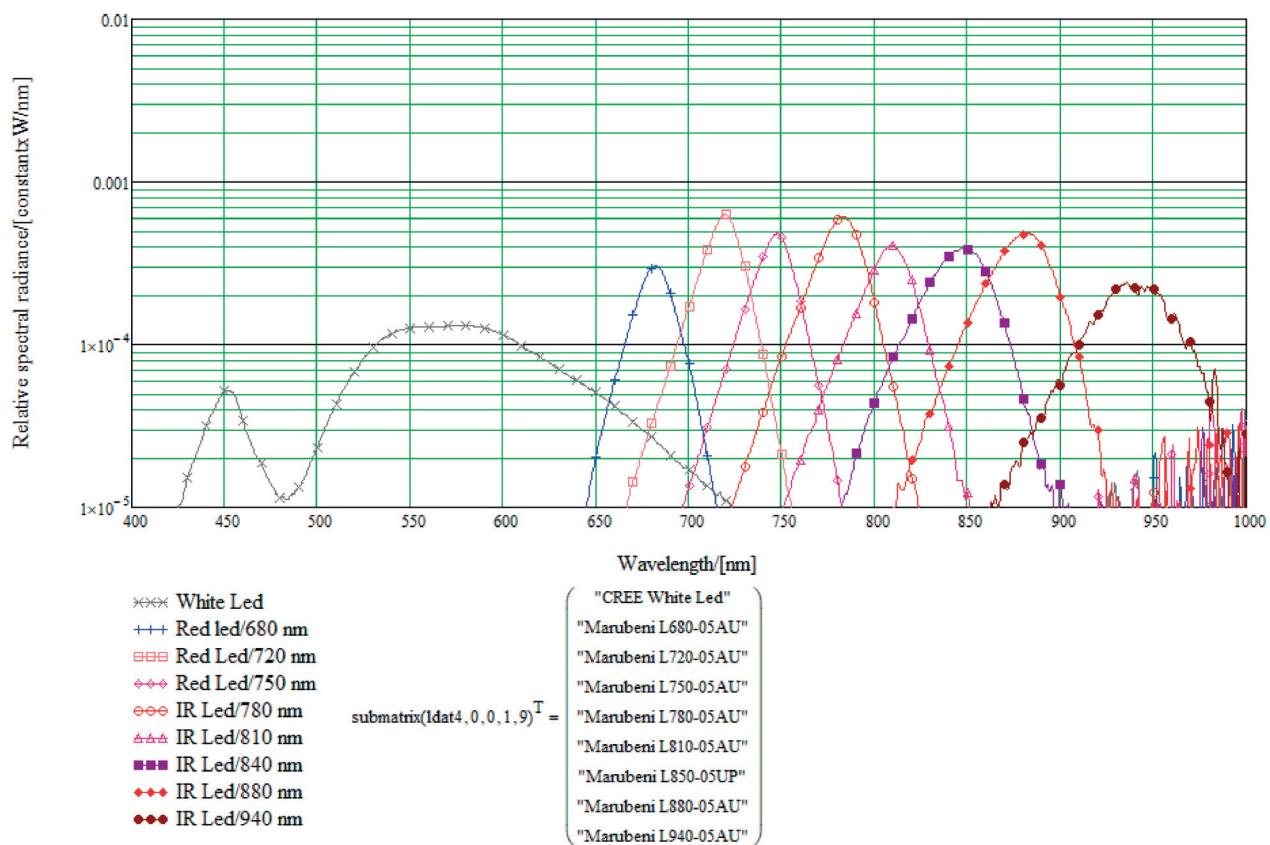


Figure 7. Results of the single LED spectral radiance measurements with the setup shown in Figure 6. The ma38 mA DC LED current was used for each LED. The distance of the target in this measurement was 50 mm.

2.5.3. Sicsurpis Hsi Camera Monochromator Calibration Pixel Sensitivity Function Results

The purpose of the monochromator calibration is to determine the combined sensitivities of the FPI and red, green and blue pixels (see Figure 8). The peak wavelengths 1, 2 and 3 are determined by the spectral transmission spectrum of the FPI (lower left part of Figure 8). When the FPI transmission curve and R-pixel quantum efficiency are multiplied, we get the Red curve in the center right part of Figure 8. Similarly, we get the Green curve for G-pixels and the Blue curve for the B-pixels in the center right part of Figure 8. An example of the results of the SICSURFIS HSI camera monochromator calibration are shown in Figures 8–10.

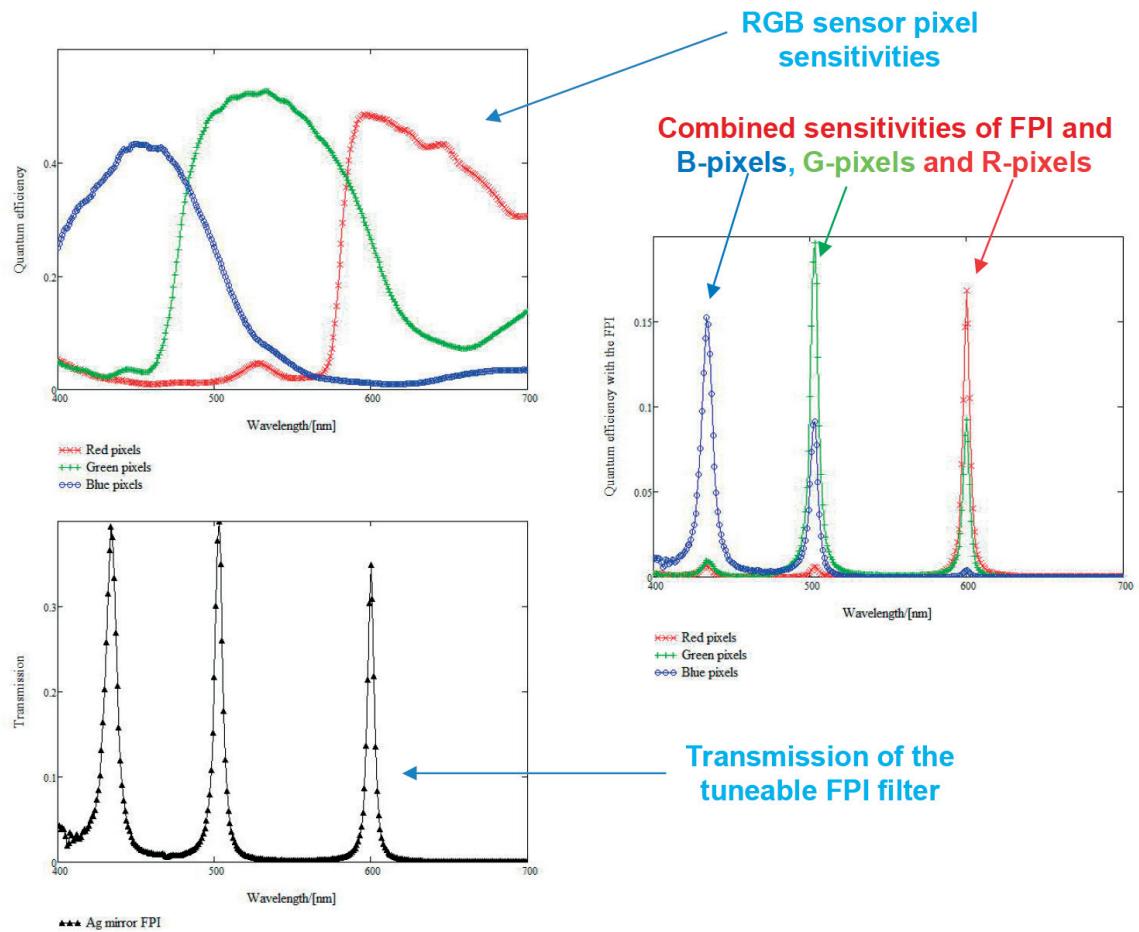


Figure 8. Quantum efficiencies of RGB CMOS image sensor (upper left) red, green and blue pixels, Transmission of metallic mirror FPI (lower left) and combined sensitivities of the FPI and red, green and blue pixels (upper right).

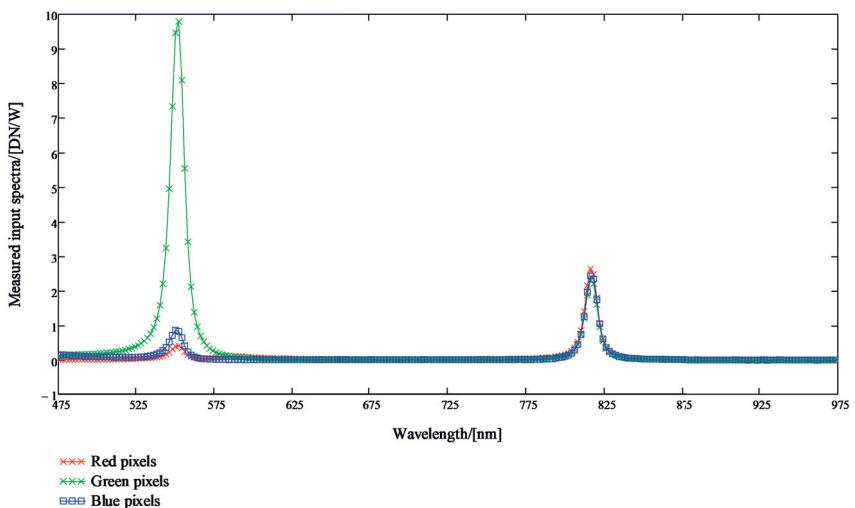


Figure 9. Results of the SICSURFIS HSI monochromator calibration. The scaled R-, G- and B-pixel sensitivity signals are plotted in units DN/(W/nm). There are two peak wavelengths for the selected PFPI drive voltage, 548.2 nm (on the left) and 812.2 nm (on the right).

In Figure 9, the scaled R-, G- and B-pixel sensitivity signals are plotted in units DN/(W/nm). The linear combination of the scaled pixel sensitivity functions for the peak wavelengths 1, 2 and 3 can be seen in Figure 10. There are two peak wavelengths for the selected PFPI drive voltage; 548.2 nm (on the left) and 812.2 nm (at the center).

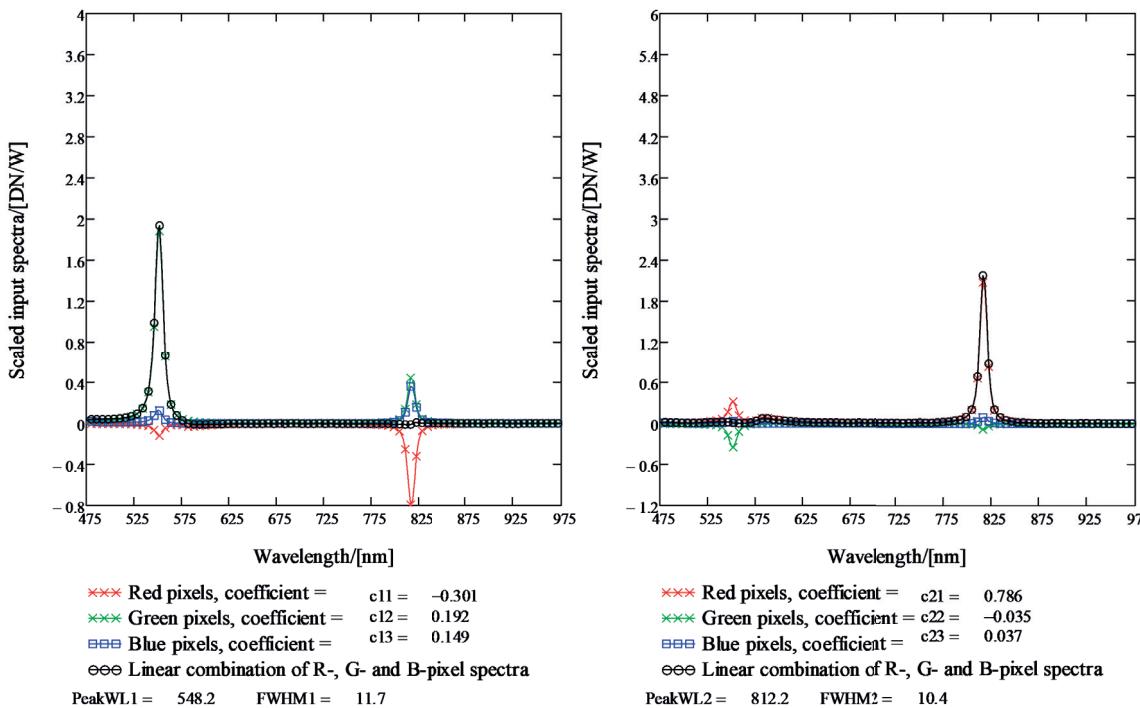


Figure 10. Results of the SICSURFIS HSI monochromator calibration. The linear combination of the scaled pixel sensitivity functions for the peak wavelengths 1, 2 and 3. For the selected PFPI drive voltage, there are two peak wavelengths, 548.2 nm (on the left) and 812.2 nm (at the center).

2.6. Photometric Stereo Imaging

Photometric stereo is an imaging setting where the imaged object is illuminated from multiple angles, and corresponding images are captured with a stationary camera. When the angles of the lights are known compared to the imaged object, one can calculate the surface normals of the imaged object based on the intensity of gathered light in different imaging angles [24]. From the surface normals, one can calculate a three-dimensional surface model with the Frankot–Chellappa algorithm [16].

The normal maps for each wavelength are calculated by multiplying the inverse matrix of the light direction matrix by the three reflectance values for each pixel, corresponding to each light direction. The light directions are defined by the length from the imaged object to the center of the LED-module, 65 ± 5 mm, the diameter of the LED-module, 33 mm, and the placement of the LEDs on the module, as described in Figure 3. Same LEDs are separated by 120 degrees, i.e., $\frac{2\pi}{3}$, on the edge of the module, and the angle between the object layer and the arriving light vector is 60 degrees, $\frac{\pi}{3}$. This gives us the normalized light direction matrix for the center of the imaging field:

$$L = \frac{1}{2} \begin{bmatrix} 0 & 1 & \sqrt{3} \\ \frac{\sqrt{3}}{2} & -\frac{1}{2} & \sqrt{3} \\ -\frac{\sqrt{3}}{2} & -\frac{1}{2} & \sqrt{3} \end{bmatrix}. \quad (3)$$

Now the normal N and albedo a matrices for each wavelength band are calculated as

$$aN = L^{-1} \cdot \begin{bmatrix} r_{000} & r_{001} & r_{002} \\ r_{010} & r_{011} & r_{012} \\ \vdots & \vdots & \vdots \\ r_{km0} & r_{km1} & r_{km2} \end{bmatrix}, \quad (4)$$

where each subscript for reflectance r mean its x -coordinate, y -coordinate, and the used light direction, respectively. The resulting matrix, which is of shape $k \cdot m$ rows and three columns, is then reorganized into the normal map by transforming it into the original image shape:

$$aN_{\text{map}} = \begin{bmatrix} a_{00}(n_{000}, n_{001}, n_{002}) & a_{01}(n_{010}, n_{011}, n_{012}) & \dots & a_{0m}(n_{0m0}, n_{0m1}, n_{0m2}) \\ \vdots & \vdots & \ddots & \vdots \\ a_{k0}(n_{k00}, n_{k01}, n_{k02}) & a_{k1}(n_{k10}, n_{k11}, n_{k12}) & \dots & a_{km}(n_{km0}, n_{km1}, n_{km2}) \end{bmatrix}, \quad (5)$$

where albedo a_{ij} are the lengths of the vectors:

$$a_{ij} = |(aN_{ij0}, aN_{ij1}, aN_{ij2})|. \quad (6)$$

From this, we can calculate the partial derivatives of the depth in x - and y -direction in every point (a,b) and denote them by p and q :

$$p_{ab} = \frac{\partial z}{\partial x} = \frac{n_{ab0}}{n_{ab2}}, \quad (7)$$

$$q_{ab} = \frac{\partial z}{\partial y} = \frac{n_{ab1}}{n_{ab2}}. \quad (8)$$

Now we calculate the Fourier transform \mathcal{F} for each p and q map:

$$P = \mathcal{F}(p), \quad (9)$$

$$Q = \mathcal{F}(q). \quad (10)$$

Now the estimated surface map is:

$$Z = \mathcal{R} \left(\mathcal{F}^{-1} \left(\frac{-i\omega_x \cdot P - i\omega_y \cdot Q}{\omega_x^2 + \omega_y^2 + \epsilon} \right) \right), \quad (11)$$

where \mathcal{R} represents the real part of the value, \mathcal{F}^{-1} the inverse Fourier transform and ω the frequencies in the Fourier transform. ϵ is added to avoid division by zero. The details of the method can be found in [16].

3. Clinical Pre-Test: Surface Model Classification of Basal Cell Carcinomas and Intradermal Nevi

The clinical pre-test section is divided into four sections. The background (Section 3.1) and instrumentation (Section 3.2) leads to the material and methods (Section 3.3), which carefully explain the steps from spectral data and pre-processing (Section 3.3.1) to machine learning pre-processing (Section 3.3.2), method validation (Section 3.3.3) and lesion classification (Section 3.3.4). After the methods, we will present the results in Section 3.4; the photometric stereo and albedo spectra (Section 3.4.1) and pixel-wise classification (Section 3.4.2) with the slice half model and the leave-one-out validation (Section 3.4.3).

3.1. Background

The clinical data gathering was performed in two phases during the spring and autumn of 2020. For the demonstrative pilot-test in this study, we examined two types

of skin lesions, malignant basal cell carcinomas (BCC) and benign intradermal nevi (ID), with a clinically similar appearance.

The HS images were captured in the first phase, and the lesions were clinically diagnosed and annotated by dermatologists at Helsinki University Hospital. In the second phase, we calculated the surface models of the skin surfaces using the methods of photometric stereo imaging. The aim was to classify the pixel spectra of BCCs, IDs, and healthy skin. We used a convolutional neural network and pixels selected from the albedo images and the lesions' depth data.

We captured images of 14 BCCs and 8 IDs on 21 volunteering patients with HS, digital and dermoscopic imaging and subsequently removed the lesions for dermatohistopathological analyses to confirm their diagnoses. All volunteering patients provided their written informed consent. The study protocol followed the Declaration of Helsinki and was approved by the Ethics Committee of Helsinki University Hospital.

One of the typical challenges related to CNN classification and HS imaging is the limited availability of the labelled training data, which can lead the models to overfit [3]. Thus we selected a pixel-wise classification approach. The training data set consisted of 6160 windowed HS-sub-cubes collected from the HS images of histologically confirmed skin lesions. After data augmentation training set contained 31,168 HS-sub-cubes.

3.2. Clinical Pre-Test Instrumentation

The HS images were captured by dermatologists and nurses (users). The SICSURFIS HSI was combined with a specially designed hospital version of the CubeView software [25]. CubeView is a spectral imaging and analysis software developed by the spectral imaging laboratory of the University of Jyväskylä, Finland [26]. It controls the SICSURFIS HSI's machine vision sensor, PFPI and LED modules by using the Camazing [27] and the Spectracular [28] Python libraries.

The data capturing setup was designed for effortless workflow. All of the device and setup settings (e.g. exposure time, LEDs, wavebands) were pre-assigned via software, and the imager was ready to work without any adjustments so that the user could concentrate on the patient.

After the patient number and lesion number was given to the system, the user interface guided the user to capture the dark and white references. The dark acquisition was performed by manually placing the imager to a light-blocking holder, seen in Figure 11. The imager was set to capture 40 frames, and the mean was used as a dark reference. The white reference procedure was similar—the imager was placed on a holder against white Teflon. The imager was set to capture with matching LED and wavelength settings, as it captures the HS images.

The user interface provided a preview video to target the image while the HS imager was placed on the patient's skin. Three different LED light and waveband combinations were captured with a single process and one click to the capture button. With one click, the system captured six HS images. After capturing the HS image, the software calculated reflectance frames automatically for each wavelength and visualized those frame by frame in an animation view. The quality of the HS images was ensured visually from the animation by the users. At the end of the effortless workflow, the user could save or dismiss captured HS images with one click.

The user interface, the HS imager, and the chosen test setup (LEDs and correct wavebands) were pre-tested at the hospital with the users before the imaging for the clinical pre-test was started. The users were educated on the software and HS imager. The imager, dark and white reference targets and a computer with capturing software were mounted into a trolley so it could be stored in a secure place while unused (Figure 11a).

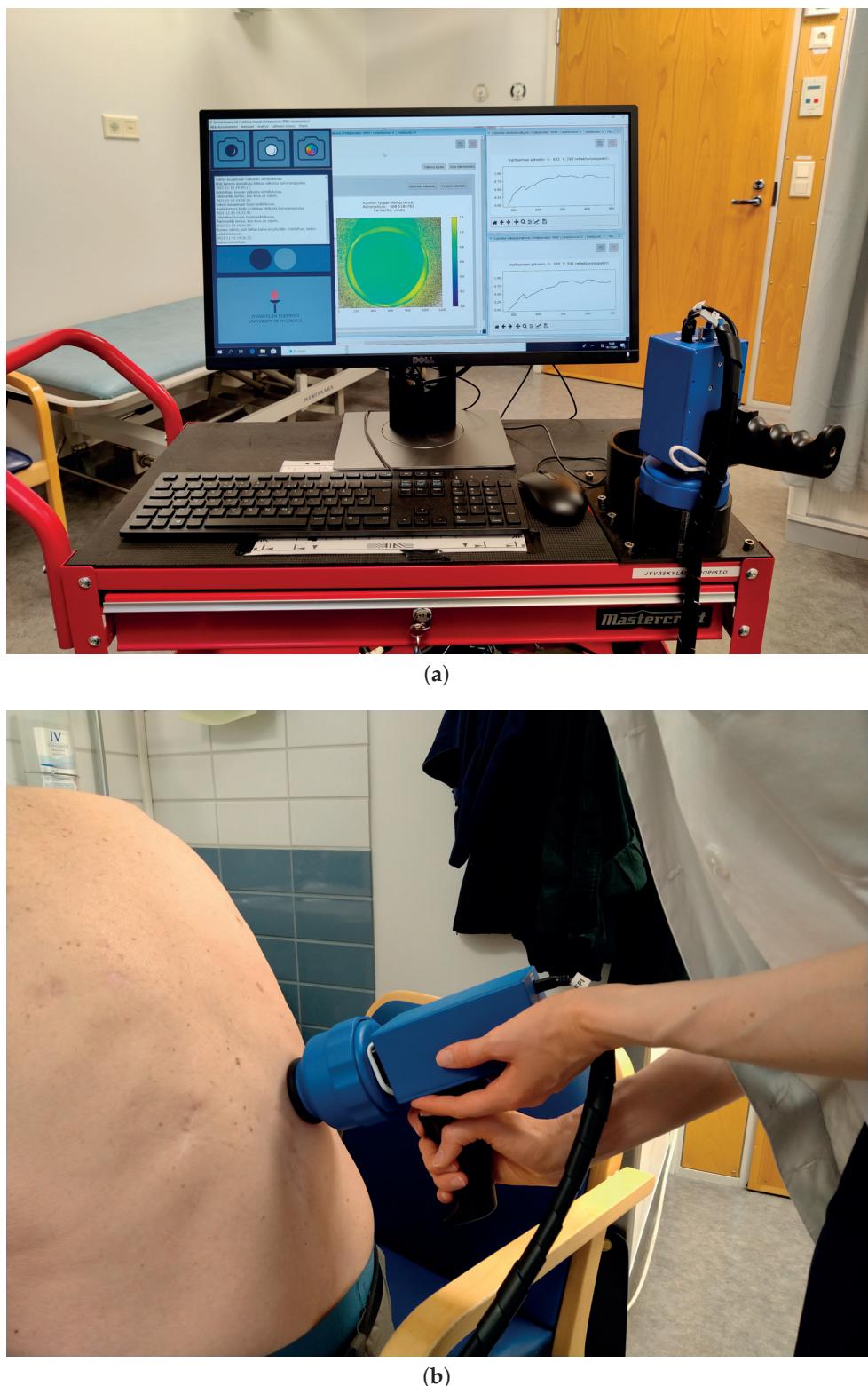


Figure 11. SICSURFIS imaging system. A handheld image sensor and a small computer for data collection are attached to a trolley that can be moved around hospital wards. Subfigure (a): SICSURFIS imaging system. Subfigure (b): SICSURFIS HSI in clinical use.

3.3. Materials and Methods

3.3.1. Spectral Data and Pre-Processing

One HS image consists of three sets of frames, captured with thirty-three selected wavebands. The visible light (VIS) had two LEDs on, and the visible and near-infrared (VNIR) bands had seven LED lights on during the capturing process. The selected wavelengths and corresponding led lights are shown in Table 1. Each set of the thirty-three wavebands was captured three times, each time with one of the three different angles of light.

Table 1. Selected wavelengths and corresponding LED light wavelengths.

VIS	477, 500, 524, 540, 550, 575, 578, 582, 600, 626, 630, 639, 651, 669, 677, 681, 686
VIS LED	white, 680
VNIR	700, 725, 735, 750, 760, 765, 775, 783, 790, 801, 812, 825, 834, 851, 878, 891
VNIR LEDS	720, 750, 780, 810, 850, 880, 940

The HS images of the lesions had identification numbers. The dermatologist made the clinical diagnosis for each lesion before imaging, and the final diagnosis was confirmed with histopathological analysis of each lesion. The dermatologists hand-drew the ground truth images based on the histologically confirmed diagnoses.

The images were saved as raw images and pre-processed twice. We call these stages raw image pre-processing and machine learning pre-processing. The raw image processing pipeline is described in Table 2. First, the radiance images are processed into combined reflectance images. After that, the photometric stereo, described in detail in the next section, is calculated based on the hyperspectral reflectance images. The resulting dataset contains an albedo image and depth map for each measurement.

Table 2. The image processing pipeline.

Phase	Description	Amount of HS Images
Radiance	The raw data are processed to radiance (R) and its white reference (W).	12 Six hyperspectral images, corresponding to combinations of three different light directions and two light wavelength ranges (visible light (VIS) and very near-infrared (VNIR)), and their white references.
Reflectance	From radiance image and its white reference, the reflectance (R_{refl}) is calculated as $R_{refl} = \frac{R}{W}$.	6 Six hyperspectral images, corresponding to combinations of three different light directions and two light wavelength ranges (VIS and VNIR).
Combine	The VIS and VNIR images are combined.	3 Three hyperspectral images corresponding to three different light directions.
Albedo and normal	The albedo (a) and normal N are calculated by Equations (3)–(6).	2 One hyperspectral image of the albedo and normal map for the same area.
Depth and albedo	Depth is calculated from N by Equations (7)–(11).	2 One hyperspectral image of the albedo and the depth map for the same area.
Smoothening	Bregman total variation denoising [29] is applied to the albedo data in the spectral direction. The algorithm is used as implemented in scikit-image version 0.17.2 [30].	2 One hyperspectral image of the albedo and the depth map for the same area.

3.3.2. Machine Learning Pre-Processing

Image quality reasons limited the number of the HS images from 14 BCCs to 10 and 8 IDs to 7. After the raw image pre-processing, each of the 17 selected HS image consisted of 33 albedo and 33 depth frames from the originally selected 33 wavebands.

In the first phase, we selected the most significant depth frame, which was the frame representing the wavelength of 575 nm in this test setup. The wavelength channel was selected based on the measured data's robustness while ensuring that the channel is well within the range of visible light. Using only one depth frame, we could reduce the unnecessary depth dimensions from the data. After reducing the dimensions, the data were normalised between 0 and 1, and the possible infinity and non-numeral (NaN) values on the areas outside the imagers field of view were set to 0.

Images were vertically sliced from the middle of each lesion. The training pixels from the histologically confirmed lesions (250) and the healthy skin (100) were randomly selected from the left side of the HS image. The testing data were selected similarly from the right side of the image. The healthy skin pixels were selected by using a healthy skin mask, which was hand-drawn based on the ground truth images and 'RGB' visualisations. The pixel selection is visualised in Figure 12. The ground truth labels of the selected pixels were selected accordingly.

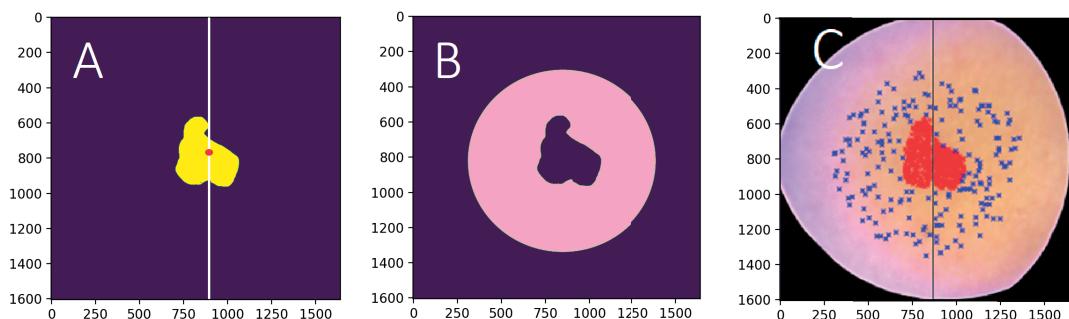


Figure 12. (A) The ground truth image, left side: training data, right side: test data. (B) The healthy skin mask is visualised with pink colour. (C) The visualisation of the selected training and testing data points. The selected lesion pixels are visualised in red, and blue represents the selected healthy skin pixels. The size of the HS image (A,B) was 1605 × 1640 px; the size of the divided training and testing portions (C) varies, depending on the lesion's location.

The training and testing subsets from the HS image were rolling window views of the selected pixels. The size of one window was 30 times 30 pixels with 34 channels, and the step was set to 5. The training subset with its 6160 windowed HS-sub-cubes were balanced with Imbalanced learn library's random over-sampling method [31]. After balancing, the data were augmented with vertical and horizontal flipping. The final size of the training data was 31,168. The validation data size was 1190. The test data had 5950 similarly windowed pixel HS-sub-cubes and their ground truth. The pixel-wise classification was performed using these randomly selected samples.

The classification maps shown in Section 3.4, were produced only for visualising the pixel-wise model's potential and challenges of classifying and delineating whole HS images. For those classification maps, the whole HS images were pre-processed and windowed as described above. The accuracy metrics presented in this study (Table 3) are based on tests conducted with the pixel-wise analysis, using the above mentioned test data pixel-wise HS-sub-cubes.

3.3.3. Method Validation

We validated the results by a leave-one-out approach to see if the pixel-wise slice half method causes biasing. The approach was conducted so that convolutional neural network (CNN) models were trained 17 times, omitting one lesion's HS-sub-cubes (250 windowed

lesion and 100 windowed healthy skin pixel HS-sub-cubes, size of $30 \times 30 \times 34$) from the training material. Each model was tested separately. The test data were randomly selected from the pixels on the right side of the omitted image (250 windowed lesion HS-sub-cubes and 100 healthy skin windowed HS-sub-cubes).

As a control for this setup, each of the 17 lesions were classified with the original slice half model, trained with the 31,168 left-side HS-sub-cubes. Each of the image-specific pixel-wise HS-sub-cube results were calculated similarly to the leave-one-out approach by selecting similar windowed 250 lesion HS-sub-cubes and 100 healthy skin cubes from the right side of the lesion HS images.

The study was implemented with Scikit-Learn [32], Scikit-Image [31], SciPy [33] and Tensorflow [34] Python libraries. The computing was performed with a Linux GPU server, 1 × Tesla P100, $\times 86_64$.

3.3.4. Cnn Pixel-Wise Classifier

As the used data contain spatial and spectral domains and one depth map from the maps constructed from visible light for the classification, the natural choice for the classification is CNN [35]. With the convolutional neural network's 3D and 2D layers, characterization of the data prior to the classifier is not needed, and the neural network is free to find connections invisible to human eye. In prior studies, the CNN has been deemed appropriate for tasks similar to the task in this study [3,11,15,36].

The limited availability of the labelled training data is one of the noted challenges related to the CNN classifier. Without a considerably large amount of training data, the models have a tendency to overfit [3]. Therefore, the selected classification approach was pixel-wise. The original 17 HS cubes were transformed into training data for the pixel-wise slice half model, consisting of 31,168 windowed $30 \times 30 \times 34$ HS-sub-cubes after data augmentation, validation 1190 HS-sub-cubes. The test data were, respectively, a set of 5950 HS-sub-cubes.

We used a convolutional neural network (CNN) for the classification. Figure 13 visualises the structure of the network. The 3D convolutional layers were used to extract features from the windowed $30 \times 30 \times 33$ HS albedo-sub-cubes. The construction of the 3D convolutional layers included the LeakyReLU activation function and max-pooling layers.

The 2D convolutional layers extracted the features from the windowed $30 \times 30 \times 1$ HS depth map sub-cube. The 2D layers were constructed with LeakyReLU and max-pooling layers. The results were flattened, concatenated and used as an input for the hidden layers. As a result, the model provided the pixel-wise output classification and prediction confidences for three classes: healthy skin, intradermal nevus, and basal cell carcinoma.

The model was trained using Adam optimizer with default parameters and the categorical cross-entropy loss function.

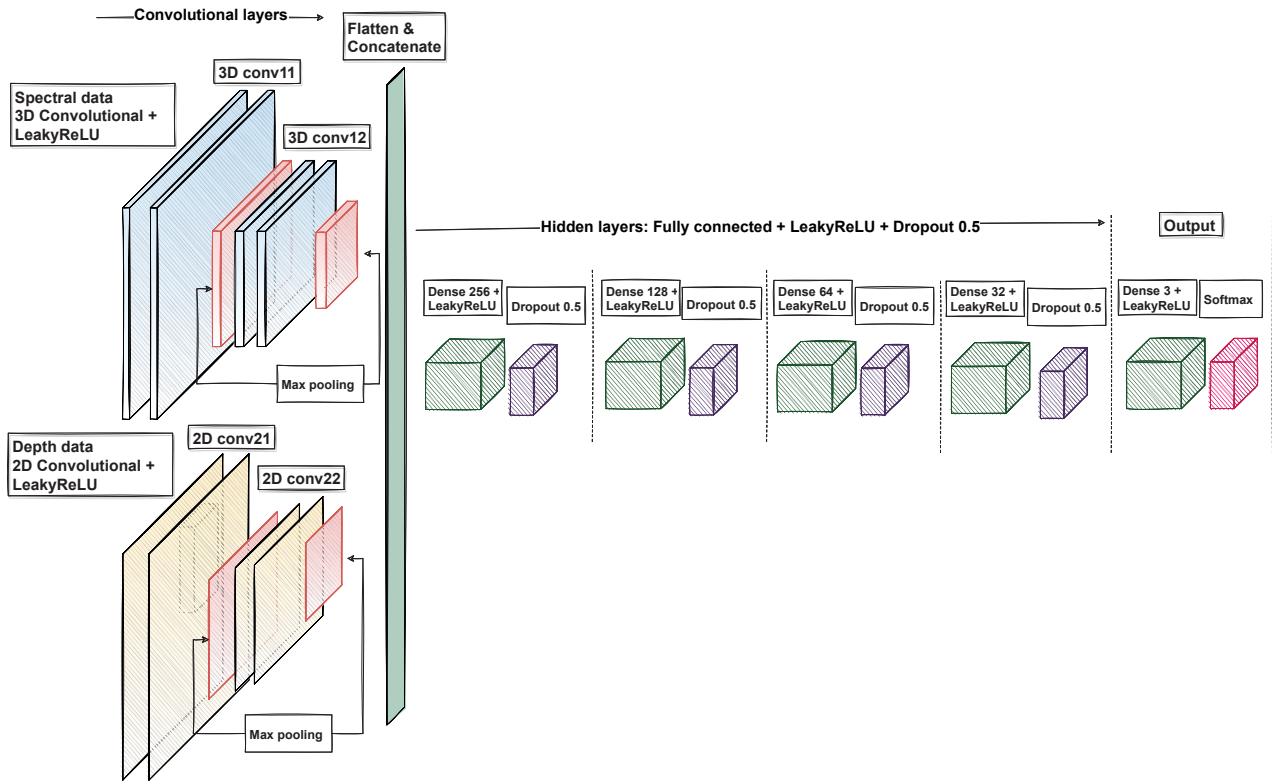


Figure 13. The visualisation of the used convolutional neural network. We used 3D layers with spectral data and 2D layers with depth data. The outputs were concatenated, flattened and used as an input for the hidden layers. The result was a three-class classifier for spectral data with depth information.

3.4. Results

3.4.1. Photometric Stereo and Albedo Spectra

To assess the quality of the photometric stereo transformation of the reflectance images, we take a look at three surface models: Surface model of a ball (Figure 14a), a Lego-brick (Figure 14b), and skin (Figure 14c,d). From them, we can see that the shape's of the objects are visible from the surface models. In these cases, the interesting part is in the middle of the imaged area, and therefore there is very little error in the images. However, as the light direction matrix is defined for the image area's centre, the error increases as the distance from the centre increases. Figure 14a–d are examples of cropped images, from the area that has a good quality. Outside the selected areas shown in the Figure 14a–d, the quality of the surface model was significantly weaker. These photometric stereo results were taken into account on selecting the training and test data by limiting the healthy skin area from the image border areas.

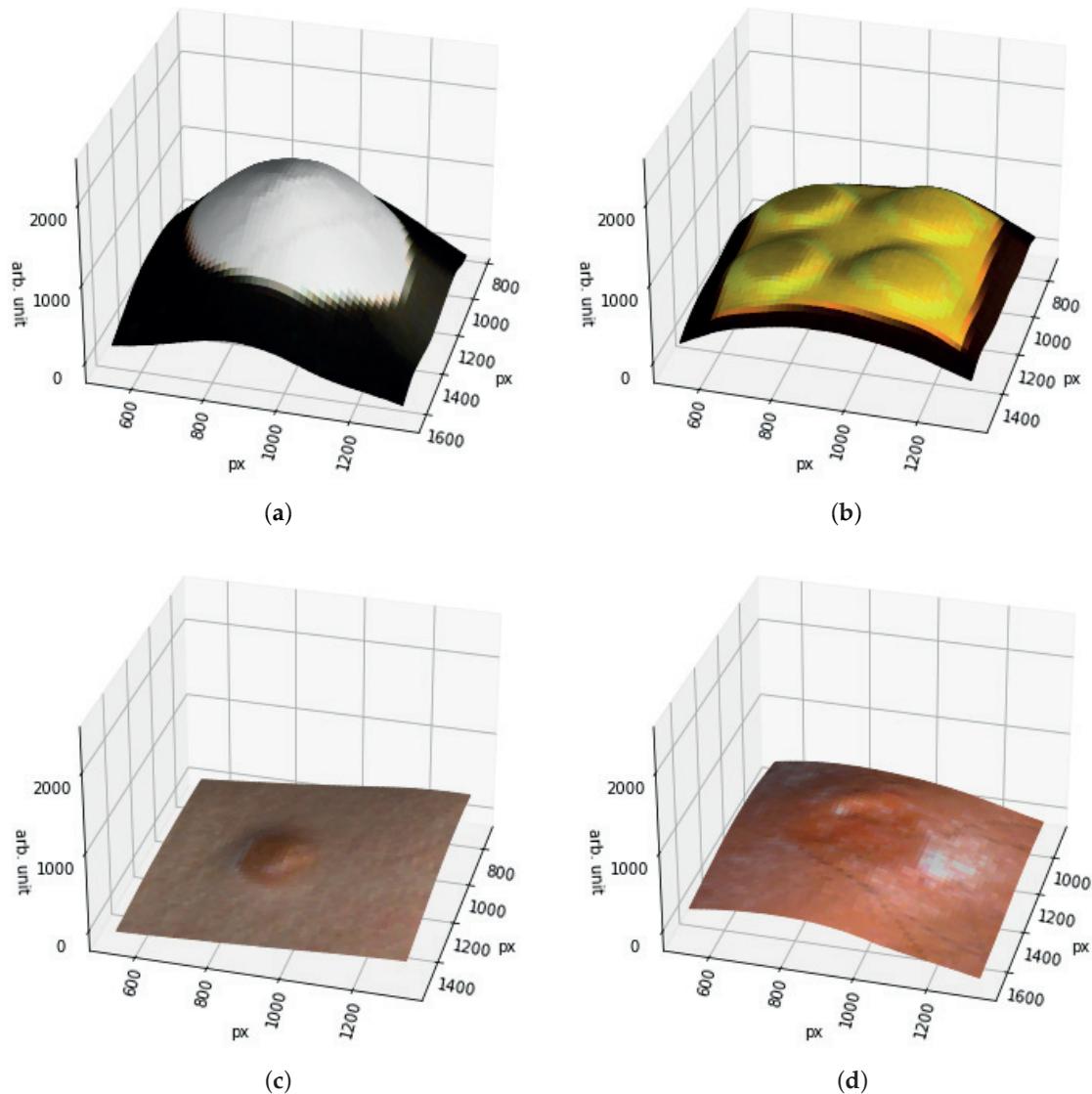


Figure 14. Surface models of (a) Small styrofoam ball, (b) Two times two yellow Lego-brick, (c) Intradermal nevus on human skin and (d) Human skin with nodular basal cell carcinoma.

An example of albedo spectra with their deviation can be seen in Figure 15.

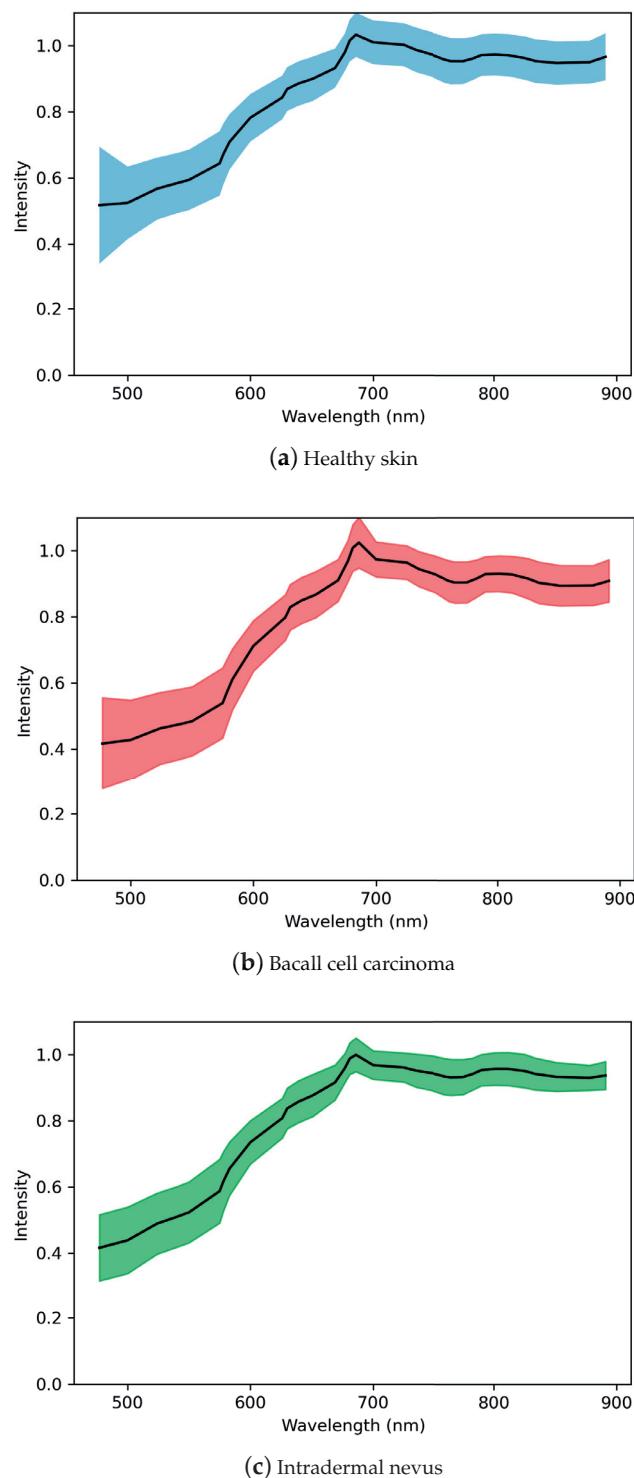


Figure 15. Sub figures (a–c) represent the albedo spectra of healthy skin (a), BCC (b) and ID (c). The minimum and maximum standard deviations are marked with coloured areas, and the mean albedo spectra are drawn with a black line.

3.4.2. Pixel-Wise Classification with Slice Half Model

The pixel-wise classification results of the slice half model in Table 3 show the precision, sensitivity and accuracy scores of the windowed HS-sub-cubes. The sensitivity is 0.81 with healthy skin and intradermal nevi and 0.76 with basal cell carcinomas. The average weighted precision was 0.81 and the accuracy over the whole testing dataset was 0.79. The results confirm that the model can distinguish the malignant and benign lesions at a pixel level. Since SICSRFIS HSI's pixel resolution is approximately $24\text{ }\mu\text{m} \times 24\text{ }\mu\text{m}$, one classified pixel is smaller than one cell.

Table 3. Classification report of the pixel classification, slice half model.

	Precision	Recall/Sensitivity	F1-Score	Support
Healthy skin	0.63	0.81	0.71	1776
Benign intradermal nevi	0.87	0.81	0.84	1697
Basal cell carcinomas	0.89	0.76	0.82	2477
Macro avg.	0.80	0.79	0.79	5950
Weighted avg.	0.81	0.79	0.79	5950

Confusion matrices (Figure 16) visualizes the pixel-wise classification results of the test data, shown in Table 3.

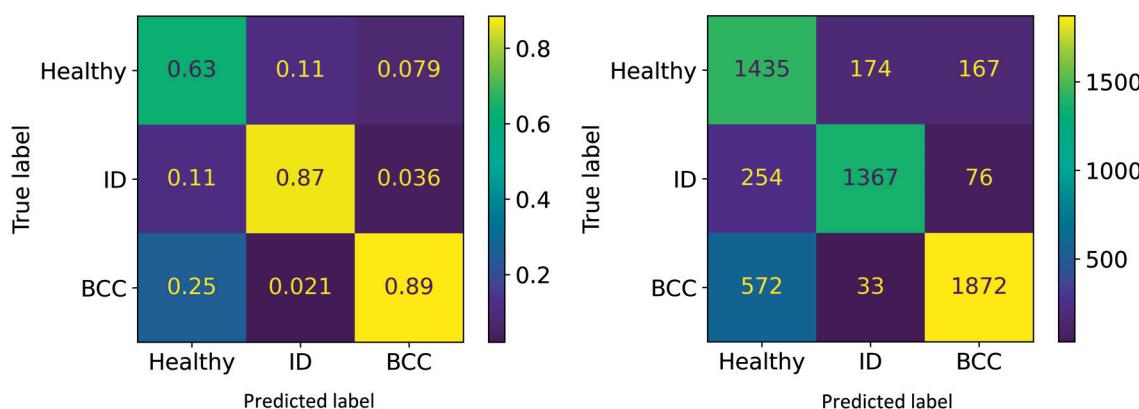


Figure 16. Confusion matrices. **Left:** Normalised predictions, **Right:** predictions.

Figures 17–19 are visualisations of the classified HS images. The images were conducted using the slice half approach model trained with 31,168 HS-sub-cubes. The classification method was pixel-wise, and these collages were conducted only for visual evaluations of the model's capabilities and challenges. The HS images were pre-processed to HS-sub-cubes similarly to the training and test data. The classification prediction confidences and classification maps can be seen on these collages.

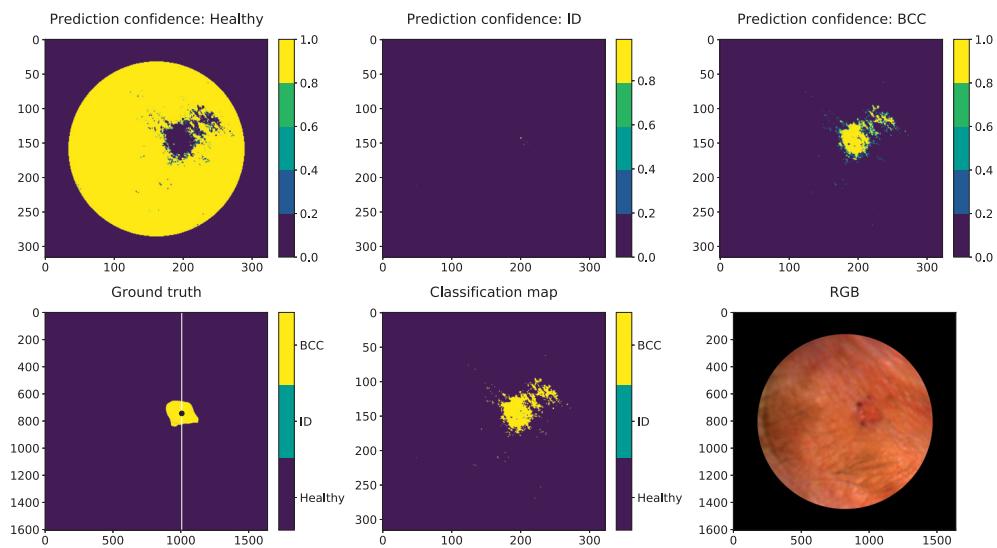


Figure 17. Nodular BCC visual pixel spectra classification example. First row: Healthy skin prediction confidence, prediction confidence of the ID and the prediction confidence of the BCC. Second row: The ground truth. The white line in the ground truth image shows the slicing position. The training data were collected as windowed ($30 \times 30 \times 34$) HS-sub-cubes. The middle pixels of those HS-sub-cubes (250 lesion and 100 healthy skin pixels) were randomly selected from the left sides of the lesions. We can see from the classification map (middle) that the nodular lesion is delineated relatively well, but some miss-classified pixels can be seen above the right upper corner of the lesion. The “RGB” illustration of the lesion is on the right. The lesion was captured from the left corner of the eye. The unit of measures is smaller with prediction maps since the HS images were windowed with 5 pixels step.

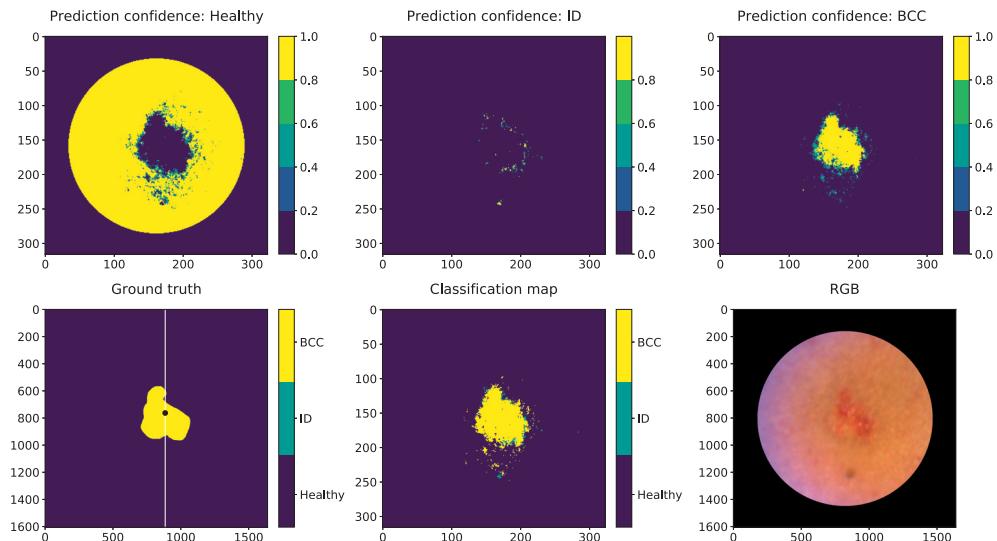


Figure 18. Nodular and superficial BCC visual pixel spectra classification example. First row: Healthy skin prediction confidence, prediction confidence of the ID and the prediction confidence of the BCC. Second row: The ground truth. The white line in the ground truth image shows the slicing position. The training data were collected as windowed ($30 \times 30 \times 34$) HS-sub-cubes. The middle pixels of those HS-sub-cubes (250 lesion and 100 healthy skin pixels) were randomly selected from the left sides of the lesions. The middle image is the classification map of this BCC HS image. “RGB” illustration of the lesion is on the right. The lesion was captured from the left upper arm. The unit of measures is smaller with prediction maps since the HS images were windowed with 5 pixels step.

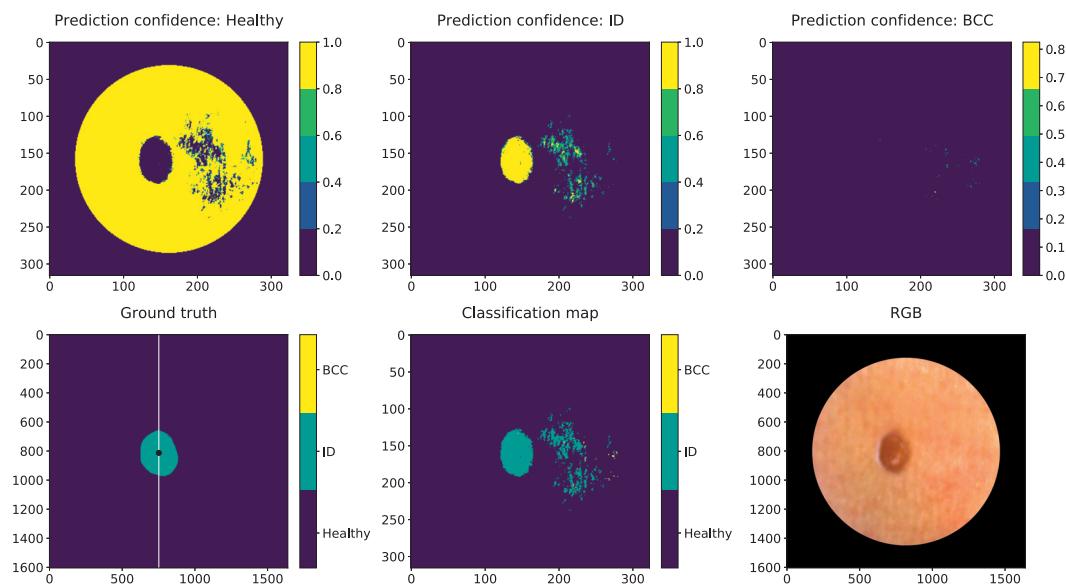


Figure 19. IN visual classification example shows that benign ID lesion pixels can be classified and the lesions can delineate efficiently. The classification map and ID prediction conference visualization show that some of the reflections or different shades of redness on the healthy skin might have confused the classification result. The lesion was captured from the right side of the patient's back. The unit of measures is smaller with prediction maps since the HS images were windowed with 5 pixels step.

3.4.3. Leave-One-Out Validation

This subsection compares the results obtained with a pixel-wise slice half model and 17 leave-one-out models. The results were obtained using subsets of the windowed ($30 \times 30 \times 34$) HS-sub-cube training data. For each lesion, the test data were selected from the right side of the lesions, which was not used to train any of the models. With leave-one-out validation, the test lesion pixel HS-sub-cubes were left out from the training subset cubes, and for the slice half model, the model was trained only using the HS-sub-cubes of pixels selected from the left side of the lesion.

Figure 20 visualises the pixel-wise classification accuracy comparison results for each lesion, which is the approach validation mentioned in Section 3.3.2. Y-axis represents the sliced data results, which model trained with 31,168 windowed HS-sub-cubes. The x-axis represents the results obtained with 17 leave-one-out models. The figure shows which lesions pixel-wise classification accuracies correlate, and indicates which lesions had unique features.

The leave-one-out validation results show how the leave-one-out models were sensitive towards special features in the training data. Some of the individual lesion's features and skin sub-types were unique. It affected the leave-one-out results decreasing it significantly. We can see from Table 4 that when the model is trained by leaving out windowed pixel HS-sub-cubes of a unique lesion, the model is unable to classify it correctly. It naturally has a strong descending effect on the leave-one-out average accuracy of all 17 models. For example, ID 17 (Table 4, 'RGB' image in Appendix A) was the only lesion covered with hair. The model with no hairy pixels in the test data could reach 0.26 accuracy, while a slice half model with hairy pixels in the training data could reach 0.94 accuracy.

Eight lesions can be considered typical without any special features. When the training data contained pixel-wise HS-sub-cubes collected from those lesions, leaving one lesion out, the leave-one-out average accuracy those eight models was 0.72. The corresponding average accuracy of the pixel-wise slice half approach with the same lesions was 0.89. Nine lesions had unique features, of which two lesions were unique and difficult for both approaches. When leaving the pixel-wise HS-sub-cubes of one of these unique lesions out

of the training data, the classification accuracy of these special lesions' windowed pixel HS-sub-cubes decreased significantly. The average accuracy of those nine models was 0.28 in the leave-one-out approach. Therefore, the average accuracy of all of the 17 leave-one-out models is affected by the unique features explained in Table 4.

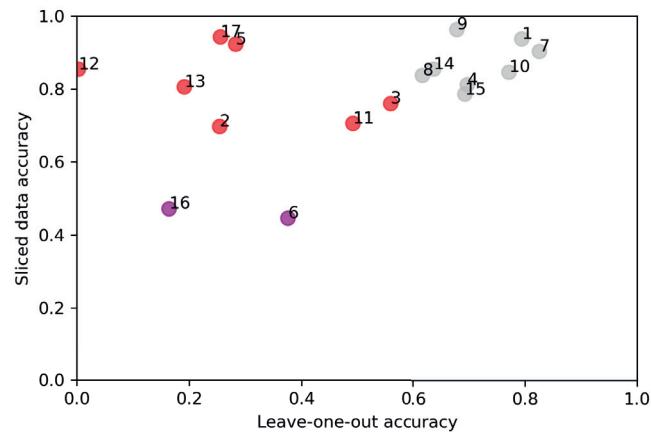


Figure 20. Comparison of pixel-wise accuracy results. Y-axis represents the results obtained with model trained and tested with sliced data. X-axis visualises the classification accuracy of leave-one-out models. The details of every HS image with RGB reconstructions can be seen in Appendix A. Grey, red and purple colours indicate the special features and the models' capability of classifying typical (grey), unique (red), difficult and unique (purple) lesions in this test set.

Table 4. Leave-one-out (L-O-O) and slice half pixel classification accuracy's.

Special Feature	Lesion ID	L-O-O	Slice Half	Unique Features
Typical	1	0.79	0.94	
Typical	9	0.68	0.96	
Typical	8	0.62	0.84	
Typical	7	0.83	0.90	
Typical	10	0.77	0.85	
Typical	4	0.70	0.81	
Typical	15	0.69	0.79	
Typical	14	0.64	0.85	
Average (typical)		0.72	0.89	
Unique	5	0.23	0.92	Broken skin, dark skin tone, convex surface
Unique	3	0.56	0.76	Broken skin, dark skin tone
Unique	2	0.25	0.68	Wound or scab on top of lesion
Unique	17	0.26	0.94	The lesion and the healthy skin is covered by hair
Unique	12	0.003	0.85	Dark skin tone and red-brown lesion tone
Unique	11	0.49	0.71	Clear pigment, small lesion
Unique	13	0.19	0.81	Two types of lesions: naevus pigmentosus junctionalis on right side of the ID
Unique & difficult	16	0.16	0.47	Overall fair skin tone, small lesion (2mm), fair toned and uneven delineation
Unique & difficult	6	0.38	0.45	Lesion is pigmented, uneven skin tone around lesion
Average (unique & difficult)		0.28	0.73	
Average (all samples)		0.47	0.81	

Table 4 and Figure 20 confirm that the results obtained with training data containing typical pixel window HS-sub-cubes from lesions with typical features among the training and test set, performed with significantly higher accuracy in the leave-one-out approach

than the results obtained with unique test data. The average performance of the leave-one-out approach with typical data (0.72) validates the results of the slice half pixel-wise approach (0.89), indicating correlation and the model's possibilities of generalization.

4. Discussion

The aim of this study was to introduce and demonstrate with a clinical pre-test a novel spectral imaging system designed for the complex skin surfaces. The pre-test demonstrated and enabled the system engineering, imaging, and analysis aspects of further system development toward a clinical application. The discussion is divided into six subsections, from technical and user-related topics (Section 4.1) to skin surface models (Section 4.2), annotation (Section 4.3), results (Section 4.5), approach validation (Section 4.6), bias conversation (Section 4.7) and finally to the notes for future research and the mentioned independent continuation studies (Section 4.8).

4.1. Technical and User-Related Issues

The SICRURFIS HSI had some issues related to the image quality or lack of healthy skin pixels with the smallest stray light protection cones. Those observations were technical, use-case related issues that can be solved in the future by changing the imaging strategy or by improving the device. With those user-related issues, some system engineering related topics were found.

The HS image capturing process was streamlined by only measuring wavebands relevant to the biophysical qualities of the skin and the illumination profiles of the used LEDs. However, the process still takes enough time, the minute movements of the medical professional using the device and the patient introduce a source for noise and inaccuracy. To solve this, image preview was implemented to the software's graphical user interface and by applying spectral smoothing to the data. However, some motion of the operators hands is to be expected with the setup, and it may have contributed negatively to the image quality. In further research, a way to lower the capturing time and reduce the strength needed to operate the camera stable could be found by using new HSI technology. Interferometers can be manufactured as a micro electro-mechanical system (MEMS) using novel atomic layer decomposition techniques. One of these prototypes was presented by Trops et al. (2091). MEMS-based HS imagers can be significantly smaller and lighter with a faster frame rate. Besides the HS imager's technical properties, the capturing time depends on the amount of the selected wavebands, so it could be further narrowed based on the suspected lesion type, making the image capturing faster.

Some of the images had quality issues which led to the decision to leave them out of the study. One of the problems was the sharpness of the images. The device required manual focusing, and some of the surfaces were challenging. For example, the nasal tip could be too small to the field of view, and the camera could not be set to the skin without passing some light from the sides of the nose. The unwanted light could be removed by covering the side areas with a hand, but it might make the manual focusing and staying on a place while capturing more difficult. Another focusing issue might be that the device is in sharp focus when the target is 6–6.5 cm away from the lens. Small and complex skin areas (e.g., nasal tip, ear) might come naturally closer than the required distance since the target might get inside the stray light protection cone. Similar quality issues have been raised in previous studies, e.g., in the study by Salmivuori et al. [10] uneven surfaces impaired the quality of the images.

The size of the stray light protection cone affected the usability of the images. Some of the images were captured with the smallest stray light protection cone. Those images had too small areas of healthy skin around the lesions. With the surface models, the best results were obtained with the stray light protection cone diameters of 4.9 and 5.8 cm. Additionally, the LED setup caused some reflections on the skin surface, which was difficult for our model to classify. For future studies, we would recommend capturing multiple pictures from the same patient, both images of lesions and separate images of only healthy skin.

That would enable using the smallest stray light protection cones in the study because the healthy pixels could be selected from a different image. The reflectivity of the skin is one issue to be assessed in future research. It could be solved with linear polarizers placed in front of the LEDs.

According to the user feedback, the imaging system was easy to use and the technical issues found can be improved with system engineering. As seen in the results, the SIC-SURFIS HSI reached complex surfaces, which were mentioned to be excluded in previous studies (e.g., [14]). Eye corners, chin, shoulders, ears, neck, arms and other challenging skin areas were reachable, and the pixel-wise classification results were promising.

4.2. About the Skin Surface Models

In the calculated surface models, the accuracy deteriorates as a function of the distance from the image center. In the border areas of the images, the effect is visible from the visualisations in Figure 14. The deterioration is an inherent feature of the used Frankot-Chellappa algorithm, as the light direction is defined only for the center of the imaging area. The poor accuracy may negatively affect the classification results, and it would be beneficial to study how to use individual light direction matrices for each point.

In our measurement setup, there are many moving pieces, and therefore the process of determining the individual light direction matrices was deemed inaccurate. The surface models are affected also by the penetration depth of the used light.

In this study, we decided to use wavelength from visible light (approx. 575 nm) as the basis of the surface model, and used it along the 33 albedo channels. The CNN's 2D layer extracted the features from this wavelength. The results are promising, but there is a need for further research on the benefits of the surface model's usage in HS image pixel classification. It could be beneficial to study the surface models of more penetrating wavelengths. If the density of the skin is significantly different between cancerous and healthy areas, the difference could be made visible with surface models formed with infrared lights.

4.3. The Annotation and Pixel Selection

The dermatologist drew the ground truth of the lesions and healthy skin. The poor signal-to-noise ratio of some of the RGB constructions might have affected on drawing of the edges of the annotated lesions. Therefore the rules of how we selected the pixels could be improved. We did not consider that there might be pixels close to the lesion edges that are incorrectly classified as healthy skin or lesion on the ground truth.

For future machine learning preprocessing, it might be necessary to select the healthy skin pixels as far as possible from the annotated lesion and leave some margin to the edges of the lesion. This challenge could also benefit from the purely healthy skin HS images captured from the same patients.

4.4. Notes from the Pixel-Wise Analysis

According to Ahmad et al., the lack of labelled HS data is a major issue since labelling is time-consuming and expensive due to human experts and investigation. In this pre-study, the HS images were annotated by dermatologists, and the ground truth was confirmed with dermatohistopathological investigation, which is a strength.

Since the CNN models have a tendency to overfit when the amount of the HS training data is too small, we selected a pixel-wise classification approach with CNN, which is a common strategy with HS data and deep neural networks [3]. The training data of the CNN was constructed by slicing the HS images into two images per HS image and using the left side as a source for the selected training data pixels and the right side as a source for the testing data pixels. Instead of training and testing the model with only 17 HS images, we could utilise the spatial, spectral and skin surface domains pixel-wise with spectral pixel sub-cubes. After the data augmentation, the training set consisted of 31,168 windowed pixels ($30 \times 30 \times 34$). Before the augmentation, from each lesion HS image, there were 250

lesion pixels and 100 healthy skin pixels, which were randomly selected as the middle points of those windowed HS-sub-cubes. The test data were collected similarly from the right sides of the lesion HS images. This way, we achieved satisfactory amount of data points for both training and testing.

The CNN seems to perform well in classifying the malignant BCC (nodular and superficial), benign ID and healthy skin lesion pixels. The nodular BCC's confidence map (Figure 17) shows that the model correctly does not find any ID pixels from the image data pixels. It draws pixel-by-pixel quite accurately the shape of the elevated lesion. The prediction confidence map shows some BCC pixels in the right upper corner, which are, according to dermatologists, false classification results. Nodular small (<1 cm) BCC are usually sharply demarcated. The width of the lesion was 5 mm. The lesion is located on the upper eyelid. The skin around the lesion has multiple colours in the image, and there are some wrinkles and light reflections. The model has classified some of the healthy skin pixels with reflections and areas with healthy skin pixels containing skin colour changes from dark to light reflection as BCC pixels.

The second example collage (Figure 18) is a 12 mm nodular and superficial BCC on the left upper arm. According to the dermatologist, the superficial parts are usually in the periphery of the lesion and can have indistinct borders. The CNN draws a BCC lesion pixel-wise with indistinct edges and some satellite lesions around it. There is a possibility that satellite lesions can occur in the periphery of a superficial BCC that are impossible to detect by the human eye. These classified lesion pixels in this study might indicate that with HS imaging and machine learning methods, it could be possible to provide information that guides the dermatologists to delineate and remove a lesion more accurately. SICSURFIS HSI's pixel resolution is approximately $24\text{ }\mu\text{m} \times 24\text{ }\mu\text{m}$, one classified pixel is smaller than one cell, which can be seen as an advance for applications requiring high accuracy.

Third collage (Figure 19) is an example of a benign intradermal nevus pixel classification results. The lesion is located on the right side of the back of the patient. The lesion is delineated and classified accurately, but there are redness and reflections in the pixels surrounding of the lesion that are miss-classified as nevus pixels. Intradermal nevi are sharply demarcated, and no satellite lesions should be seen. Some of the pixel level challenges relate to the healthy skin pixels. In the future, larger samples of HS images of healthy skin could improve the classification results of healthy skin.

4.5. Sensitivity and Precision

The numerical results of the pixel-wise slice half model's test data (Table 3) show relatively good weighted sensitivity (0.79) and precision (0.81) for the model.

In our study, the sensitivity is mainly impaired by inaccurate delineation of the lesions, which again seem to be caused mostly by light reflections and uneven colour of the healthy skin in the images. The sensitivity is also decreased because the sensitivity calculations compare the diagnosis separately for each pixel of a lesion, not by a voting method with one diagnosis per lesion only. Our test data pixels were selected randomly from the right side of the HS images lesion and healthy skin areas, and since the CNN is a three class pixel-classifier, the results of those pixels may contain two types of lesion or healthy skin pixels.

These sensitivity and precision results are not directly comparable with previous studies (e.g., [36]), which uses the majority voting method on the annotated lesion areas. In the majority voting approach, the whole area of the annotated lesion is pixel-wise classified based on the majority of the pixels, so there cannot be, for example, two types of lesion pixels on the same lesion area.

As an important note, the used data contain an inherit selection bias which in some cases increases and in some cases decreases the accuracy. The slice half method also cause bias by increasing the the obtained accuracy. Therefore, the results are promising, but further research is needed. In the further studies, the results could be addressed also with a majority voting method, which would enable more discussion with previous research.

4.6. Approach Validation

Besides the promising results, this kind of clinical pre-study has limitations and concerns related to approaches and methods. The pixel-wise classification approach, using only one HS cube as a source for training, validation and testing is a common approach in HS image analysis and classification studies [3] (examples in [1,37,38]).

One of the possible downsides of the pixel-wise slice half approach is that after the training, the neural network has been presented with data very similar to the test data, which might bias the results. For observing the possible bias, the pixel-wise slice half classification results were validated with a pixel-wise leave-one-out approach.

As a result, the leave-one-out validation metrics were strongly decreased with unique lesion features (see Table 4, Figure 20 and Appendix A). Only the results obtained with models trained with typical data should be considered a validation; the leave-one-out models trained with data containing windowed pixels (HS-sub-cubes) from the eight typical lesions reached 0.72 average accuracy, varying between 0.83 and 0.64. The average accuracy of the same eight typical lesion's pixel-wise HS-sub-cubes with the model trained with slice-half approach and 31,168 HS-sub-cubes was 0.89, varying between 0.96 and 0.79. This finding supports the potential results of SICSURFIS HSI's capability to classify, differ, and delineate the BCC, ID, and healthy skin pixels, and validates the approach.

The decreasing effect in the leave-one-out results is visible in Figure 20 and Table 4. We can see that the accuracy of the pixel-wise test data (HS-sub-cubes) collected from lesions 1, 4, 7, 8, 9, 10, 14 and 15 were correlating, which indicates that those lesions had some typical pixel-level features that enable both approaches models to generalise. The effect can be seen with nine lesions; the pixel-wise test data collected from lesions 2, 3, 5, 6, 11, 12, 13, 16 and 17 had some special features, making the lesions unique in the training set. Therefore, nine of the seventeen leave-one-out models could not generalise well. These models could not classify a pixel that was vastly different from the training data set. For example, there was only one hairy lesion, only one red-brown lesion or a single lesion with a scab. Those features in the collected pixel-wise HS-cub-cubes could be a reason for a model not to be able to classify pixels containing those features correctly. Two of the lesions, 6 and 16, were difficult and unique, which can be seen in the pixel-wise accuracy of both approaches.

The approach validation shows that the results obtained with eight typical lesions and both approaches are promising. The level of bias is acceptable for a study that has the purpose of the first demonstration of a prototype imager and points out the future improvement needs. The slice half results are higher, and the leave-one-results lower, but the model's capability to generalise with unbiased data can be seen, but further studies and more data are still needed.

For closer investigation of the results and validation, the Appendix A presents the RGB-reconstructions, histopathologically confirmed ground truth, accuracy results of both tests, and clinical details (lesion type, size, location) and possible mentions of special features. Based on the special features that are visible in RGB reconstructions, the lesions were divided into three classes, grey (typical), red (unique) and purple (unique and difficult). The colors and image IDs match with the results shown in Figure 20 and Table 4.

4.7. Selection Bias and Other Data-Based Biases and Their Effect to the Results

Selection bias is “A systematic error that results in differences between a study population and a target population; selection bias primarily affects the external validity of the results of a study” [39]. In this study, the selection bias concerns the study population and study lesions. The variability of the real target population (humans all over the world) is not fully covered and the variability of different sub-types of lesions (i.e., hairy, growth style, different pigment variations, wounds, blood, thin or thick skin etc.) has the same low coverage.

When developing an application to clinical imaging, the selection bias is probably the most important reason for a system with high accuracy on pre-tests to be unreliable in clinical use [39]. This effect has been noticed for example in machine learning algorithms utilised with radiology images. The recent study points out the selection bias as the

possible reason, why the results at the hospital with real patients are less expressive [39]. This research, and the SICSURFIS HSI, as it is, is a long way from a clinical application. According to Yu and Eng, the selection bias has been largely unaddressed in the medical imaging machine learning literature. Bias in the collected data causes distortion among the results. For example, the classification accuracy can increase or decrease, but the level of bias can and should be examined and controlled [39–41]. Untrained unique features decreases the accuracy and features that are covered in the training data enables higher generalisation and accuracy.

As shown in this study, the bias is difficult to avoid. It can be caused by a method or data. The development of a real-life medical application is expensive. There are plenty of sources of bias that affect any results: patient skin tone (native Asian, north European, African vs south European), clinical practices, patient age distribution, users of the imagers and so on. If the inherent bias is not caused by a method, it is caused by a data set. The level of bias is important to estimate, and further research should address decreasing it. Deep learning algorithms, such as CNN, is detecting and distinguishing features automatically, giving them weights. Basically, it is a black box, which does not explain why the diagnosis is made [39]. In this study, the selection bias is obvious; the number of lesions were relatively low, and the selected lesions had strong variation. Those, that had unique features failed on the leave-one-analysis, showing, that in order to develop a more robust system and a real application, it would require a enormous amount of data, captured in many countries, with a wide diverse among the patients and remarkable amounts of lesions.

Traditional image classification is different than pixel-classification mainly due to the spectral domain, and the fact that each pixel is classified one-by one. Previous image classification studies show, that in image classification, it is possible to train the model and classify sub-images with no sensible, visually interpret information, and depending on the sub-images size, to obtain even good accuracy levels. The reason of this is, that the model might be able to detect and weight features also from the background; use other visual aspects the human eye does not see features [40]. The size of the sub window seemed to correlate with the accuracy. With all tested known standard data sets and regardless of the level of bias factors, the small (20×20) subwindows reached lower accuracy than the larger ones (200×200) [40].

In our case, we used hyperspectral data and a pixel-wise classification approach. Each sub-cube contained albedo and depth channels, providing a spectrum for each pixel. Most of the training data were augmented, e.g., flipped, which differs the selected $30 \times 30 \times 34$ sub-cubes from the originally selected training pixels. The test pixels were not flipped. The size of the original training set was 6160 samples and after the augmentation, the size was 31,168. Therefore, we utilised a small-sized samples from each HS cube to control the possible data-based bias levels, and we gave pixel-wise information for CNN to extract with 3D and 2D layers. This causes bias, since the source of test and training pixels is the same data set (HS cube from a lesion), but as shown in the above mentioned previous studies, the small size of the used sub-cubes was chosen to keep the level low.

The sources for bias are various. For example if several data sets are collected in a controlled environment and same session, all of these HS images might contain characteristics that are caused by the image acquisition. So far, the machine learning model might classify them based on the session, not by the features seen in images [40]. In our data, this can be one of the bias reasons besides the selection bias; a subtle changes in the lighting conditions (a stray light passes the cones) or even the temperature of the sensor at the time of imaging might have a result in artifacts that might lead to differences that are to be detected by machine learning models [40]. As an example, the stray light increases the illumination when imaging a lesion, and a global feature, such as pixel intensity statistics, can be used as a decision feature inside the CNN. This phenomena was controlled with the protection cones, but some of the surfaces were challenging and the data might contain individual images with features caused by the stray light. An other example is that if the imager is in use for several hours, the sensor temperature increases, and it can cause

some features in the collected data; the dark-correction is performed for each HS cube. Dark reference images captured right before capturing the lesions, with similar exposure time, so the correction includes the current effect of sensor temperature. This might cause difference among the captured data, some of the HS images might have characteristics from the acquiring process, which influences to the results. One of the benefits of the randomly selected pixels and sub-cubes is, that if the sensor has dead pixels, that exists in every frame, this kind of feature is not automatically presented to the machine learning model.

As shown by this discussion, the perceived worth of this study might be diminished by the bias. However, the reason for this clinical pre-study was to introduce and pre-test a new imaging system. Without the first models and results, it would have been impossible to evaluate the technical aspects; how the imager reaches the complex surfaces, can those images be classified by the CNN. It was important to test, how the imager's special LED module, stray light protection and optical components affect on the image quality and to the results, and finally evaluate the clinical procedures. Based on this study, for example, the smallest protection cone has been rejected from the following data gathering steps and the stray light blocking in the challenging areas is improved with extra protection. One could say, that the accuracy of the leave-one-out approach is the real performance, but it is not the whole truth. Another opinion is, that due to the unseen features, the leave-one-out results reflect the selection bias on the collected data.

The main point is not, what interpretation of the approaches and results is the right or how much and what kind of bias the collected data contains. The outcome is more in the potential, that the SICSRFIS HSI has, and in the results that can be obtained in the future via improvements. The results of the both approaches indicates that the proposed models would be practical with a larger amount of training data, which contains no unique lesions, and so far the selection bias could be minimised. The hypothesis for future studies is that the accuracy will increase when the collected data enables the models to generalise better. The accuracy difference between the leave-one-out method and the slice half method is expected to decrease as the data amount increases, and the level of selection bias should be taken into account in the further studies.

4.8. Future Research

In the following studies, we will increase the amount and variability of the data, improve the technical and user-related issues and capture more lesion types for developing a more effective CNN classifier to get one step closer to an optical biopsy. Current results are promising, but the need for further studies is obvious.

5. Conclusions

This article aimed to introduce and demonstrate a Fabry–Perot interferometer-based hyperspectral imaging system for complex surfaces. This study was the first step in a three-phase pilot study demonstrating the possibilities of using a new system as a first prototype for future real-life applications. This section concludes our findings in technical aspects (Section 5.1), methodological approaches (Section 5.2) and results, and future research (Section 5.3).

5.1. Technical Aspects

As the described device is a prototype, there are still some issues to fix before achieving results that promote the devices use in clinical use. There is some variation in the focus between the spectral channels. Some of the frames might be slightly more out of focus than others. Bigger light protection cones provided better quality data than the smaller ones. There might not be enough healthy skin around the lesions with the small cones. The LED setup caused some reflections to the skin areas, which were difficult to handle for the convolutional neural network model. The depth maps were calculated for each wavelength, and one of the noted issues was the quality of the 3D models. The best quality was in the middle of the image. The quality deteriorated towards the edges of the frames.

It is possible to improve the depth of the focus in the whole wavelength range by replacing the used commercial S- and C-mount lenses with a dedicated custom-designed optical system. The specular reflections can be reduced by using linear polarizers in front of the LEDs and the imaging optics.

5.2. Classification Results and Method Validation

Based on the pre-test, the HS imager and machine learning system accurately differentiate malignant BCC from benign ID and healthy skin, achieving a weighted sensitivity of 0.79 and weighted precision of 0.81. The classification report reveals the results of the entire test data pixel-by-pixel classification. The results are not directly comparable with previous studies using voting methods for classifying pixel-wise the lesion types.

We used a pixel-wise classification approach with CNN classifier. The approach is common with HS images, due to the lack of labelled HS data. The number of HS cubes used in this study was 17, which is generally 16 HS images more than in a typical HS image classification method development experiments; it is typical to use the pixels from one standard data set, and divide them to training, validation and test portions [3]. In our approach, we sliced each of the lesion HS images and selected the training data from the left sides of the lesions. The training data consisted of 31,168 windowed pixels (HS-sub-cubes). Our test data consisted of similarly collected windowed pixels (HS-sub-cubes), selected randomly from the right side of the HS images lesion and healthy skin areas. The bright side of the pixel-wise approach is the large amount of the training data since it consists of sub samples. The CNN was not overfitted due to lack of training data, but the approach might bias the results. Therefore, the results were validated with a pixel-wise leave-one-out approach. Eight of the lesions consisted of pixel windows with typical features, providing relatively well-generalising models and correlating accuracy results with the sliced half approach. The rest of the leave-one-out models performed poorly due to special features in those lesions. Therefore, the average accuracy results of all of the 17 models in the validation were lower. For example, a model trained without hairy pixels could not accurately classify pixels collected from a hairy lesion image. These phenomena were seen with nine lesions and models that used the pixels collected from those lesions as test data, representing those unique features.

For the comparison of validation to the slice half approach, only the results of the eight typical lesions should be taken into account. The average accuracy of those results was 0.72, whereas the similar average accuracy with the slice half approach was 0.89. These results are shown in Table 4 and Figure 20. The validation confirms that this 3D approach has promising results, but there is a need for further studies with a larger amount of data with no unique images. The bias caused by the approach is seen in the slice half method, but the capability of models to generalise was proven with the validation approach. The accuracy results without bias might be lower than the achieved, but the results of this study might indicate that with HS imaging and machine learning methods, it could be possible to provide information that guides dermatologists to delineate and thereby remove a lesion more accurately. On the other hand, the number of captured lesions was small, containing unique lesions, impacting the results' generalisation.

5.3. Conclusions for the Future

This was a development starting point and a demonstration of a novel SICSURFIS HSI system, which seems to have potential, but the need for further studies is obvious. With its results and notes, the pre-test pointed out benefits and findings for many future improvement aspects.

COVID-19 had a decreasing effect on the data gathering phase's patient recruitment. As the system uses unique, specifically selected LEDs and wavelength bands, most of the previously gathered HSI data are inappropriate for training the machine learning algorithms used in the research. Therefore, data gathering with a similar device should continue in the future.

As the dataset size increases, the leave-one-out cross-validation method should be adopted. The accuracy difference between the leave-one-out method and the slice half method is expected to decrease as the data amount increases. There is no risk for cross-contamination between training and testing data in the leave-one-out method.

Before the clinical use, all of these issues should be inspected. Another technical step toward the future might be the MEMS FPI, which could reduce the size and weight of the handheld device. There is also a need for developing the system to be a real-time solution. It might mean the FPGA computation and a screen on top of the device for the doctors to see the results immediately.

Author Contributions: Conceptualization, I.P., H.S., A.R.; methodology, I.P., L.A., A.-M.R.-H.; software, A.-M.R.-H. and L.A.; embedded software, R.T. and A.N.; electrical and opto-mechanical design R.T. and A.N.; validation, A.-M.R.-H. and L.A.; formal analysis, A.-M.R.-H. and L.A.; investigation, A.-M.R.-H. and L.A., V.L.; resources, V.L., A.-M.R.-H., L.A., H.S., A.N., R.T. and I.P.; data curation, A.-M.R.-H. and L.A.; writing—original draft preparation, A.-M.R.-H.; writing—review and editing, A.-M.R.-H. and L.A., V.L.; visualization, A.-M.R.-H., L.A. and H.S.; supervision, I.P., H.S., A.R.; project administration, I.P., H.S., A.R.; funding acquisition, I.P., H.S., A.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Academy of Finland, grant numbers 314519, 314520 and 314521.

Institutional Review Board Statement: The study was conducted in accordance with the Declaration of Helsinki and approved by the Institutional Review Board of Helsinki University Hospital (protocol code HUS/2089/2019, date of approval 30 July 2019).

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: The data presented in this study are not publicly available or available upon request due to ethical and privacy reasons.

Acknowledgments: Sari Koskenmies, Kirsil Isoherranen, Sari Pitkänen, Mari Salmivuori, Leila Jeskanen, Jenni Ylisaari, Kimmo Riihiaho, Jarmo Alander.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

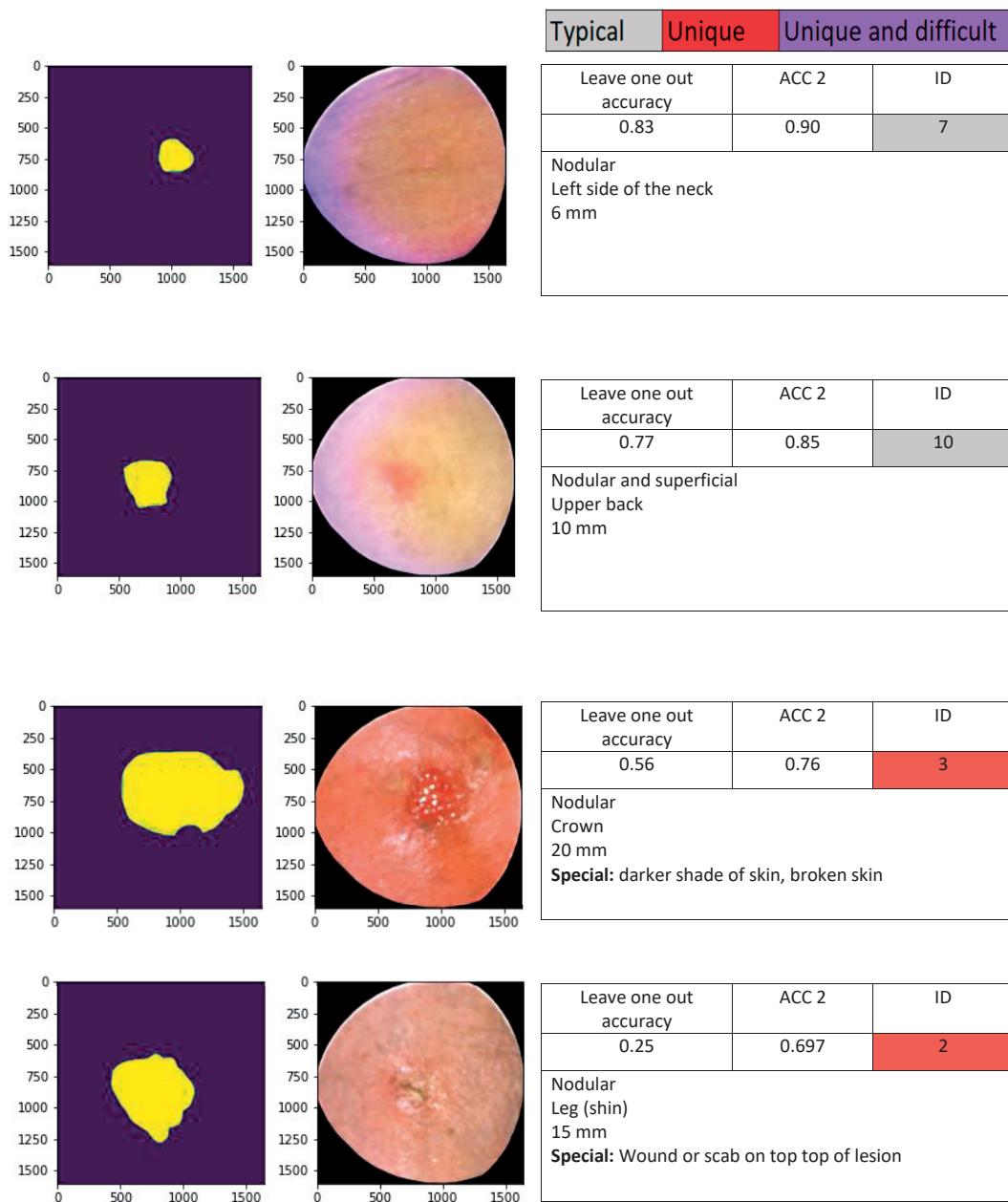
Appendix A

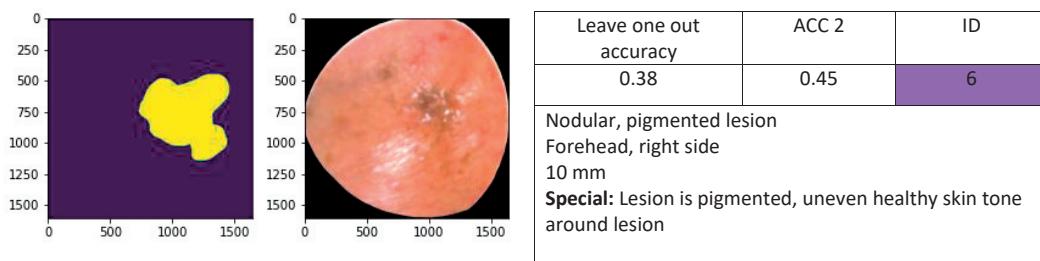
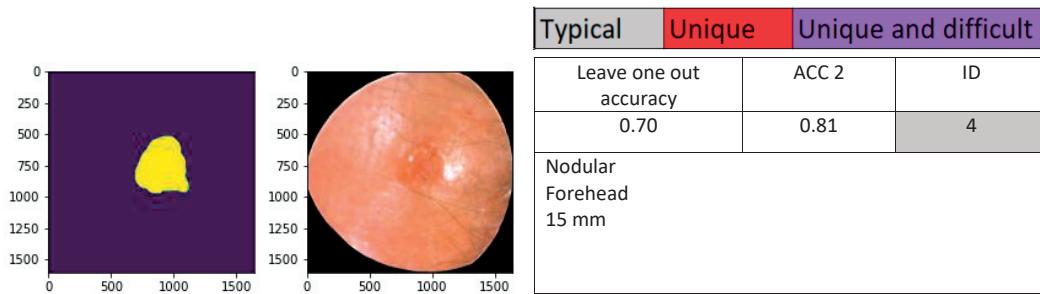
Bacall Cell Carcinomas (10 HS images)

Typical	Unique	Unique and difficult
Leave one out accuracy 0.79	ACC 2 0.94	ID 1
Nodular and superficial Left upper arm 12 mm		
Leave one out accuracy 0.68	ACC 2 0.96	ID 9
Superficial Back 10 mm		
Leave one out accuracy 0.62	ACC 2 0.84	ID 8
Nodular and superficial Right upper eyelid 4 mm		
Leave one out accuracy 0.23	ACC 2 0.92	ID 5
Nodular Left corner of eye 7 mm Special: Broken skin, darker skin shade, convex skin surface		

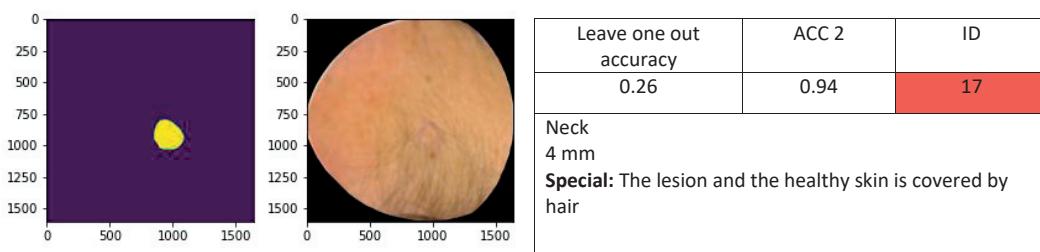
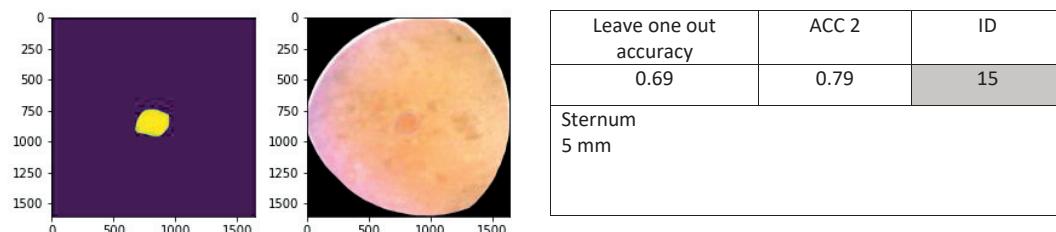
The figure consists of four rows, each containing two side-by-side images. The left image is a heatmap with a yellow blob indicating the lesion location, and the right image is a corresponding high-resolution skin image. To the right of each pair of images is a table with diagnostic information.

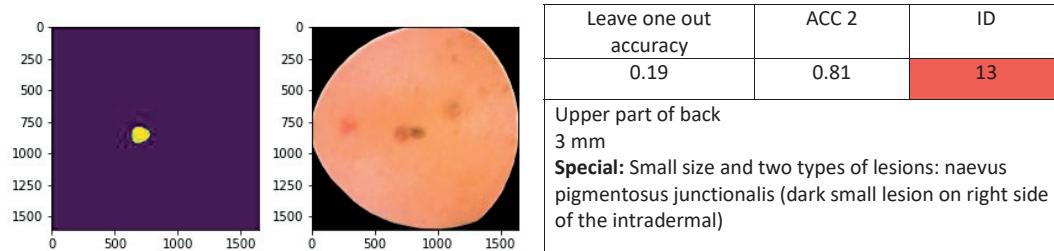
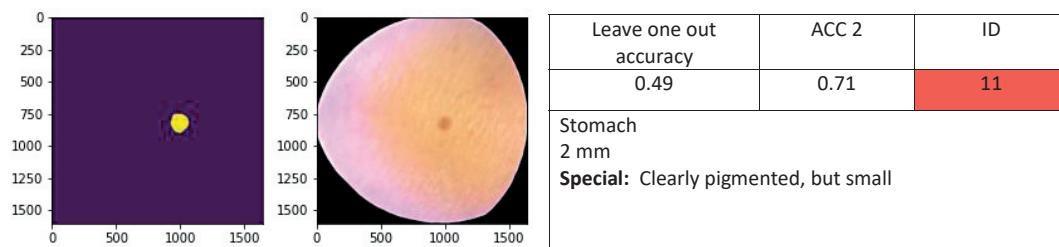
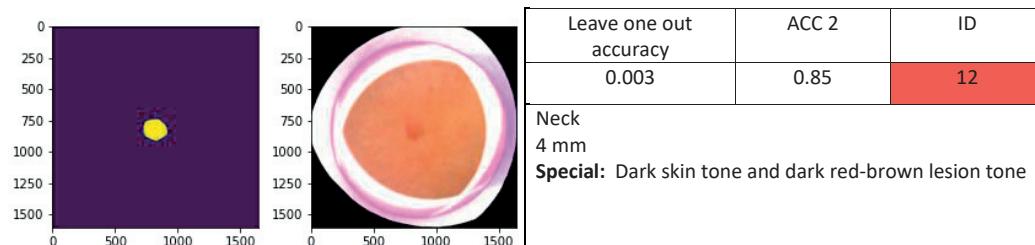
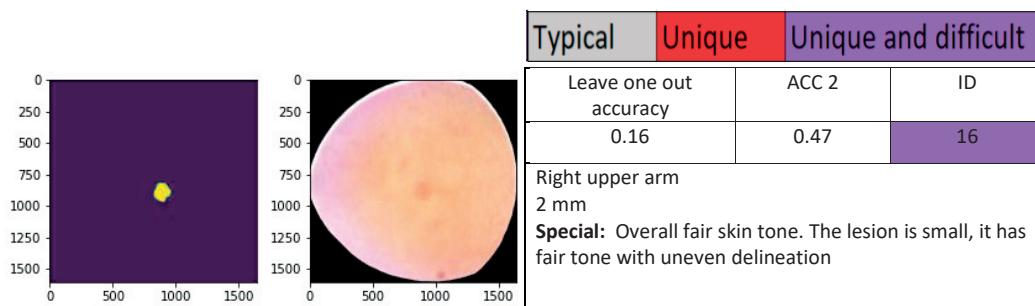
- Row 1:** HS image (0-1500) and heatmap (0-1500). Table: Leave one out accuracy 0.79, ACC 2 0.94, ID 1. Description: Nodular and superficial, Left upper arm, 12 mm.
- Row 2:** HS image (0-1500) and heatmap (0-1500). Table: Leave one out accuracy 0.68, ACC 2 0.96, ID 9. Description: Superficial, Back, 10 mm.
- Row 3:** HS image (0-1500) and heatmap (0-1500). Table: Leave one out accuracy 0.62, ACC 2 0.84, ID 8. Description: Nodular and superficial, Right upper eyelid, 4 mm.
- Row 4:** HS image (0-1500) and heatmap (0-1500). Table: Leave one out accuracy 0.23, ACC 2 0.92, ID 5. Description: Nodular, Left corner of eye, 7 mm. Special note: Broken skin, darker skin shade, convex skin surface.

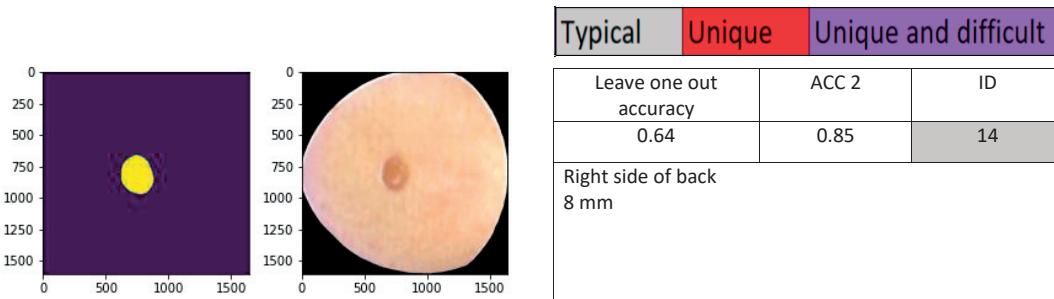




Intradermal Nevus Lesions (7 HS images)







References

1. Camps-Valls, G.; Bruzzone, L. Kernel-based methods for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2005**, *43*, 1351–1362. <https://doi.org/10.1109/TGRS.2005.846154>.
2. Bioucas-Dias, J.M.; Plaza, A.; Camps-Valls, G.; Scheunders, P.; Nasrabadi, N.M.; Chanussot, J. Hyperspectral remote sensing data analysis and future challenges. *IEEE Geosci. Remote Sens. Mag.* **2013**, *1*, 6–36. <https://doi.org/10.1109/MGRS.2013.2244672>. Hyperspectral Image Classification—Traditional to Deep Models: A Survey for Future Prospects. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2022**, *15*, 968–999. <https://doi.org/10.1109/JSTARS.2021.3133021>.
3. Ahmad, M.; Shabbir, S.; Roy, S.K.; Hong, D.; Wu, X.; Yao, J.; Khan, A.M.; Mazzara, M.; Distefano, S.; Chanussot, J. Hyperspectral Image Classification—Traditional to Deep Models: A Survey for Future Prospects. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *11*, 968–999.
4. Saari, H.; Pöölönen, I.; Salo, H.; Honkavaara, E.; Hakala, T.; Holmlund, C.; Mäkinen, J.; Mannila, R.; Antila, T.; Akujärvi, A. Miniaturized hyperspectral imager calibration and UAV flight campaigns. In *Sensors, Systems, and Next-Generation Satellites XVII*; International Society for Optics and Photonics: Bellingham, WA, USA, 2013; Volume 8889.
5. Halicek, M.; Fabelo, H.; Ortega, S.; Callico, G.M.; Fei, B. In-Vivo and Ex-Vivo Tissue Analysis through Hyperspectral Imaging Techniques: Revealing the Invisible Features of Cancer. *Cancers* **2019**, *11*, 756. <https://doi.org/10.3390/cancers11060756>.
6. Aboughaleb, I.H.; Aref, M.H.; El-Sharkawy, Y.H. Hyperspectral imaging for diagnosis and detection of ex-vivo breast cancer. *Photodiagnosis Photodyn. Ther.* **2020**, *31*, 101922. <https://doi.org/10.1016/j.pdpdt.2020.101922>.
7. Jansen-Winkel, B.; Barberio, M.; Chalopin, C.; Schierle, K.; Diana, M.; Köhler, H.; Gockel, I.; Maktabi, M. Feedforward Artificial Neural Network-Based Colorectal Cancer Detection Using Hyperspectral Imaging: A Step towards Automatic Optical Biopsy. *Cancers* **2021**, *13*, 967. <https://doi.org/10.3390/cancers13050967>.
8. Neittaanmäki-Perttu, N.; Grönroos, M.; Jeskanen, L.; Pöölönen, I.; Ranki, A.; Saksela, O.; Snellman, E. Delineating margins of lentigo maligna using a hyperspectral imaging system. *Acta Derm.-Venereol.* **2015**, *95*, 549–552.
9. Zheludev, V.; Pöölönen, I.; Neittaanmäki-Perttu, N.; Averbuch, A.; Neittaanmäki, P.; Grönroos, M.; Saari, H. Delineation of malignant skin tumors by hyperspectral imaging using diffusion maps dimensionality reduction. *Biomed. Signal Process. Control* **2015**, *16*, 48–60.
10. Salmivuori, M.; Neittaanmäki, N.; Pöölönen, I.; Jeskanen, L.; Snellman, E.; Grönroos, M. Hyperspectral imaging system in the delineation of ill-defined basal cell carcinomas: A pilot study. *J. Eur. Acad. Dermatol. Venereol.* **2019**, *33*, 71–78.
11. Johansen, T.H.; Møllersen, K.; Ortega, S.; Fabelo, H.; Garcia, A.; Callico, G.M.; Godtliebsen, F. Recent advances in hyperspectral imaging for melanoma detection. *Wiley Interdiscip. Rev. Comput. Stat.* **2020**, *12*, e1465.
12. Dicker, D.T.; Lerner, J.; Van Belle, P.; Guerry, 4th, D.; Herlyn, M.; Elder, D.E.; El-Deiry, W.S. Differentiation of normal skin and melanoma using high resolution hyperspectral imaging. *Cancer Biol. Ther.* **2006**, *4047*, 4–10. <https://doi.org/10.4161/cbt.5.8.3261>.
13. Nagaoka, T.; Nakamura, A.; Kiyohara, Y.; Sota, T. Melanoma screening system using hyperspectral imager attached to imaging fiberscope. In Proceedings of the 2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, San Diego, CA, USA, 28 August–1 September 2012; Volume 30, pp. 3728–3731. <https://doi.org/10.1109/EMBC.2012.6346777>.
14. Leon, R.; Martinez-Vega, B.; Fabelo, H.; Ortega, S.; Melian, V.; Castaño, I.; Carretero, G.; Almeida, P.; Garcia, A.; Quevedo, E.; et al. Non-Invasive Skin Cancer Diagnosis Using Hyperspectral Imaging for In-Situ Clinical Support. *J. Clin. Med.* **2020**, *9*, 1662.
15. Pöölönen, I.; Rahkonen, S.; Annala, L.; Neittaanmäki, N. Convolutional neural networks in skin cancer detection using spatial and spectral domain. In *Photonics in Dermatology and Plastic Surgery 2019*; International Society for Optics and Photonics: Bellingham, WA, USA, 2019; Volume 10851, p. 108510B.
16. Frankot, R.T.; Chellappa, R. A method for enforcing integrability in shape from shading algorithms. *IEEE Trans. Pattern Anal. Mach. Intell.* **1988**, *10*, 439–451.
17. Barun, V.; Ivanov, A.; Volotovskaya, A.; Ulashchik, V. Absorption spectra and light penetration depth of normal and pathologically altered human skin. *J. Appl. Spectrosc.* **2007**, *74*, 430–439. <https://doi.org/10.1007/s10812-007-0071-2>.
18. Rey-Barroso, L.; Peña-Gutiérrez, S.; Yáñez, C.; Burgos-Fernández, F.J.; Vilaseca, M.; Royo, S. Optical Technologies for the Improvement of Skin Cancer Diagnosis: A Review. *Sensors* **2021**, *21*, 252.
19. Nagaoka, T.; Nakamura, A.; Okutani, H.; Kiyohara, Y.; Sota, T. A possible melanoma discrimination index based on hyperspectral data: A pilot study. *Skin Res. Technol.* **2012**, *18*, 301–310.

20. Tishby, N.; Zaslavsky, N. Deep learning and the information bottleneck principle. In Proceedings of the 2015 IEEE Information Theory Workshop, ITW 2015, Jeju Island, Korea, 11–15 October 2015. <https://doi.org/10.1109/ITW.2015.7133169>.
21. Courtenay, L.A.; González-Aguilera, D.; Lagüela, S.; del Pozo, S.; Ruiz-Mendez, C.; Barbero-García, I.; Román-Curto, C.; nueto, J.C.; Santos-Durán, C.; noso Álvarez, M.E.C.; et al. Hyperspectral imaging and robust statistics in non-melanoma skin cancer analysis. *Biomed. Opt. Express* **2021**, *12*, 5107–5127. <https://doi.org/10.1364/BOE.428143>.
22. Zherebtsov, E.; Dremin, V.; Popov, A.; Doronin, A.; Kurakina, D.; Kirillin, M.; Meglinski, I.; Bykov, A. Hyperspectral imaging of human skin aided by artificial neural networks. *Biomed. Opt. Express* **2019**, *10*, 3545–3559. <https://doi.org/10.1364/BOE.10.003545>.
23. Mäkinen, J.; Holmlund, C.; Saari, H.; Ojala, K.; Antila, T. Unmanned aerial vehicle (UAV) operated megapixel spectral camera. *Electro-Opt. Remote Sensing, Photonic Technol. Appl. V* **2011**, *8186*. <https://doi.org/10.1117/12.897712>.
24. Woodham, R.J. Photometric stereo: A reflectance map technique for determining surface orientation from image intensity. In *Image Understanding Systems and Industrial Applications I*; International Society for Optics and Photonics: Bellingham, WA, USA, 1979; Volume 155, pp. 136–143.
25. Trops, R.; Hakola, A.M.; Jääskeläinen, S.; Näsilä, A.; Annala, L.; Eskelinen, M.A.; Saari, H.; Pöllönen, I.; Rissanen, A. Miniature MOEMS hyperspectral imager with versatile analysis tools. In *MOEMS and Miniaturized Systems XVIII*; International Society for Optics and Photonics: Bellingham, WA, USA, 2019; Volume 10931, p. 109310W.
26. University of Jyväskylä, Spectral Imaging Laboratory. Available online: <https://www.jyu.fi/it/en/research/our-laboratories/> (accessed on 4 May 2021).
27. Jääskeläinen, S.; Eskelinen, M.; Annala, L.; Raita-Hakola, A.M. Camazing Python Library. Available online: <https://pypi.org/project/camazing/> (accessed on 4 May 2021).
28. Jääskeläinen, S.; Eskelinen, M.; Annala, L.; Raita-Hakola, A.M. Spectacular Python Library. Available online: <https://www.jyu.fi/it/en/research/our-laboratories/hsi/contact-info> (accessed on 4 May 2021).
29. Goldstein, T.; Osher, S. The split Bregman method for L1-regularized problems. *SIAM J. Imaging Sci.* **2009**, *2*, 323–343.
30. Varoquaux, G.; Buitinck, L.; Louppe, G.; Grisel, O.; Pedregosa, F.; Mueller, A. Scikit-learn. *GetMobile Mob. Comput. Commun.* **2015**, *19*, 29–33. <https://doi.org/10.1145/2786984.2786995>.
31. van der Walt, S.; Schönberger, J.L.; Nunez-Iglesias, J.; Boulogne, F.; Warner, J.D.; Yager, N.; Gouillart, E.; Yu, T. Scikit-image: Image processing in Python. *PeerJ* **2014**, *2*, e453. <https://doi.org/10.7717/peerj.453>.
32. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
33. Virtanen, P.; Gommers, R.; Oliphant, T.E.; Haberland, M.; Reddy, T.; Cournapeau, D.; Burovski, E.; Peterson, P.; Weckesser, W.; Bright, J.; van der Walt, S.J.; et al. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nat. Methods* **2020**, *17*, 261–272. <https://doi.org/10.1038/s41592-019-0686-2>.
34. Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G.S.; Davis, A.; Dean, J.; Devin, M.; et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems, 2015. Software. Available online: tensorflow.org (accessed on 1 May 2021).
35. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016. Available online: <http://www.deeplearningbook.org> (accessed on 1 January 2022).
36. Räsänen, J.; Salmivuori, M.; Pöllönen, I.; Grönroos, M.; Neittaanmäki, N. Hyperspectral Imaging Reveals Spectral Differences and Can Distinguish Malignant Melanoma from Pigmented Basal Cell Carcinomas: A Pilot Study. *Acta Derm.-Venereol.* **2021**, *101*, adv00405.
37. Camps-Valls, G.; Gomez-Chova, L.; Munoz-Mari, J.; Vila-Frances, J.; Calpe-Maravilla, J. Composite kernels for hyperspectral image classification. *IEEE Geosci. Remote Sens. Lett.* **2006**, *3*, 93–97. <https://doi.org/10.1109/LGRS.2005.857031>.
38. Yang, X.; Ye, Y.; Li, X.; Lau, R.Y.K.; Zhang, X.; Huang, X. Hyperspectral Image Classification With Deep Learning Models. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 5408–5423. <https://doi.org/10.1109/TGRS.2018.2815613>.
39. Yu, A.C.; Eng, J. One Algorithm May Not Fit All: How Selection Bias Affects Machine Learning Performance. *RadioGraphics* **2020**, *40*, 1932–1937. <https://doi.org/10.1148/rg.2020200040>.
40. Model, I.; Shamir, L. Comparison of Data Set Bias in Object Recognition Benchmarks. *IEEE Access* **2015**, *3*, 1953–1962.
41. Paullada, A.; Raji, I.D.; Bender, E.M.; Denton, E.; Hanna, A. Data and its (dis)contents: A survey of dataset development and use in machine learning research. *Patterns* **2021**, *2*, 100336. <https://doi.org/10.1016/j.patter.2021.100336>.

PV

**DIFFERIANTING MALIGNANT FROM BENING PIGMENTED
OR NON-PIGMENTED SKIN TUMORS - A PILOT STUDY ON
3D HYPERSPECTRAL IMAGING AND CONVOLUTIONAL
NEURAL NETWORKS**

by

Lindholm Vivian, Raita-Hakola Anna-Maria, Salmivuori Mari, Jeskanen Leila,
Saari Heikki, Koskenmies Sari, Pitkänen Sari, Pölönen Ilkka, Isoherranen Kirsi
and Ranki Annamari 2022

MDPI - Journal of Clinical Medicine, Volume 11, Number 7, Pages 1914,
<https://doi.org/10.3390/jcm11071914>

Reproduced with kind permission of MDPI.

Article

Differentiating Malignant from Benign Pigmented or Non-Pigmented Skin Tumours—A Pilot Study on 3D Hyperspectral Imaging of Complex Skin Surfaces and Convolutional Neural Networks

Vivian Lindholm ^{1,*},[†], Anna-Maria Raita-Hakola ^{2,*},[†], Leevi Annala ², Mari Salmivuori ¹, Leila Jeskanen ¹, Heikki Saari ³, Sari Koskenmies ¹, Sari Pitkänen ¹, Ilkka Pölönen ², Kirsi Isoherranen ^{1,†} and Annamari Ranki ^{1,†}

¹ Department of Dermatology and Allergology, University of Helsinki and Helsinki University Hospital, 00290 Helsinki, Finland; mari.salmivuori@hus.fi (M.S.); leila.jeskanen@hus.fi (L.J.); sari.koskenmies@hus.fi (S.K.); sari.pitkanen@hus.fi (S.P.); kirsi.isoherrenan@hus.fi (K.I.); annamari.ranki@hus.fi (A.R.)

² Faculty of Information Technology, University of Jyväskylä, 40100 Jyväskylä, Finland; leevi.a.annala@jyu.fi (L.A.); ilkka.polonen@jyu.fi (I.P.)

³ VTT Technical Research Centre of Finland, 02150 Espoo, Finland; heikki.saari@vtt.fi

* Correspondence: vivian.lindholm@helsinki.fi (V.L.); anna.m.hakola@jyu.fi (A.-M.R.-H.); Tel.: +358-9471-86355 (V.L.)

† These authors contributed equally to this work.



Citation: Lindholm, V.; Raita-Hakola, A.-M.; Annala, L.; Salmivuori, M.; Jeskanen, L.; Saari, H.; Koskenmies, S.; Pitkänen, S.; Pölönen, I.; Isoherranen, K.; et al. Differentiating Malignant from Benign Pigmented or Non-Pigmented Skin Tumours—A Pilot Study on 3D Hyperspectral Imaging of Complex Skin Surfaces and Convolutional Neural Networks. *J. Clin. Med.* **2022**, *11*, 1914. <https://doi.org/10.3390/jcm11071914>

Academic Editors: Marco Manfredini, Silvana Ciardo and Ardigo Marco

Received: 5 March 2022

Accepted: 28 March 2022

Published: 30 March 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Abstract: Several optical imaging techniques have been developed to ease the burden of skin cancer disease on our health care system. Hyperspectral images can be used to identify biological tissues by their diffuse reflected spectra. In this second part of a three-phase pilot study, we used a novel hand-held SICSRIFIS Spectral Imager with an adaptable field of view and target-wise selectable wavelength channels to provide detailed spectral and spatial data for lesions on complex surfaces. The hyperspectral images (33 wavelengths, 477–891 nm) provided photometric data through individually controlled illumination modules, enabling convolutional networks to utilise spectral, spatial, and skin-surface models for the analyses. In total, 42 lesions were studied: 7 melanomas, 13 pigmented and 7 intradermal nevi, 10 basal cell carcinomas, and 5 squamous cell carcinomas. All lesions were excised for histological analyses. A pixel-wise analysis provided map-like images and classified pigmented lesions with a sensitivity of 87% and a specificity of 93%, and 79% and 91%, respectively, for non-pigmented lesions. A majority voting analysis, which provided the most probable lesion diagnosis, diagnosed 41 of 42 lesions correctly. This pilot study indicates that our non-invasive hyperspectral imaging system, which involves shape and depth data analysed by convolutional neural networks, is feasible for differentiating between malignant and benign pigmented and non-pigmented skin tumours, even on complex skin surfaces.

Keywords: biomedical optical imaging; convolutional neural networks; hyperspectral imaging; non-invasive imaging; optical modelling; photometric stereo; skin cancer; skin imaging

1. Introduction

Skin cancers comprise one third of all cancer diagnoses worldwide, and their incidence is continuously increasing [1]. This increase is producing a substantial burden on our health care system, introducing a need for fast and precise diagnostic tools. Dermoscopy, reflectance confocal microscopy (RCM), optical coherence tomography (OCT), and multispectral (MS) imaging are examples of non-invasive imaging tools that have been studied. Among these tools, dermoscopy is the most widely used and accepted, with good evidence for improving diagnostic accuracy [2]. Hyperspectral (HS) imaging is a promising non-invasive imaging option for biomedical and dermatological applications.

In contrast to MS imaging, the higher number and width of the spectral channels in HS imaging enables a higher spectral resolution, and thus, recognizes more subtle spectral differences [3]. Longer wavelengths penetrate deeper into the tissue revealing structures and biophysical phenomena, and HS can utilize up to near-infrared (NIR) wavelengths [4].

In dermatological applications, a hyperspectral imaging system (HSI) can identify biological tissues by their diffuse reflected spectra, which depend upon certain tissue chromophores such as melanin, haemoglobin, water, beta-carotene, collagen, and bilirubin [5]. Thus, the advantages of HS imaging offer the unique possibility of visualizing deeper structures (up to 3.5 mm) [4] with a large field of view and fast processing.

In the field of skin tumours, HSI has been used to successfully identify solar field carcinization [6]. HSI can also distinguish lentigo maligna from lentigo maligna melanoma [7], and pigmented basal cell carcinomas (BCCs) from malignant melanomas (MMs) [8], as well as identifying the borders of lentigo maligna [9] and BCCs [10]. HSI has been used successfully in melanoma screening [11–14] and the diagnosis of skin lesions [15]. However, there are few HSI studies on non-pigmented lesions [6,10], and in several studies, the diagnosis of supposedly benign lesions was not histologically confirmed, possibly biasing the results. Based on previous HSI studies, one of the major challenges is the complex surface topography and tomography of certain areas of the body, such as the shoulder, nose, ear, and other parts of the face.

HS imaging has advances and challenges. An HS image consists of a stack of frames. Each frame represents the intensity of a different wavelength of light, and each pixel (px) has a spectrum. The spectral data have both spectral and spatial domains, which provide detailed information on the target. Another major challenge for HS image processing is the large amount of data. An imager can capture tens or hundreds of frames, which can lead to the Hughes phenomenon, whereby the accuracy gradually increases as the number of dimensions increases, but decreases after a certain number of dimensions is reached [16], as well as producing redundancy among the samples [17]. These challenges can be avoided computationally, for instance, by employing common feature extraction methods. In this study, our solution was to customise the imager to capture only the necessary wavelengths. By selecting the wavelength channels and corresponding light-emitting diodes (LEDs) to represent the spectral absorption peaks of tissue chromophores from visible (VIS) to NIR light and using common reflectance calculations, the HS image contained the main diffuse tissue reflectance spectrums, thus providing a multidimensional view of a lesion with depth information. In this way, we employed a computationally effective solution since the amount of captured data and pre-processing could be limited.

This study represents the second stage in our three-stage pilot project to introduce and clinically test a new concept for skin cancer diagnosis. The core of the study is the SICSURFIS Spectral Imager, which is a compact, hand-held, piezo-actuated metallic mirror Fabry–Pérot interferometer (FPI)-based hyperspectral imager suitable for complex skin surfaces, as described in detail elsewhere [18]. The imager has integrated LED-based illumination, which is set to special lightning angles for photometric stereo imaging. Each LED and FPI position are individually controllable. Since the FPI is the spectral separator, the captured wavelengths can be selected by the software to match the LED illumination. The imager's field of view is adaptable based on the different sizes of specially designed light protection cones, which block unwanted light and, through their soft silicon collars, adapt to complex skin surfaces more easily than the imagers used in previous studies. The system's photometric stereo imaging provides skin-surface models, which, combined with the spectral and spatial domains, introduce more information on the analysed complex surfaces to the machine learning model. Another advantage of the SICSURFIS Spectral Imager is its feasibility: the hand-held device is small and light and can be used by one person without assistance (Figure 1).



Figure 1. The SICSURFIS imager, light-emitting diode (LED) module, and stray light protection cones (A). The imaging setup and software (B). Image source [18].

In this study, we captured HS images of pigmented and non-pigmented malignant and clinically reminiscent benign skin lesions using the novel SICSURFIS Spectral Imager. After imaging, the lesions were excised for histological analyses. The data were pre-processed with photometric stereo imaging methods (see the detailed explanation in Hakola et al. [18]), which provided skin-surface models for each of the captured wavebands. Besides the skin-surface models, the raw data were calculated from albedo images.

As a classification and delineation method, we used a specially designed convolutional neural network (CNN) that utilises spectral and spatial information (albedos) with the skin-surface model. Previous studies have estimated CNNs for diagnosing skin tumours based on plain red-green-blue (RGB) or dermoscopy pictures as accurately as an experienced dermatologist—at times even outperforming the human experts [19–23]. We calculated both the pixel-wise and majority voting classification results.

The aim of this pilot study was to compare the discriminatory capacity of combined HSI and CNN analysis for pigmented and non-pigmented, malignant, and benign skin tumours with similar appearances. As this combined analysis includes more data on the shape and depth of the lesion than previous studies with HSI [18], our hypothesis is that high resolution photometric stereo imaging and specific depth data for each wavelength will improve the classification and delineation results compared to previous studies, and that complex skin surfaces will no longer be a major challenge.

2. Materials and Methods

2.1. The Clinical Study

The clinical study was performed in February and March, 2020, at the Dermatology Outpatient Clinic, Helsinki University Hospital (HUS). The volunteering patients were recruited in an unselected fashion from patients attending the dermatology outpatient clinic due to diagnosed or suspected skin cancer, or from patients that wished to have their likely benign nevus removed. All patients provided their written informed consent. The study protocol followed the Declaration of Helsinki and was approved by the Ethical Review Board of HUS.

The physicians (V.L., K.I.) and the research nurse (J.Y.) captured images of the lesions *in vivo* with the HSI, as well as by digital and dermoscopy imaging, and subsequently biopsied or removed all lesions for the histological analyses. The histological analysis of each sample was performed by an experienced dermatopathologist (L.J.) at the dermatopathology laboratory of the Skin and Allergy Hospital, HUS. In total, 42 patients with 54 lesions were recruited. Of these, 6 lesions were excluded due to imaging artefacts (e.g., imaged with a stray light protection cone that was too small) and 6 lesions were excluded due to other histology (lentigo, seborrheic keratosis, neurofibroma, pyogenic granuloma, or carcinoma *in situ*). Thus, the final analyses included 42 lesions from 33 patients. As

pigmented lesions, we studied 7 MMs and 13 benign pigmented nevi (PN); and as non-pigmented lesions, we studied 10 BCCs, 5 squamous cell carcinomas (SCCs), and 7 benign intradermal (ID) nevi. Pigmented nevi included both combined and junctional nevi. ID nevi were clinically non-pigmented.

2.2. Patient Demographics and Lesion Characteristics

Patient demographics and lesion characteristics are listed in Table 1. Seventeen patients had a history of one or several skin cancers, and six had a history of one or several other cancers. Four patients were on immunosuppressive medication, among whom one was an organ transplant patient. Thirteen (31%) of the lesions were located on the head or neck, including three lesions on particularly complex sites (the ear, eyelid, and corner of the eye). The Breslow thicknesses of the SCCs varied between 1.1 and 5.2 mm, and that of the melanomas varied between 0.2 and 1.6 mm. All melanomas were of the superficial spreading type. The melanoma with a Breslow thickness 1.6 mm had a diameter of 28 mm and surrounding satellite lesions.

Table 1. Patient demographics and lesion characteristics (*n*, %).

PATIENTS	33	LESIONS	42
Mean age	68	Mean diameter	10.3 mm (2–30 mm)
Males	16 (48%)	Diagnosis	
Females	17 (52%)	Pigmented lesions:	
Fitzpatrick skin type		MM	7 (17%)
I	9 (27%)	Superficial spreading MM	7 (17%)
II	12 (36%)	PN	13 (31%)
III	12 (36%)	Junctional nevi	5 (12%)
History of skin cancer	17 (52%)	Compound nevi	8 (19%)
BCC	8 (24%)	High-grade dysplastic PN	2 (5%)
MM	7 (21%)	Low-grade dysplastic PN	3 (7%)
SCC	2 (6%)	Nevus recurrence	1 (2%)
BCC + MM	3 (9%)	Non-pigmented lesions:	
History of other cancers	6 (18%)	BCC	10 (24%)
Breast	3 (9%)	Nodular	6 (14%)
GI	2 (6%)	Nodular + superficial	3 (7%)
Prostate	2 (6%)	Superficial	1 (2%)
Blood	1 (3%)	SCC	5 (12%)
Immunosuppression	4 (12%)	ID	7 (17%)
Radiation therapy	3 (9%)	Location:	
Multiple nevus syndrome	2 (6%)	Head/neck	13 (31%)
Dysplastic nevi	6 (18%)	Torso	21 (50%)
Family history of skin cancer	4 (12%)	Upper extremities	2 (5%)
Multiple nevus syndrome in the family	4 (12%)	Lower extremities	6 (14%)

BCC = basal cell carcinoma, GI = gastrointestinal, ID = intradermal nevus, MM = malignant melanoma, PN = pigment nevus, SCC = squamous cell carcinoma.

2.3. The SICSRFIS Hyperspectral Imager

The SICSRFIS Spectral Imager (VTT Technical Research Centre of Finland, Espoo, Finland) prototype used in this study is composed of a hand-held Piezo-actuated metallic mirror FPI hyperspectral imager, an RGB sensor, and an LED light source. The hand-held

device weighs only 890 g. The light source has three series of intentionally selected nine LEDs, emitting light from white to 940 nm. The LEDs can be tilted at specific angles to enable photometric stereo imaging. Each LED and FPI position are individually controllable. Since the FPI is the spectral separator, the captured wavelengths can be selected to match the LED illumination by the software. To enable the imaging of complex skin surfaces, the imager has four stray light protection cones (Figure 1) that block unwanted light and adjust the imager on the skin. The stray light protection cones are different sizes ($3.1\text{--}26.4\text{ cm}^2$), and thus, offer the imager an adaptable field of view for different-sized lesions [18]. The soft silicon collars adapt to the shape of the skin and provide comfort. Detailed block diagrams and details of the imager and its calibration processes can be found in [18].

The imager is connected to a computer via Cube View software [24] (University of Jyväskylä, Spectral Imaging Laboratory [25]). The software is pre-set to capture 33 wavebands with wavelengths from visible to near infrared light (477–891 nm) [18]. The user interface is designed for an effortless workflow, allowing the user to concentrate on the patient. With one click, the imager captures three sets of raw frames (in total, six HS images) of one lesion with different light directions, providing spectral data for the three-dimensional skin-surface models and the albedos. The imager has a pixel resolution of $1\text{ px} \approx 24 \times 24\text{ }\mu\text{m}$. The tissue penetration depth depends upon the wavelength; for the wavelengths used in this work, the penetration depth was 0–6 mm [26]. Ease of use was considered when developing the software, which guides the user to capture dark and white references, followed by images, using three different LED light and waveband combinations with one click.

2.4. Data Pre-Processing

The three sets of raw frames, each including 33 wavelengths, were pre-processed twice. The two stages were raw image pre-processing and machine learning pre-processing. The detailed information and mathematical formulas for this process can be found in [18].

2.5. Raw Data Pre-Processing

The SICSRFIS imager is a high-quality *wavelength scan* HS imager with a collection speed similar to that of a *snapshot scan imager* [3], which captures the wavelengths frame by frame. The raw image pre-processing was conducted as shown in Figure 2. We obtained six raw HS images for each lesion. The VIS and visible and near-infrared (VNIR) channels were captured under three different angles of light. These images were calculated to determine their radiances and white references. The radiances and white references provided six reflectance HS images. The different light angles were combined into three HS images. The albedos and normals were calculated based on these images, forming one HS cube per lesion. The depth maps (the skin-surface model) were calculated from these cubes, and, after smoothening, one HS cube per lesion was formed. Each cube contained the albedo images and skin-surface models for each of the 33 captured wavelengths.

Since an HS image (HS cube) is a stack of frames, and the size of one frame was $1605 \times 1640\text{ px}$, the size of the whole processed HS cube was $1605 \times 1640 \times 66\text{ px}$ after the raw image pre-processing phase. Here, the first 33 of 66 pixels represented the skin-surface albedo images and the last 33 pixels were the respective skin-surface models. An “RGB image” reconstruction from the albedo images consisting of three wavelength channels is illustrated in Figure 3 on the left. On the right is a reconstruction of the skin-surface model for the same ID.

2.6. Machine Learning Pre-Processing

A physician (V.L.) manually annotated the RGB images (i.e., marked the areas of tumour and healthy skin based on a histological diagnosis). The annotations were digitised to binary form and formed the ground truth (Figure 4). We then created healthy skin masks, which were necessary for selecting the healthy skin pixels around the lesion. Our first SICSRFIS study highlighted that the skin-surface models had the best quality in

the middle of the images. We also noted some uncertainly labelled pixels in the lesion border areas, which may have influenced the results [18]. To overcome this, we created an algorithm that provided an annular boundary around the lesions to leave out low-quality areas at the periphery of the images. A 60-px-wide margin at the lesion border was also left out of the analyses to reduce bias from the annotations.

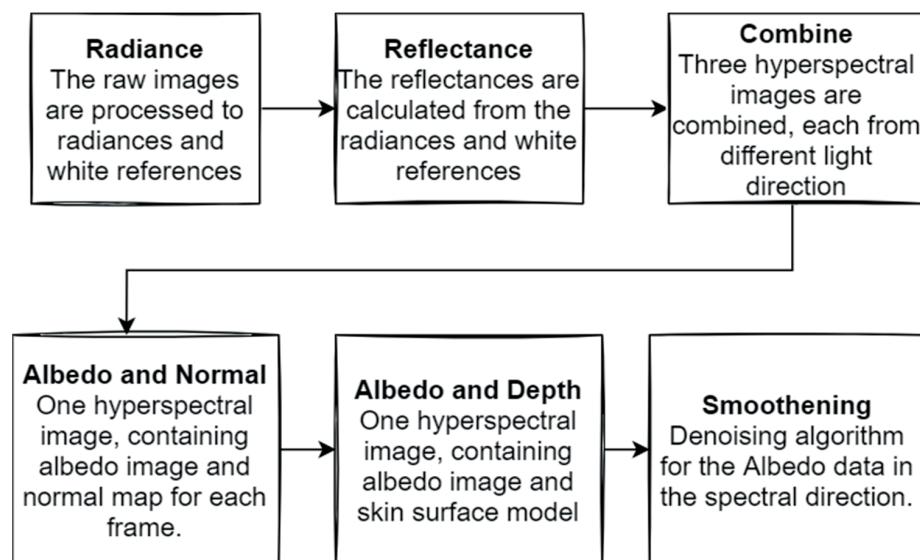


Figure 2. Raw-image pre-processing. The detailed process and mathematical formulas can be found in [18].

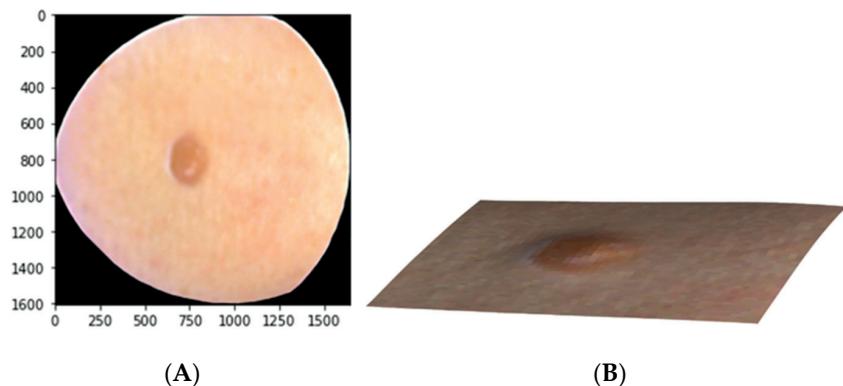


Figure 3. Example of the data after the raw image pre-processing. Reconstruction of an ID nevus using red-green-blue (RGB) from the albedo images (A) and its skin-surface model (B).

After finishing the masks, we selected the most significant skin-surface frame, which was the 575 nm wavelength. Each HS image consisted of 33 albedo images and the 575 nm skin-surface frame. Selection was based on our first pilot study [18]. The data were normalised between 0 and 1, and the possible infinity and NaN values were set to 0.

Figure 4 presents the training and test pixel selection, and the masks. Before windowing and selecting the training and test pixels, the images were vertically sliced through the middle of the lesions. The left side of each HS image was taken as the training image and the right side as the test image.

For each training and test image, 250 lesion and 100 healthy skin pixels were selected randomly. These pixels were the middle pixels of 30×30 pixel windows with 34 channels, which were selected using a rolling-window algorithm [18]. After data selection, the training set's windowed pixels were balanced using the imbalanced learn library random over-sampling method [27] and augmented with vertical and horizontal

tal flipping. The training set for classification of the pigmented lesions (MM, PN, and healthy skin) included 29,136 px, the validation set included 1400 px, and the test set included 7000 windowed px. The non-pigmented (BCC, SCC, ID, and healthy skin) training set encompassed 31,168 px, the validation set totalled 1540 px, and the test set totalled 7700 px. The machine learning models were tested via pixel-wise classification with whole HS images ($1605 \times 1640 \times 34$ px), which were pre-processed similarly with the test and training data. The windowed pixels from whole HS images were used for predicting the classification and confidence maps for each lesion image and for the majority voting analysis. Detailed information on the applied pre-processing algorithms has been described elsewhere [18].

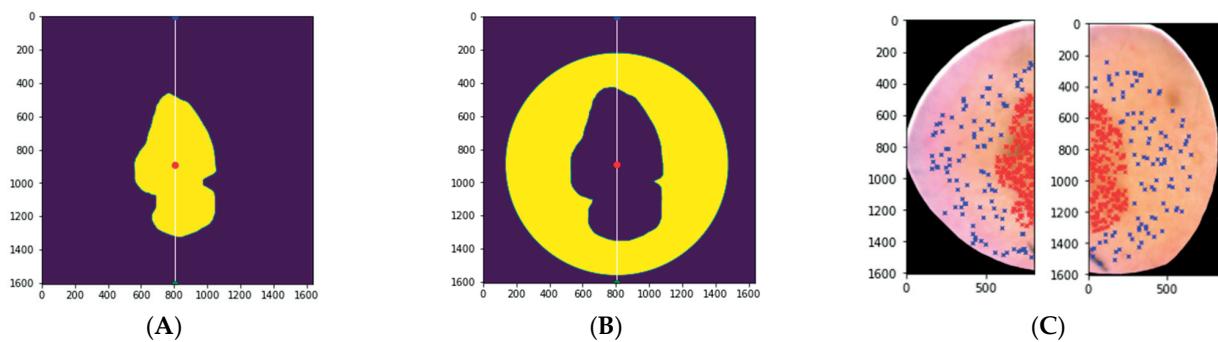


Figure 4. The ground truth with the image sliced through the middle (white line); training data on the left and test data on the right (A). The healthy skin mask (B). The training and test data. Blue represents the healthy skin pixels; red represents the lesion pixels (C). The size of the original lesion annotation was minimized to a lesion binary map of 30 pixels and similarly enlarged to a healthy skin mask. A 60-pixel margin at the lesion border was applied where no pixels were selected.

2.7. Data Analysis

The pre-processed training and validation data were analysed at the University of Jyväskylä using CNN models. We used two models, a three-class classifier (MM, PN, and healthy skin) for the pigmented lesions and a four-class classifier (BCC, SCC, ID, and healthy skin) for the non-pigmented lesions. Both models had similar structures, wherein the 3D and 2D layers were used to utilize the spectral, spatial, and depth information from the data. The CNN 3D layer extracted features from the albedo images, and the 2D layer extracted features from the skin-surface models. The CNN construction included the LeakyReLU activation function and max-pooling layers as shown in Figure 5.

After feature extraction, the 2D and 3D layer results were flattened, concatenated, and used as input for the hidden layers. Depending on the examination, the final output dense layer provided the results for three- or four-class classification. The model was trained via the Adam optimizer with default parameters using the categorical cross-entry loss function.

With both test setups (i.e., the three- and four-class classifiers), the results were obtained both pixel-wise and for the whole HS image. The pixel-wise results were conducted from randomly selected windowed test data containing the lesion types and healthy skin samples. The whole windowed HS images were classified to produce the classification confident and classification maps for visual evaluation. Since the whole HS images also contained pixels outside the imager's field of view (the black area seen in image C, Figure 4), the overall accuracy results were biased by these outlier pixels. Therefore, we performed a majority voting test for the whole image classification results. With the original lesion binary maps delineated by a dermatologist and confirmed via histology, we selected all pixels that were annotated as lesion pixels from the right side of the image (sliced vertically as in the training and test data, Figure 4). Then, we calculated the number of lesion pixels for each lesion type. By assuming that there was only one type of lesion pixel per lesion, we re-classified the images to the majority class. That way, majority voting provided one

primary diagnosis for each lesion based on the most common diagnosis for the lesion pixels in the test data.

We provided accuracy reports for both the pixel-wise analysis and majority voting. The analysis included sensitivity, specificity, positive predictive values (PPVs), and F1-score measurements, which were provided by Scikit-Learn metrics [28]. The results were completed via confusion matrices.

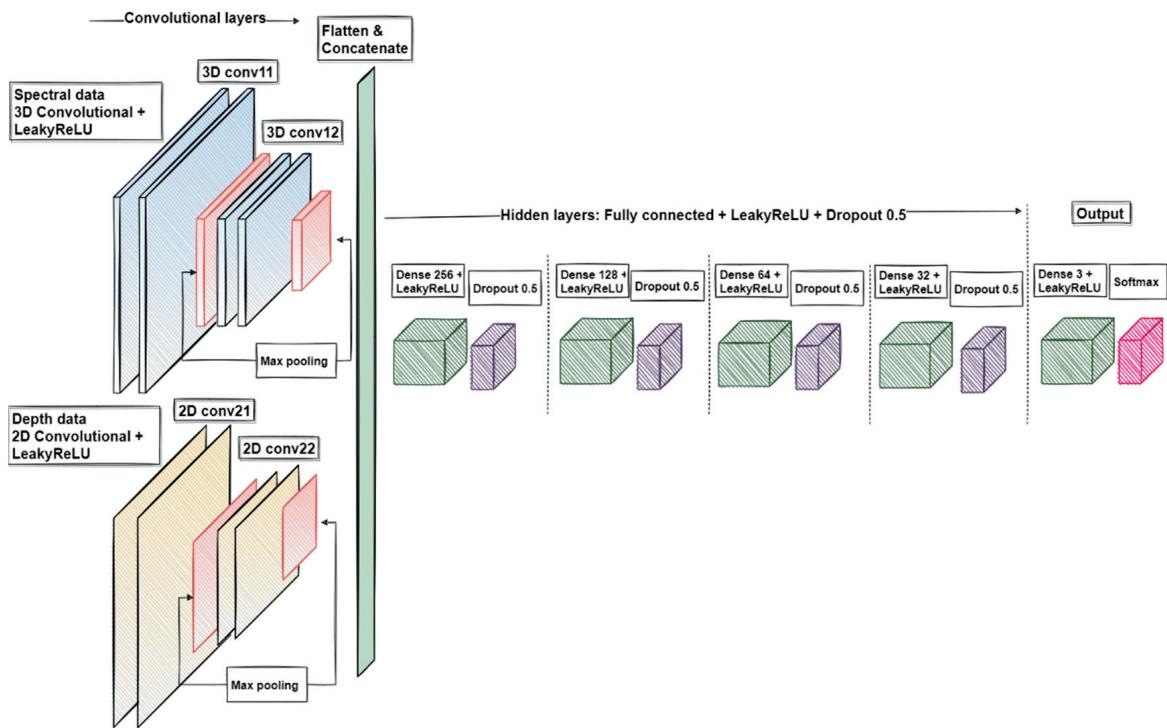


Figure 5. A visualization of the convolutional neural network. The 3D convolutional layers were used with the albedo images, and the 2D convolutional layers processed the skin-surface model. The outputs were concatenated, flattened, and used as input for the hidden layers. Depending on the examination, the output layer was 3- or 4-class classification (dense layer). Image source [18].

The study was implemented with Scikit-Learn [28], Scikit-Image [27], SciPy [29], and Tensorflow Python libraries [30]. Computing was performed using a Linux GPU server, 1 × Tesla P100, x86_64. The HSI and computational analyses are described in detail in a previous article by Raita-Hakola et al. [18].

3. Results

3.1. Classification Results for Pigmented Lesions

In total, 20 pigmented lesions, 7 MMs, and 13 benign PNs were studied. The pixel-wise classification for pigmented lesions reached a weighted sensitivity of 87%, a specificity of 93%, and a PPV of 87%. In majority voting, the weighted sensitivity was 95%, specificity was 97%, and PPV was 96%. None of the melanomas were classified as nevus, but one low-grade dysplastic benign compound nevus was classified as melanoma by the majority voting test. Figure 6 presents the PPVs for the separate diagnoses by each analysis method. A representative example of a pixel-wise classification map for a pigmented lesion is illustrated in Figure 7.

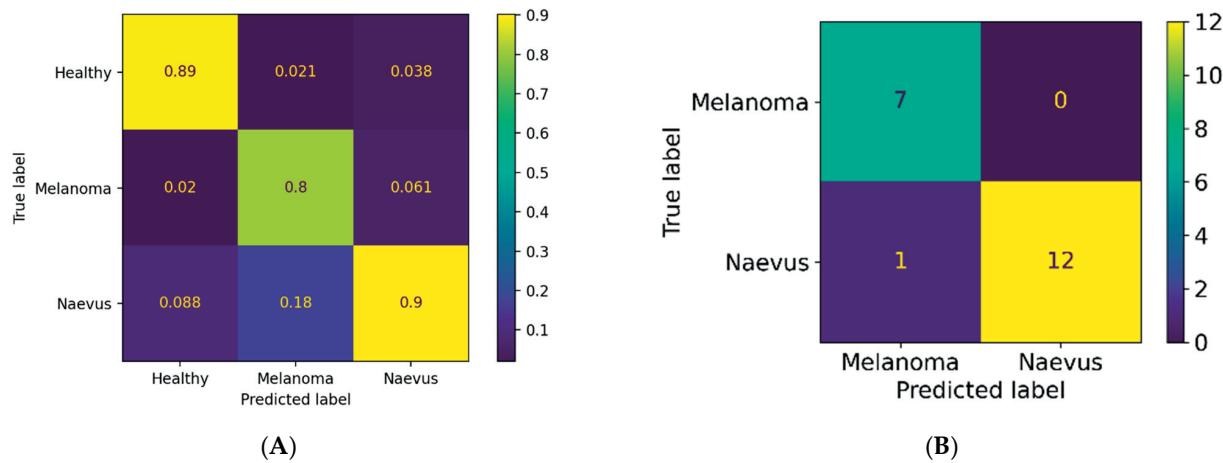


Figure 6. Confusion matrices presenting positive prediction values (PPVs) for pigmented lesions in pixel-wise (A) and majority voting analyses (B). Melanoma = malignant melanoma, naevus = pigmented nevus.

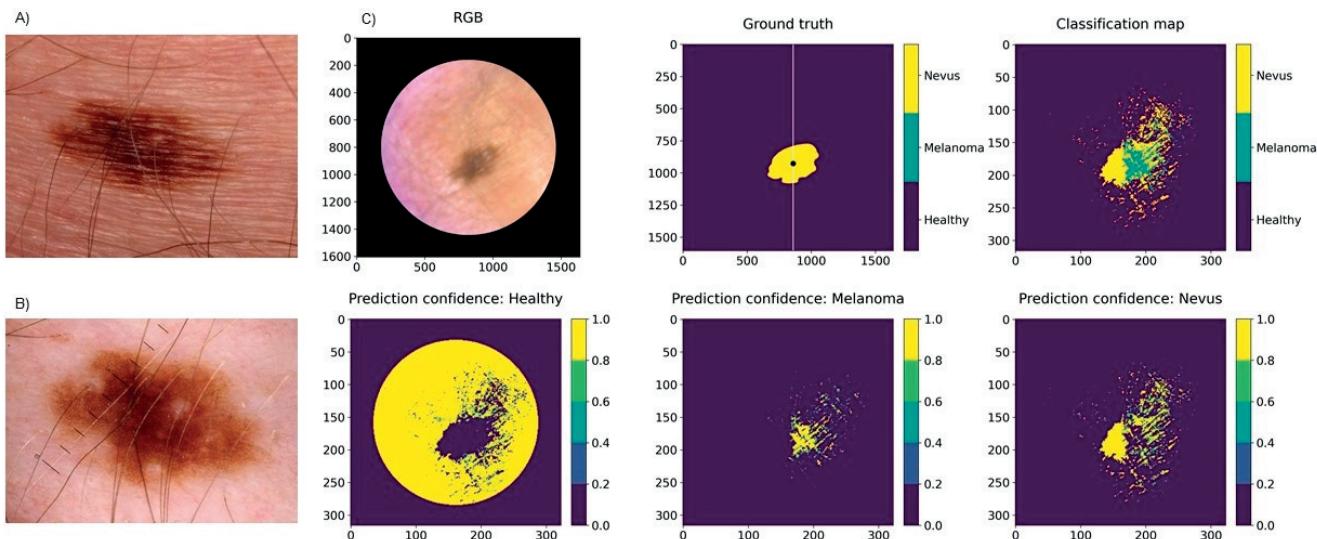


Figure 7. Clinical (A) and dermoscopy images (B) and a classification map (C) of a 10 mm low-grade dysplastic compound nevus of the chest, clinically and histologically diagnosed as a benign nevus but classified as a melanoma by the SICSRIFIS system. The test data (right half of the image) included pixels classified as both melanoma and nevus. However, according to the majority voting analysis, there were more melanoma pixels. Light reflection caused probable artefacts in the surrounding area of the lesion.

3.2. Classification Results for Non-Pigmented Lesions

The pixel-wise classification for non-pigmented lesions (22 lesions: 10 BCCs, 5 SCCs, and 7 benign IDs) reached a weighted sensitivity of 79%, a specificity of 91%, and a PPV of 80%. In the majority voting analysis, the weighted sensitivity was 100%, specificity was 100%, and PPV was 100%. Thus, all lesions were classified correctly by majority voting. Figure 8 presents the PPVs for the separate diagnoses by each analysis method, and Figure 9 illustrates a pixel-wise classification map of a non-pigmented lesion.

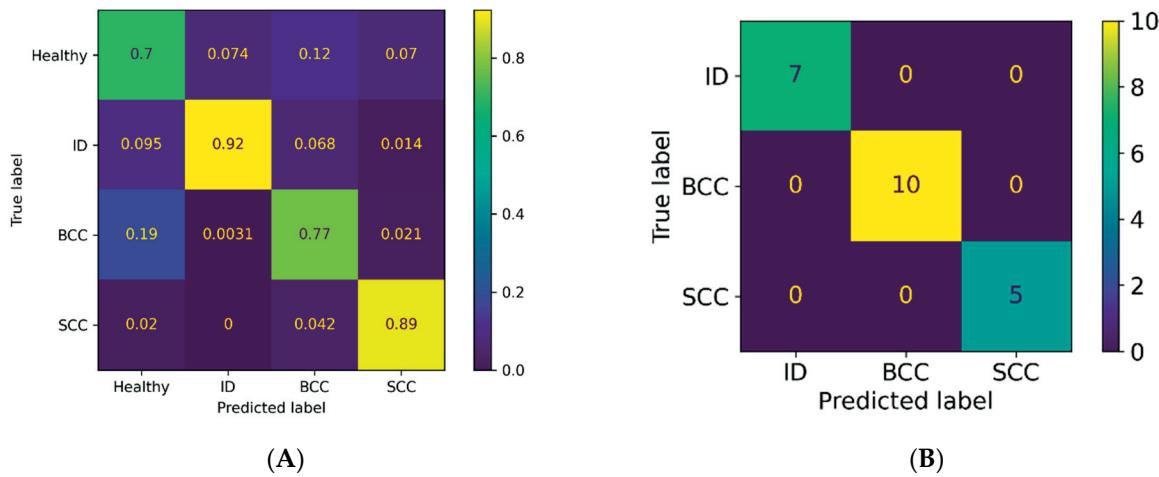


Figure 8. Confusion matrices presenting the PPVs for non-pigmented lesions in the (A) pixel-wise and (B) majority voting analyses.

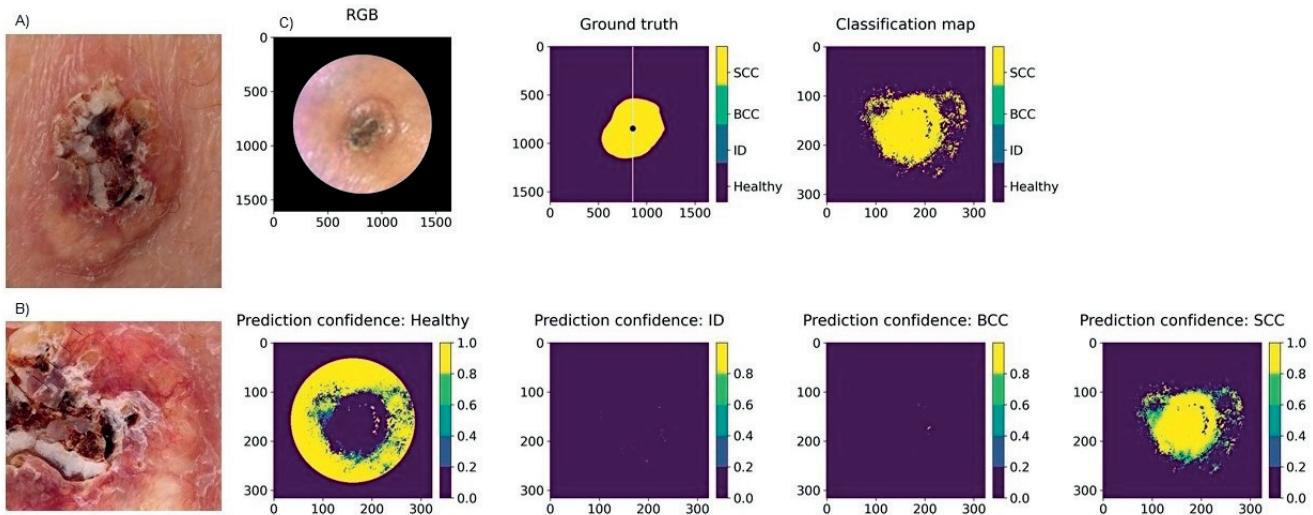


Figure 9. Clinical (A) and dermoscopy image (B) and the classification map (C) of a 15 mm SCC on the leg. Clinically, this lesion could be either a BCC or a SCC, but it was correctly classified as a SCC by the system. The SICSURFIS system delineated the lesion accurately, although it was surrounded by some probable imaging artefacts caused by the uneven skin colour of the healthy skin.

4. Discussion

We achieved a sensitivity of 87% and a specificity of 93% for recognising melanoma from pigmented nevi and healthy skin with a pixel-wise analysis, and even higher results using majority voting (95% and 97%, respectively). Two previous studies on melanoma recognition combined HSI with deep learning. Hirano et al. studied 619 lesions and classified them utilizing GoogLeNet pretrained with ImageNet [12]. However, the results were unsatisfactory, achieving a sensitivity of 72% and a specificity of 81% after data augmentation, likely due to the reduction of the wavelengths used from 84 to 3. Three wavelengths are considered more comparable to digital or multispectral, rather than hyperspectral, imaging. Kato et al. similarly utilized a pretrained GoogLeNet to analyse 619 lesions [13]. The authors used automated analyses with transfer learning and reached somewhat higher classification results based on two analysis methods with sensitivities of 80% and 77% and specificities of 82% and 82% after data augmentation. Our results are

more accurate but remain in line with those of Kato et al. Thus, it seems that the results obtained via HS imaging are repeatable.

In previous studies on the recognition of pigmented lesions with HSI that did not utilize machine learning, Christensen et al. studied 202 skin lesions and reached a sensitivity of 97% for malignant lesions and a specificity of 42% for benign lesions [11]. Three other well-powered studies reached sensitivities of 88–97% and specificities of 87–100% [14,31,32]. However, none of these previous studies histologically confirmed the diagnosis of supposedly benign lesions [11–14,31,32], and thus, the results are not directly comparable with those obtained for histologically verified lesions. In the present study, all lesions were confirmed histologically, and only healthy skin was assessed by clinical inspection alone.

CNN analyses of digital or dermoscope images have achieved similar, or even greater, accuracy for melanoma recognition than that achieved by dermatologists [19–21,23]. For instance, in a study by Haenssle et al. on dermoscopy images, 58 dermatologists reached a sensitivity of 87% and a specificity of 71%, whereas the CNN achieved a higher specificity of 83% [20]. In a meta-analysis of 22 melanoma recognition-studies, dermoscopy-CNN achieved a sensitivity of 90% and a specificity of 74% [21]. For MS imaging with computer-aided diagnosis, the values for sensitivity and specificity based on 15 studies were 93% and 44%, respectively, demonstrating that MS imaging is often limited by low specificity values [33]. In a recent meta-analysis of melanoma detection via RCM, the pooled sensitivity and specificity were 92% and 70%, respectively [34]. Two studies on OCT reached sensitivities of 74–89% and specificities of 61–92% [35,36]. Thus, the results of our HS analyses are strong and offer high specificity values compared to other analysis methods. In the studies by Pezzini et al. and Gamblicher et al. on RCM and OCT [34,36], trained experts were required for the analyses.

For the detection of non-pigmented or non-melanoma skin cancer (NMSC) with HSI, studies are rare. Our results seem promising with a sensitivity of 79% and a specificity of 91% for separating BCC, SCC, ID, and healthy skin, considering that non-pigmented lesions are more difficult than pigmented lesions to diagnose via dermoscopy. For the majority voting analysis, all non-pigmented lesions were classified correctly (sensitivity and specificity of 100%). A meta-analysis of the dermoscopy of BCC reached a sensitivity of 93% with a fixed specificity of 80%, but no conclusions could be drawn on the dermoscopy of SCCs [37]. CNN has been compared with medical personnel for recognizing malignant pigmented and non-pigmented lesions from dermoscopic and digital images, and it achieved a higher sensitivity of 81% compared to human raters (78%) [22]. Our HSI-CNN system achieved a higher accuracy compared to the aforementioned study by Tschandl et al. [22]. HSI images include wavelengths of NIR in addition to the VIS light used solely by digital and dermoscopy imaging; thus, some tissue chromophore-specific information that is not visible to the human eye may be obtained by HS imaging.

For non-pigmented skin lesions, a meta-analysis of RCM including four studies on BCC reached a pooled sensitivity of 76% and a specificity of 95% [38]. Another meta-analysis of three studies utilizing OCT reached a sensitivity of 95% and a specificity of 77% [39]. Notably, in these studies, machine learning was not used for interpretation, and a high-level of expertise is required for the use of RCM and OCT. Studies on MS imaging of NMSC are rare [40,41], making it impossible to draw conclusions on this technique's feasibility.

The PPV can be used to describe the performance of a diagnostic imaging tool. The pixel-wise analyses of pigmented lesions had a higher PPV for nevus (90%) than for melanoma (80%) (Figure 6). Thus, most of the pixels classified as nevus were true nevus pixels (90%), but a high number of pixels classified as melanoma were true nevus pixels (18%). Additionally, many healthy skin or nevus pixels were classified as melanoma—in total, 20.1% (false positives). Only 8.1% of true melanoma pixels were misclassified (false negatives), which is acceptable for a diagnostic tool that is used to recognize a malignant condition.

For the single case where a 10 mm low-grade dysplastic combined nevus was classified falsely as a melanoma by majority voting (Figure 7), the lesion's more strongly pigmented

parts were classified as melanoma, while the less strongly pigmented parts were classified as nevus. In the pixel-wise map, about as many pixels were classified as melanoma as nevus. Dysplastic nevi are atypical both clinically and histologically and can thus have similar features to melanomas. Clinically speaking, it is better that majority voting classified a lesion as a melanoma, as it would be more serious to misdiagnose a melanoma as a nevus. As only a small portion of the lesion was evaluated by histology (we analysed eight cuts), and the whole lesion was analysed by HSI, it could be speculated that HSI may recognize areas of early melanoma growth that were overlooked by histology.

For the non-pigmented lesions, the PPV was higher for ID (92%) and SCC (89%) than BCC (77%), as a high number of true BCC pixels were classified as healthy skin (19%) (Figure 8). In our study, four of the ten BCCs belonged to the superficial or partly superficial subtypes. Tumours of this subtype can be ill-defined at the tumour border, and small tumour islands can be spread out among healthy skin at the lateral tumour border. As these islands are not detectable with the naked eye, parts of the actual BCC can be annotated falsely as healthy skin, impairing the results of the HSI-CNN. In contrast, nodular BCC, SCC, and ID usually have a clear and well-defined tumour border.

Because of the small number of samples, the CNN in this study was trained on one half of a lesion, and the other half of the same lesion was used for the analyses. This approach may have positively influenced the results, as the training and test data were similar. We noticed that colour changes in the healthy skin (reflections, uneven skin tone, hair, and nipples) were occasionally classified as lesions by the CNN and not as healthy skin. This result is understandable, as we did not have a class for features such as hair, whose colour and structure differ from healthy skin. Another limitation of this study when considering biology is that the CNN should have also been trained on actinic keratoses and *in situ* carcinomas, which are precursors of SCC and can surround SCCs on sun-damaged skin.

The SICSRFIS Imager can capture lesions of different sizes that are located on difficult sites, although the risk of imaging artefacts is higher for lesions on complex sites. The smallest protection cone is ideal when capturing small lesions on complex sites. With such a cone, the correct distance to the lens is more easily maintained, enabling the image to remain focused. However, when using the smallest cone, the amount of healthy skin surrounding the lesion was found to be insufficient. When using the larger cones for these locations, stray light was able to pass the protection cone and cause artefacts. Using a broader soft ring or cloth around the protective cone could decrease the risk for artefacts, and images of exclusively healthy skin would provide the additional data required for healthy skin. Some artefacts could have been caused by movement of the imager during imaging, as the process required up to 20 s. A faster imaging process could thus enhance the image quality.

The strengths of this study include the histological verification of all lesions and the multi-classification of lesions into different diagnosis categories, not just malignant or benign. The SICSRFIS imager has a high resolution and was developed for photometric stereo imaging. Moreover, the feasibility of both the imager and the software has improved. Unlike other studies [14], we included lesions from complex sites. Additionally, we used a CNN (not only indexed limit values) for the analyses and employed both spatial and spectral data. The CNN used in this pilot study could be further trained and applied in larger-scale studies.

Based on these results and previous studies on HSI, it seems that HSI could be used not only to aid in the diagnosis of pigmented and non-pigmented lesions but also to form a classification map of lesions that could be used to delineate lesions more accurately, and to indicate different diagnoses or parts with deeper invasion in a lesion, thus detecting the most informative biopsy site of a lesion [10,42]. Although even experienced dermatologists can mistake malignant skin lesions as benign, this diagnostic aid could be even more helpful for non-dermatologists. Contrary to many other imaging techniques, HSI does not require training or histological knowledge for the user [5]. Accurate delineation could, moreover, reduce the need for re-excisions of skin tumours. Reducing the number of biopsies and

re-excisions would save costs and thus be important for the treatment of the increasing number of skin cancer cases in the future.

In conclusion, the results of this pilot study with 42 pigmented and non-pigmented lesions indicate that the novel SICSURFIS HSI-CNN system with shape and depth data can reliably aid in the differentiation of malignant and benign skin lesions, even on complex skin surfaces. All lesions, except one, were diagnosed correctly by the majority voting analyses. However, the results must still be validated by larger-scale studies, which are anticipated to reveal the advantages of our 3D spectral imaging system.

Author Contributions: Conceptualization, I.P., H.S., S.P. and A.R.; methodology, I.P., A.R., K.I., S.P., A.-M.R.-H., L.A., V.L., S.K. and M.S.; software, A.-M.R.-H. and L.A.; validation, A.-M.R.-H., L.A., A.R. and K.I.; formal analysis, A.-M.R.-H. and L.A.; investigation, V.L., K.I., A.-M.R.-H., L.A. and L.J.; resources, I.P., H.S., L.J. and A.R.; data curation, A.-M.R.-H., L.A. and V.L.; writing—original draft preparation, V.L. and A.-M.R.-H.; writing—review and editing, A.R., K.I., I.P., S.P., S.K., M.S., L.A., H.S. and L.J.; visualization, A.-M.R.-H., L.A. and V.L.; supervision, I.P., H.S. and A.R.; project administration, I.P., H.S., K.I. and A.R.; funding acquisition, I.P., H.S. and A.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the RADDESS program of the Academy of Finland, consortium grant number 314519. Open access funding was provided by University of Helsinki.

Institutional Review Board Statement: The study was conducted in accordance with the Declaration of Helsinki and approved by the Institutional Review Board of Helsinki University Hospital (protocol code HUS/2089/2019, date of approval 30 July 2019).

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: The data presented in this study are not publicly available or available upon request due to ethical and privacy reasons.

Acknowledgments: We acknowledge the skilful assistance of Jenni Ylisaari, R.N.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. World Health Organisation. Radiation: Ultraviolet (UV) Radiation and Skin Cancer. Available online: <https://www.who.int/uv/faq/skincancer/en/index1.html> (accessed on 10 January 2022).
2. Yélamos, O.; Braun, R.P.; Liopyris, K.; Wolner, Z.J.; Kerl, K.; Gerami, P.; Marghoob, A.A. Usefulness of Dermoscopy to Improve the Clinical and Histopathologic Diagnosis of Skin Cancers. *J. Am. Acad. Dermatol.* **2019**, *80*, 365–377. [CrossRef] [PubMed]
3. Shapey, J.; Xie, Y.; Nabavi, E.; Bradford, R.; Saeed, S.R.; Ourselin, S.; Vercauteren, T. Intraoperative Multispectral and Hyperspectral Label-free Imaging: A Systematic Review of in Vivo Clinical Studies. *J. Biophotonics* **2019**, *12*, e201800455. [CrossRef] [PubMed]
4. Lu, G.; Fei, B. Medical Hyperspectral Imaging: A Review. *J. Biomed. Opt.* **2014**, *19*, 010901. [CrossRef] [PubMed]
5. Rey-Barroso, L.; Peña-Gutiérrez, S.; Yáñez, C.; Burgos-Fernández, F.J.; Vilaseca, M.; Royo, S. Optical Technologies for the Improvement of Skin Cancer Diagnosis: A Review. *Sensors* **2021**, *21*, 252. [CrossRef] [PubMed]
6. Neittaanmäki-Perttu, N.; Grönroos, M.; Tani, T.; Pöölönen, I.; Ranki, A.; Saksela, O.; Snellman, E. Detecting Field Cancerization Using a Hyperspectral Imaging System. *Lasers Surg. Med.* **2013**, *45*, 410–417. [CrossRef]
7. Neittaanmäki-Perttu, N.; Karppinen, T.; Tani, T.; Snellman, E.; Grönroos, M. Long-Term Outcome of Low-Concentration Hexyl-5-Aminolaevulinate Daylight Photodynamic Therapy for Treatment of Actinic Keratoses. *Acta Derm. Venereol.* **2017**, *97*, 120–121. [CrossRef]
8. Räsänen, J.; Salmivuori, M.; Pöölönen, I.; Grönroos, M.; Neittaanmäki, N. Hyperspectral Imaging Reveals Spectral Differences and Can Distinguish Malignant Melanoma from Pigmented Basal Cell Carcinomas: A Pilot Study. *Acta Derm. Venereol.* **2021**, *101*, adv00405. [CrossRef]
9. Neittaanmäki-Perttu, N.; Grönroos, M.; Jeskanen, L.; Pöölönen, I.; Ranki, A.; Saksela, O.; Snellman, E. Delineating Margins of Lentigo Maligna Using a Hyperspectral Imaging System. *Acta Derm. Venereol.* **2015**, *95*, 549–552. [CrossRef]
10. Salmivuori, M.; Neittaanmäki, N.; Pöölönen, I.; Jeskanen, L.; Snellman, E.; Grönroos, M. Hyperspectral Imaging System in the Delineation of Ill-defined Basal Cell Carcinomas: A Pilot Study. *J. Eur. Acad. Dermatol.* **2019**, *33*, 71–78. [CrossRef]
11. Christensen, G.B.; Nagaoka, T.; Kiyoohara, Y.; Johansson, I.; Ingvar, C.; Nakamura, A.; Sota, T.; Nielsen, K. Clinical Performance of a Novel Hyperspectral Imaging Device for Cutaneous Melanoma and Pigmented Skin Lesions in Caucasian Skin. *Skin Res. Technol.* **2021**, *27*, 803–809. [CrossRef]

12. Hirano, G.; Nemoto, M.; Kimura, Y.; Kiyohara, Y.; Koga, H.; Yamazaki, N.; Christensen, G.; Ingvar, C.; Nielsen, K.; Nakamura, A.; et al. Automatic Diagnosis of Melanoma Using Hyperspectral Data and GoogLeNet. *Skin Res. Technol.* **2020**, *26*, 891–897. [CrossRef] [PubMed]
13. Kato, K.; Nemoto, M.; Kimura, Y.; Kiyohara, Y.; Koga, H.; Yamazaki, N.; Christensen, G.; Ingvar, C.; Nielsen, K.; Nakamura, A.; et al. Performance Improvement of Automated Melanoma Diagnosis System by Data Augmentation. *ABE* **2020**, *9*, 62–70. [CrossRef]
14. Leon, R.; Martinez-Vega, B.; Fabelo, H.; Ortega, S.; Melian, V.; Castaño, I.; Carretero, G.; Almeida, P.; Garcia, A.; Quevedo, E.; et al. Non-Invasive Skin Cancer Diagnosis Using Hyperspectral Imaging for In-Situ Clinical Support. *J. Clin. Med.* **2020**, *9*, 1662. [CrossRef] [PubMed]
15. Pölönen, I.; Rahkonen, S.; Annala, L.; Neittaanmäki, N. Convolutional neural networks in cancer detection using spatial and spectral domain. *Proc. SPIE* **2019**, *2019*, 108510B. [CrossRef]
16. Abend, K.; Harley, T.; Chandrasekaran, B.; Hughes, G. Comments on “On the Mean Accuracy of Statistical Pattern Recognizers” by Hughes, G.F. *IEEE Trans. Inform. Theory* **1969**, *15*, 420–423. [CrossRef]
17. Ding, S.; Keal, C.A.; Zhao, L.; Yu, D. Dimensionality Reduction and Classification for Hyperspectral Image Based on Robust Supervised ISOMAP. *J. Int. Prod. Eng.* **2022**, *39*, 19–29. [CrossRef]
18. Raita-Hakola, A.M.; Annala, L.; Lindholm, V.; Trops, R.; Näsilä, A.; Saari, H.; Ranki, A.; Pölönen, I. FPI based Hyperspectral Imager for the Complex Surfaces—Calibration, Illumination and Applications. *Sensors* **2022**, submitted.
19. Esteva, A.; Kuprel, B.; Novoa, R.A.; Ko, J.; Swetter, S.M.; Blau, H.M.; Thrun, S. Dermatologist-Level Classification of Skin Cancer with Deep Neural Networks. *Nature* **2017**, *542*, 115–118. [CrossRef]
20. Haenssle, H.A.; Fink, C.; Schneiderbauer, R.; Toberer, F.; Buhl, T.; Blum, A.; Kalloo, A.; Hassen, A.B.H.; Thomas, L.; Enk, A.; et al. Man against Machine: Diagnostic Performance of a Deep Learning Convolutional Neural Network for Dermoscopic Melanoma Recognition in Comparison to 58 Dermatologists. *Ann. Oncol.* **2018**, *29*, 1836–1842. [CrossRef]
21. Ferrante di Ruffano, L.; Takwoingi, Y.; Dinnes, J.; Chuchu, N.; Bayliss, S.E.; Davenport, C.; Matin, R.N.; Godfrey, K.; O’Sullivan, C.; Gulati, A.; et al. Computer-Assisted Diagnosis Techniques (Dermoscopy and Spectroscopy-Based) for Diagnosing Skin Cancer in Adults. *Cochrane Database Syst. Rev.* **2018**, *2018*, CD013186. [CrossRef]
22. Tschanzl, P.; Rosendahl, C.; Akay, B.N.; Argenziano, G.; Blum, A.; Braun, R.P.; Cabo, H.; Gourhant, J.-Y.; Kreusch, J.; Lallas, A.; et al. Expert-Level Diagnosis of Nonpigmented Skin Cancer by Combined Convolutional Neural Networks. *JAMA Dermatol.* **2019**, *155*, 58–65. [CrossRef] [PubMed]
23. Tschanzl, P.; Codella, N.; Akay, B.N.; Argenziano, G.; Braun, R.P.; Cabo, H.; Gutman, D.; Halpern, A.; Helba, B.; Hofmann-Wellenhof, R.; et al. Comparison of the Accuracy of Human Readers versus Machine-Learning Algorithms for Pigmented Skin Lesion Classification: An Open, Web-Based, International, Diagnostic Study. *Lancet Oncol.* **2019**, *20*, 938–947. [CrossRef]
24. Trops, R.; Hakola, A.-M.; Jääskeläinen, S.; Näsilä, A.; Annala, L.; Eskelinen, M.A.; Saari, H.; Pölönen, I.; Rissanen, A. Miniature MOEMS Hyperspectral Imager with Versatile Analysis Tools. In Proceedings of the MOEMS and Miniaturized Systems XVIII, San Francisco, CA, USA, 2–7 February 2019; Piyawattanametha, W., Park, Y.-H., Zappe, H., Eds.; SPIE: Washington, DC, USA, 2019; p. 31.
25. University of Jyväskylä, Spectral Imaging Laboratory. Available online: <https://www.jyu.fi/it/en/research/our-laboratories/hsi> (accessed on 10 January 2022).
26. Barun, V.V.; Ivanov, A.P.; Volotovskaya, A.V.; Ulashchik, V.S. Absorption Spectra and Light Penetration Depth of Normal and Pathologically Altered Human Skin. *J. Appl. Spectrosc.* **2007**, *74*, 430–439. [CrossRef]
27. van der Walt, S.; Schönberger, J.L.; Nunez-Iglesias, J.; Boulogne, F.; Warner, J.D.; Yager, N.; Gouillart, E.; Yu, T. Scikit-Image: Image Processing in Python. *PeerJ* **2014**, *2*, e453. [CrossRef]
28. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-Learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
29. Virtanen, P.; Gommers, R.; Oliphant, T.E.; Haberland, M.; Reddy, T.; Cournapeau, D.; Burovski, E.; Peterson, P.; Weckesser, W.; Bright, J.; et al. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nat. Methods* **2020**, *17*, 261–272. [CrossRef]
30. Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G.S.; Davis, A.; Dean, J.; Devin, M.; et al. Tensor Flow Large-Scale Machine Learning on Heterogeneous Systems. Available online: <https://arxiv.org/abs/1603.04467> (accessed on 10 January 2022).
31. Nagaoka, T.; Kiyohara, Y.; Koga, H.; Nakamura, A.; Saida, T.; Sota, T. Modification of a Melanoma Discrimination Index Derived from Hyperspectral Data: A Clinical Trial Conducted in 2 Centers between March 2011 and December 2013. *Skin Res. Technol.* **2015**, *21*, 278–283. [CrossRef]
32. Pardo, A.; Gutiérrez-Gutiérrez, J.A.; Lihacova, I.; López-Higuera, J.M.; Conde, O.M. On the Spectral Signature of Melanoma: A Non-Parametric Classification Framework for Cancer Detection in Hyperspectral Imaging of Melanocytic Lesions. *Biomed. Opt. Express* **2018**, *9*, 6283. [CrossRef]
33. Heibel, H.D.; Hooey, L.; Cockerell, C.J. A Review of Noninvasive Techniques for Skin Cancer Detection in Dermatology. *Am. J. Clin. Dermatol.* **2020**, *21*, 513–524. [CrossRef]
34. Pezzini, C.; Kaleci, S.; Chester, J.; Farnetani, F.; Longo, C.; Pellacani, G. Reflectance Confocal Microscopy Diagnostic Accuracy for Malignant Melanoma in Different Clinical Settings: Systematic Review and Meta-analysis. *J. Eur. Acad. Dermatol. Venereol.* **2020**, *34*, 2268–2279. [CrossRef]

35. Wessels, R.; de Bruin, D.M.; Relyveld, G.N.; Faber, D.J.; Vincent, A.D.; Sanders, J.; van Leeuwen, T.G.; Ruers, T.J.M. Functional Optical Coherence Tomography of Pigmented Lesions. *J. Eur. Acad. Dermatol. Venereol.* **2015**, *29*, 738–744. [[CrossRef](#)] [[PubMed](#)]
36. Gambichler, T.; Schmid-Wendtner, M.H.; Plura, I.; Kampilafkos, P.; Stückler, M.; Berking, C.; Maier, T. A Multicentre Pilot Study Investigating High-Definition Optical Coherence Tomography in the Differentiation of Cutaneous Melanoma and Melanocytic Naevi. *J. Eur. Acad. Dermatol. Venereol.* **2015**, *29*, 537–541. [[CrossRef](#)] [[PubMed](#)]
37. Dinnes, J.; Deeks, J.J.; Chuchu, N.; Matin, R.N.; Wong, K.Y.; Aldridge, R.B.; Durack, A.; Gulati, A.; Chan, S.A.; Johnston, L.; et al. Visual Inspection and Dermoscopy, Alone or in Combination, for Diagnosing Keratinocyte Skin Cancers in Adults. *Cochrane Database Syst. Rev.* **2018**, *2018*, CD011901. [[CrossRef](#)] [[PubMed](#)]
38. Dinnes, J.; Deeks, J.J.; Chuchu, N.; Saleh, D.; Bayliss, S.E.; Takwoingi, Y.; Davenport, C.; Patel, L.; Matin, R.N.; O’Sullivan, C.; et al. Reflectance Confocal Microscopy for Diagnosing Keratinocyte Skin Cancers in Adults. *Cochrane Database Syst. Rev.* **2018**, *2018*, CD013191. [[CrossRef](#)]
39. Ferrante di Ruffano, L.; Dinnes, J.; Deeks, J.J.; Chuchu, N.; Bayliss, S.E.; Davenport, C.; Takwoingi, Y.; Godfrey, K.; O’Sullivan, C.; Matin, R.N.; et al. Optical Coherence Tomography for Diagnosing Skin Cancer in Adults. *Cochrane Database Syst. Rev.* **2018**, *2018*, CD013189. [[CrossRef](#)]
40. Terstappen, K.; Suurküla, M.; Hallberg, H.; Ericson, M.B.; Wennberg, A.-M. Poor Correlation between Spectrophotometric Intracutaneous Analysis and Histopathology in Melanoma and Nonmelanoma Lesions. *J. Biomed. Opt.* **2013**, *18*, 061223. [[CrossRef](#)]
41. Rajaram, N.; Reichenberg, J.S.; Migden, M.R.; Nguyen, T.H.; Tunnell, J.W. Pilot Clinical Study for Quantitative Spectral Diagnosis of Non-Melanoma Skin Cancer: Spectral Diagnosis of Skin Cancers. *Lasers Surg. Med.* **2010**, *42*, 876–887. [[CrossRef](#)]
42. Neittaanmäki, N.; Salmivuori, M.; Pöllönen, I.; Jeskanen, L.; Ranki, A.; Saksela, O.; Snellman, E.; Grönroos, M. Hyperspectral Imaging in Detecting Dermal Invasion in Lentigo Maligna Melanoma. *Br. J. Dermatol.* **2017**, *177*, 1742–1744. [[CrossRef](#)]

PVI

MINIATURE MOEMS HYPERSPECTRAL IMAGER WITH VERSATILE ANALYSIS TOOLS

by

Trops Roberts, Hakola Anna-Maria, Jääskeläinen Severi, Näsilä Antti, Leevi Annala, Eskelinen Matti A., Saari Heikki, Pölönen Ilkka and Rissanen Anna 2019

SPIE - Society of Photo-Optical Instrumentation Engineers, Proceedings Volume 10931, MOEMS and Miniaturized Systems XVIII; 109310W,
<https://doi.org/10.1117/12.2506366>

Reproduced with kind permission of SPIE.

Miniature MOEMS hyperspectral imager with versatile analysis tools

Roberts Trops^a, Anna-Maria Hakola^b, Severi Jääskeläinen^b, Antti Näsilä^a, Leevi Annala^b, Matti A. Eskelinen^b, Heikki Saari^a, Ilkka Pölönen^b, and Anna Rissanen^{*a}

^aVTT Technical Research Centre of Finland, Espoo, Finland

^bFaculty of Information Technology, Jyväskylä University, Jyväskylä, Finland

ABSTRACT

The Fabry-Perot interferometers (FPI) are essential components of many hyperspectral imagers (HSI). While the Piezo-FPI (PFPI) are still very relevant in low volume, high performance applications, the tunable MOEMS FPI (MFPI) technology enables volume-scalable manufacturing, thus having potential to be a major game changer with the advantages of low costs and miniaturization. However, before a FPI can be utilized, it must be integrated with matching optical assembly, driving electronics and imaging sensor. Most importantly, the whole HSI system must be calibrated to account for wide variety of unwanted physical and environmental effects, that significantly influence quality of hyperspectral data. Another challenge of hyperspectral imaging is the applicability of produced raw data. Typically it is relatively low and an application specific software is necessary to turn data into meaningful information. A versatile analysis tools can help to breach the gap between raw hyperspectral data and the user application. This paper presents a novel HSI hardware platform that is compatible with both MFPI and PFPI technologies. With an MFPI installed, the new imager can have operating range of $\lambda = 600 - 1000$ nm with FWHM of 15 – 25 nm and tuning speed of < 2 ms. Similar to previous imager in Ref. 1, the new integrated HSI system is well suited for mobile and cloud based applications due to its small dimensions and connectivity options. In addition to new hardware platform, a new hyperspectral imaging analysis software was developed. The new software used in conjunction with the HSI provides a platform for spectral data acquisition and a versatile analysis tool for a processing raw data into more meaningful information.

Keywords: Fabry-Perot interferometer, hyperspectral imager, MOEMS, VNIR, data analysis

1. INTRODUCTION

The advances in Fabry-Perot (FPI) interferometer technology has led the development of new miniaturized micro-spectrometers. These micro-spectrometers can be integrated in portable mobile devices, such as smartphones and tablets as well as specialized industrial equipment. The possibility to non-intrusively analyze spectral information of objects with mass producible mobile hyperspectral device is something that has applications in many different areas, such as agriculture, medicine, remote sensing and waste recycling.

The typical way of developing FPI based spectrometers involves selection FPI that offers best performance for the particular use case. The mechanics, optics and electronics are designed to adapt to selected FPI and imaging sensor. Unlike the previous imagers the new hardware described in this paper can be configured by interchanging FPIs and optics. With this feature, the capabilities of HSI hardware are extended and the same hardware platform can be used in more applications. Another challenge of hyperspectral imaging is the analysis of hyperspectral data cubes. The hyperspectral data cube contains the information about reflectance of light in certain wavelength range, yet unique spectral features might not be immediately apparent to the user. Each pixel in image has its own spectral response, but it is difficult to visualize and perceive such large amounts of data at once. In addition the spectra of each pixel might not even be relevant in most cases. Instead the user is typically interested in abundance or just existence of certain unique spectral features and the spatial information. This makes use of microspectrometers more challenging. It is not clear if the raw hyperspectral data contain

Further author information: (Send correspondence to Roberts Trops*)

Roberts Trops: E-mail: roberts.trops@vtt.fi, Telephone: +358-4017-6004-9, www.vtt.fi

any meaningful information. Furthermore, if there are any unique spectral features it is important to visualize in which areas of image these features occur.

Overall, it is clear that the emerging technologies for microspectrometers require a versatile and low cost hardware development platform as well as a data analysis tool, that would help the user to easily extract the most significant information out of hyperspectral data.

2. FABRY-PEROT INTERFEROMETER

Fabry-Perot interferometer is a type of optical measurement device, that utilizes two parallel mirrors to produce certain spectral response. There are two main types of FPIs that VTT is developing. These types are MOEMS FPI (MFPI) and Piezo-actuated FPI (PFPI) both of which possess different inherent advantages, when used in hyperspectral imaging applications. Both types are electrically tunable, yet each of them employ a different actuation method. Typically the selection of FPI type for HSI is based on their general features. These features are summarized in table 1 and described in Ref. 2. Overall, the PFPIs with larger optical apertures are better for high performance applications where high signal-to-noise ratio (SNR) is essential. In contrast, MFPIs are excellent for volume scaling and applications where low sensor cost is necessary.

Table 1: General features of MFPI and PFPI technologies.

Technology	PFPI	MFPI
Manufacturing	Assembled structure for small-to-medium volumes	Mass-producible wafer-level processes scalable to volume manufacturing
Optical apertures	Up to 24 (mm) allow high throughput of light	Optical apertures of 2–4 (mm)
Passband tuning	Wide tuning range and easy mirror customization	Tuning range determined by mirrors

The MEMS FPI has a near-zero loss quartz substrate for visible range (wavelengths approximately 400 – 800 nm). The figure 1 shows the structure of an MFPI chip. A drive voltage is applied to electrodes of MFPI to change the passband. If the drive voltage is exceeded, the mirrors in MFPI are pulled together by attractive electrostatic force created by the electric field between mirrors. This is called pull-in and it is typically non-reversible or not easily reversible state, depending on the conditions the pull-in occurred. Even chips that are manufactured on the same wafer will likely have different maximum drive voltage values. For practical purposes pull-in means that for the MFPI can no longer be used. This is typically avoided by characterizing MFPI chips and setting hardware or software limits on voltage applied to the MFPI.³

Unlike the PFPIs the MFPIs can be mass-produced by using modern semiconductor fabrication techniques. This allows the chips to be made at much lower price. The MFPI chips are first manufactured on wafers, then diced and finally packaged. Figure 2 shows diced MFPI chips in a tray and a single packaged PFPI. The packaging adds mechanical protection, electrical interface and additional circuitry for FPIs.

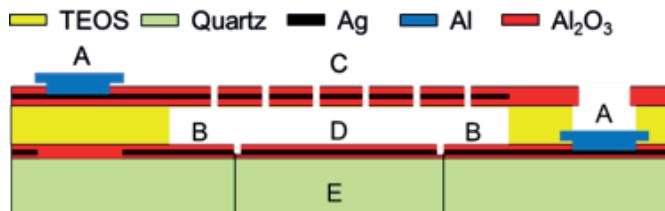


Figure 1: Structure of the MFPI. A: Contact pads; B: Actuation electrode; C: Upper movable mirror; D: Lower fixed mirror; E: Optical aperture.¹

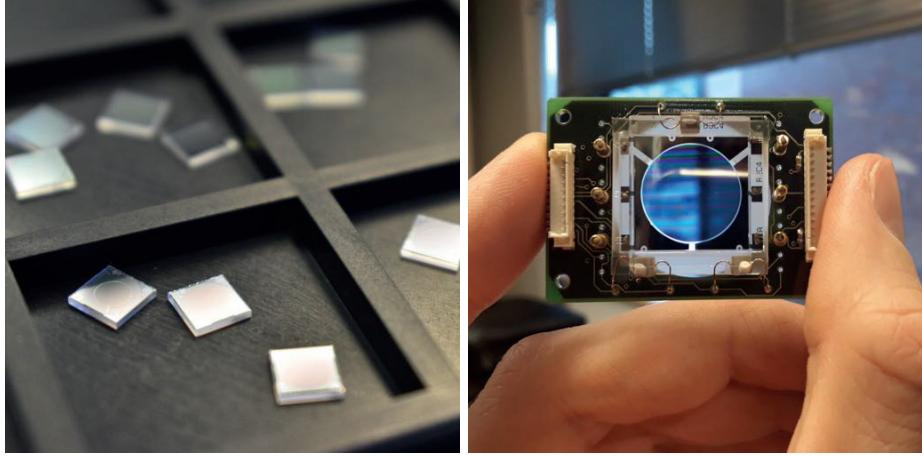


Figure 2: Unpackaged 5x5 mm MFPI chips (left) and packaged PFPI (right).

3. HYPERSPECTRAL IMAGER CONCEPT

This section describes a hyperspectral imaging hardware platform, that was developed based on MFPI and PFPI technologies. This platform is developed to be pseudo-autonomous system, that has integrated embedded Linux computer, that handles data acquisition, pre-processing, storage and transferring of data. In addition, the Linux computer has Ethernet, USB and WiFi connectivity, which can be used to add additional instruments or transfer image data. In addition to interchangeable FPIs, the hardware allows different optical assemblies.

3.1 Electronics

The new hyperspectral imager electronics design is partly based on previous hand-held imager design described in Ref. 1. The main components are camera module, which has MT9P031 5MP 12-bit CMOS sensor, Linux embedded computer and FPI controller. The camera module is connected to Linux computer via USB. The images are acquired using Python API. Similar to previous camera, the new design uses a controller board to interface between processing unit and respective FPI installed in imager.

3.2 Mechanics and optics

The optomechanical part of imager seen in figure 3 consists of aluminum baseplate, that holds optics, MFPI and sensor. In addition, it provides a way to focus and adapt the optics for different applications. The 3D printed case of imager can be seen in 4. It was designed to allow easy access to optomechanics, enclose electronics and provide tripod mounting interface. The table 2 summarizes two different PFPI and MFPI configurations, which the new HSI concept supports. These parameters are the most commonly used to describe FPIs, as well as select them for certain application, therefore having a hardware platform that is compatible with both types is a major advantage.

3.3 System calibration

After assembling HSI, it is necessary to perform system calibration. The monochromator setup shown in figure 5 is used to measure and calculate the spectral response of each pixel in camera sensor by gradually changing the wavelength of monochromatic light and the passband of FPI. Light from halogen lamp is passed into monochromator. Then, monochromatic light is passed into a integrating sphere, which has a reference detector and HSI attached. The passband of FPI and monochromator output are changed by software script running on PC. In addition, PC captures the images and calculates values for calibration table.

Table 2: Hardware configurations of HSI.

FPI type	<i>PFFI</i>	<i>MFPI</i>
Spectral range (nm)	450 – 850	600 – 1000
Spectral resolution (nm)	7 – 25 @ FWHM	15 – 25 @ FWHM
Settling time* (ms)	10	2
Clear aperture (mm)	14	2.5

*Settling time for small wavelength steps (typ. <10 nm).

3.4 Measurement calibration

In order to capture a valid hyperspectral data cube, first it is necessary to perform measurement calibration sequence. The calibration sequence consists of capturing the dark reference and a white reference. Measurement calibration has to be repeated in case of changes in the light source or significant changes in the temperature of imaging sensor and FPI. White reference target is typically a diffused target, that has uniformly high reflectance over whole measured wavelength range. The dark reference target is typically a highly absorbing material, that is used to fully cover the aperture of imager, while dark signal level is recorded.

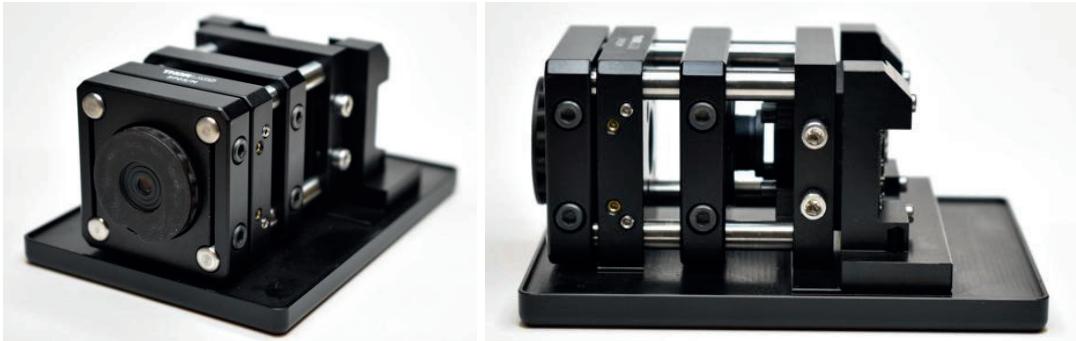


Figure 3: Interchangeable optomechanics mounted to baseplate.

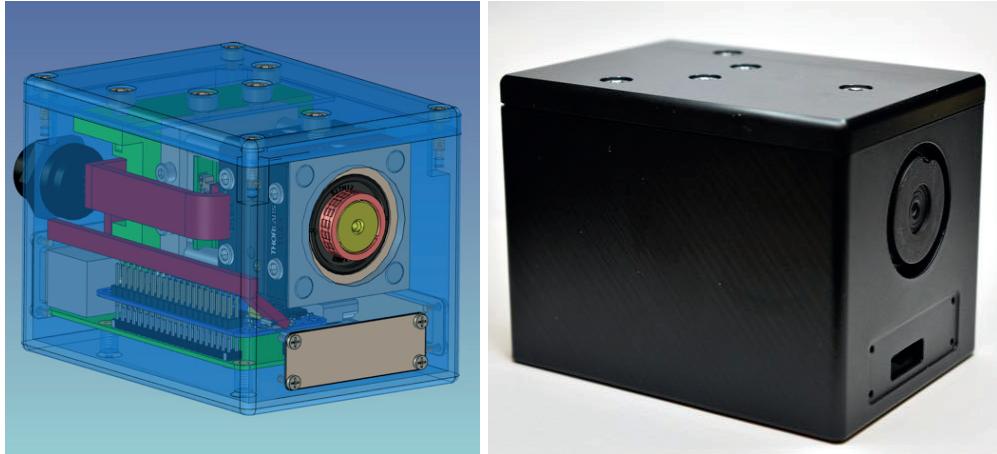


Figure 4: 3D CAD model and assembled version of hyperspectral imager.

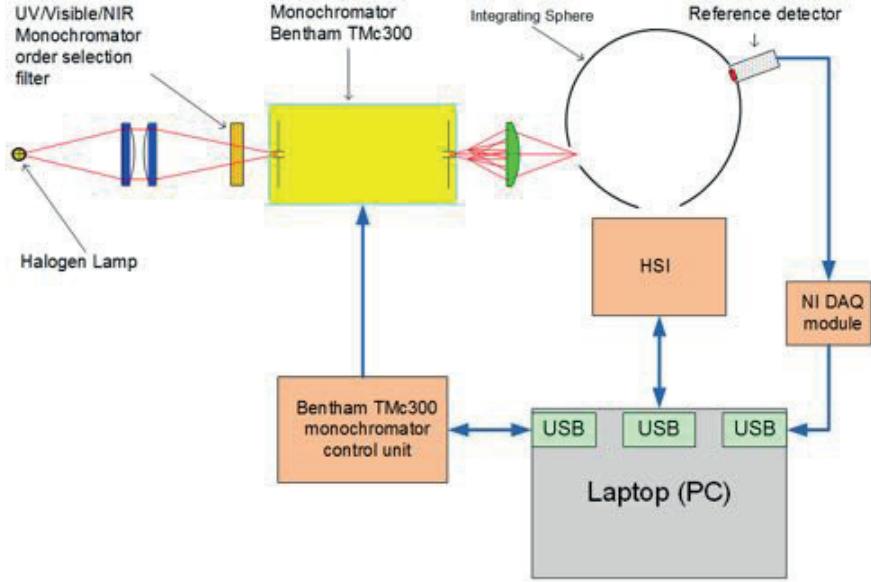


Figure 5: Monochromator calibration setup for HSI.

4. CUBEVIEW SOFTWARE

CubeView is a software, created by the Spectral Imaging Laboratory from the University of Jyväskylä, Finland. It's main purpose is to offer an easy to use and simple, but scientifically accurate interface for spectral imaging and spectral analysis. Software will be released under MIT license and one of the main strengths compared to other similar softwares is the compatibility with GenICam standard. Other worth to mention features are the different ways to use CubeView: with devices it provides easy and fast imaging and without devices it can be used for analyzing an existing spectral data cubes.

4.1 Functionality

The essence of CubeView is the deep understanding of user requests and needs by using design thinking methods through the development phases. User requests and needs were based on Spectral Imaging Laboratory's past and ongoing research. The core of CubeView uses Spectral Imaging Laboratory's python libraries, `fpiipy`,⁴ `pyspindl`⁵ and `spectracular`. CubeView has three main functionalities: LiveView, CubeView (imager) and CubeView Analyze.

LiveView is part of LiveView imager and it's main function is to offer video stream from the hyperspectral camera. LiveView has camera settings tool (based on `spectracular`), and some helpful functionalities to adjust imaging session settings. CubeView uses mainly `fpiipy` and provides the view over captured cubes. Users can easily slide through raw, reflectance or radiance cubes, check radiance pixel spectra, or select frames by wavelengths to closer attention. CubeView uses `matplotlib` back-end with visualizations and saves and reads data from netCDF format.

CubeView Analyze tool is created so that users can customize it according to their wishes and needs. Selected algorithms, including spectral unmixing based on vertex component analysis and fast least-squares approach, spectral angle mapper, principal component analysis and library of spectral indices are integrated to the basic version of CubeView Analyze. Users own analysis algorithms are easy to add to tool.

4.2 Algorithm

In spectral unmixing we assume that spectra are a linear mixture of some spectra, which are characteristic to the image and present in it. Now, let X be list of imaged spectra. Then linear mixture can be expressed in

matrix form following way

$$X = YM, \quad (1)$$

where $M \subset R^{k \times n}$ is the mixing matrix and $Y \subset R^{d \times k}$ are characteristic spectra, which are called endmembers. Here n is number of spectra in image, d is number of spectral bands and k is number endmembers. To solve M from the equation it is necessary to determinate the endmembers Y .

There are different ways to approach to determinate Y from the imaged data. If we rely on data's geometry, we can project data to the low dimensional space and try to determinate convex hull, which is covering all or majority of data points. Now, vertices of this hull are considered as endmembers. There are several methods, which are developed to detect these vertices, such as the vertex component analysis (VCA)⁶ or Pixel-Purity Index (PPI).⁷ Our implemented unmixing method uses VCA to determinate endmembers.

As evident there are several ways to determinate M after endmember detection. If we discard some boundary conditions, most easiest way to solve M is to use least-square method with pseudoinversion.⁸ It holds that

$$(Y^T Y)^{-1} Y^T X = M. \quad (2)$$

This quite simple matrix operation solves M for us.

Spectral angle mapper is relatively simple measurement to compare similarity of two spectra.⁹ We measure angle between spectra \mathbf{x} and \mathbf{y} by calculating angle between these two vectors. Spectral angle

$$\theta = \cos^{-1} \frac{\mathbf{x} \cdot \mathbf{y}}{\|\mathbf{x}\| \cdot \|\mathbf{y}\|}, \quad (3)$$

where $\|\cdot\|$ is euclidean norm of the vector. By selecting some threshold for the angle θ is SAM actually a binary classifier. In analysis example user has to select reference spectra from the spectral image.

With principal component analysis (PCA) it is possible to search those components which are causing most of the variance within recorded spectra. The difference between PCA and spectral unmixing is that PCA doesn't take account any physical meaning. Thus we will just see which areas of the image is causing variance, while spectral unmixing can actually tell us, which spectra's are present in the data cube. PCA's implementation is straight forward. First, we center data around origo by extracting mean of each spectra. Then we calculate covariance matrix. At last singular value decomposition is done on the covariance matrix. As a results we will get singular values and singular vectors. Here singular values tell's us how much of the variance is explained by corresponding principal components. Now, principal components are calculated by multiplying spectra with singular numbers.

Spectral indices are ratios, sums or different arithmetic combinations of spectral bands. They return single map, which can correlate with some properties of the images object. For example, in the remote sensing of vegetation indices such as normalized difference vegetation index (NDVI) or modified chlorophyll absorption in reflectance index (MCARI). NDVI is calculated as

$$\frac{NIR - RED}{NIR + RED}, \quad (4)$$

where NIR is some near infrared band (between 0.7 to 1 μm) and RED is band from red region of the visible light (between 0.65 to 0.7 μm). MCARI is defined as

$$\frac{(R_{850} - R_{730} - 0.2 \times (R_{850} - R_{570}))}{R_{730}}, \quad (5)$$

where R_i denotes to reflectance on wavelength i . Implemented indices library includes almost 300 different spectral indices.

The development focus was mainly on workflow and processes that produces magic behind the curtains: the least amount of effort is three clicks from start to whole spectral cube and one click more to start analyzing. What CubeView offers, is an effective and new way to capture cubes, show analysis results and produce quick demos to the researchers, corporate decision makers and even to consumers. It brings new innovations, wraps algorithms and scientific research in to easily approached and partly productized form.

Spectral Imaging Laboratory will release mentioned libraries and first standalone versions of CubeView during the spring 2019.

5. RESULTS

A measurement was done to test a hyperspectral imager and the new CubeView software. The selected imaging target was plastic leaves of a non-organic plant and one real leaf from a healthy plant. Without close examination both types of leaves appear to be organic to naked eye. The target was illuminated by a broad band light source to ensure that whole wavelength range of measurement is covered. The parameters of PFPI that was used in HSI for this measurement are summarized in table 2. The figure 6 shows the resulting PCA and abundance maps, as well as endmembers, which correlate to certain unique material features of the target. The measured wavelength range was 450 – 850 nm with camera set to analog gain of 10x and exposure to 120 ms. The figure 7 shows the output of spectral angle mapper. It can be seen in figure 7, that with threshold value of 0.1 the real plant leaf is highlighted, indicating a strong similarities in spectra with the real plant. Finally, the figure 8 shows a comparison of regular RGB-composite image and a NDVI image. It can be seen, that NDVI is much higher for real plant as it is expected due to chlorophyll absorption.

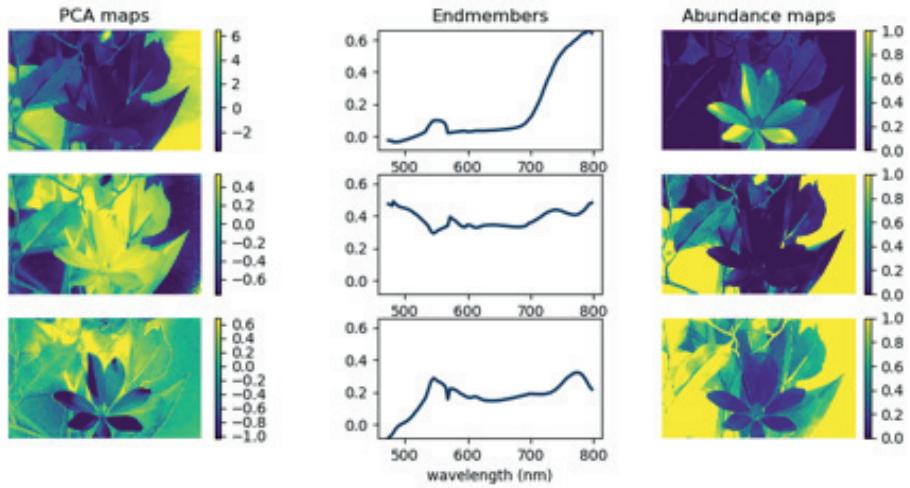


Figure 6: Results of principal component analysis (PCA) (left), and vertex component analysis (VCA) (middle and right).

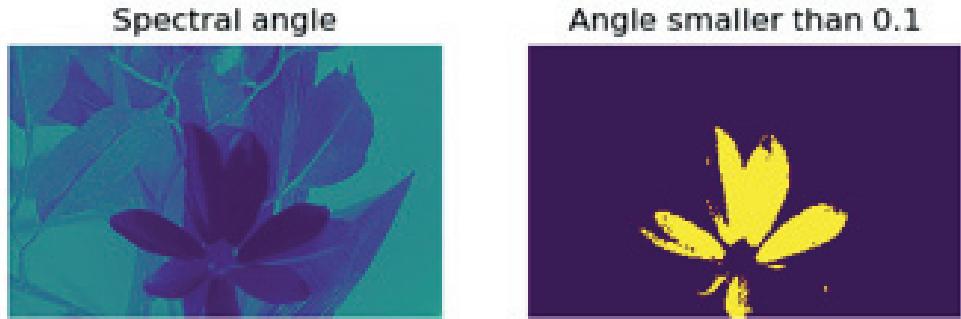


Figure 7: Spectral angle as compared to a pixel spectra from the real plant (left) and thresholded spectral angle (right).

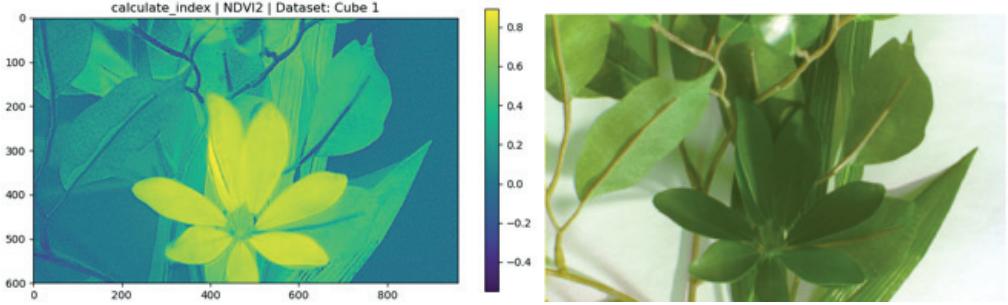


Figure 8: NDVI-index (left) and RGB-composite of three selected bands (right).

6. SUMMARY AND CONCLUSIONS

This paper presented a way to mitigate two common problems that are associated with hyperspectral imagery. The first is the unavailability of flexible and affordable imager hardware platform. The other is the raw spectral data, that are typically unusable for most of applications without an analysis algorithm. It was shown that it is possible to develop compact, low cost, portable, hyperspectral imager, that supports both MFPI and PFPI technologies. As result, the hardware platform can be used in wider range of applications, further reducing costs and increasing availability of hyperspectral imaging. In addition to the hardware, the newly developed software tool was tested with a hyperspectral camera system and it was shown, that it is possible to use PCA analysis to find and highlight the unique spectral features within raw data cube. Therefore, the user is given quick access to more visually understandable spectral information.

REFERENCES

- [1] Näsilä, A., Trops, R., Stuns, I., Havia, T., Saari, H., Guo, B., Ojanen, H. J., Akujärvi, A., and Rissanen, A., “Hand-held mems hyperspectral imager for vnir mobile applications,” *Proc.SPIE* **10545**, 10545 – 10545 – 9 (2018).
- [2] Rissanen, A., Guo, B., Saari, H., Näsilä, A., Mannila, R., Akujärvi, A., and Ojanen, H., “Vtt’s fabry-perot interferometer technologies for hyperspectral imaging and mobile sensing applications,” *Proc.SPIE* **10116**, 10116 – 10116 – 12 (2017).
- [3] Rissanen, A., Saari, H., Rainio, K., Stuns, I., Viherkanto, K., Holmlund, C., Nääkki, I., and Ojanen, H., “Mems fpi-based smartphone hyperspectral imager,” *Proc.SPIE* **9855**, 9855 – 9855 – 16 (2016).
- [4] Eskelinen, M. A. and Hämäläinen, J., “Fabry-perot imaging in python.” <https://github.com/silmae/fpipy> (2018). [Online; accessed 19-Dec-2018].
- [5] Annala, L., Eskelinen, M., Hämäläinen, J., Riihinne, A., and Pölönen, I., “Practical approach for hyperspectral image processing in python,” in [*International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*], **42**(3), International Society for Photogrammetry and Remote Sensing (2018).
- [6] Nascimento, J. M. and Dias, J. M., “Vertex component analysis: A fast algorithm to unmix hyperspectral data,” *IEEE transactions on Geoscience and Remote Sensing* **43**(4), 898–910 (2005).
- [7] Chang, C.-I. and Plaza, A., “A fast iterative algorithm for implementation of pixel purity index,” *IEEE Geoscience and Remote Sensing Letters* **3**(1), 63–67 (2006).
- [8] Lawson, C. L. and Hanson, R. J., [*Solving least squares problems*], vol. 15, Siam (1995).
- [9] Yuhas, R. H., Goetz, A. F., and Boardman, J. W., “Discrimination among semi-arid landscape endmembers using the spectral angle mapper (sam) algorithm,” in [*JPL, Summaries of the Third Annual JPL Airborne Geoscience Workshop*], **1**, 147–149 (1992).