Photometric Methods for Autonomous Tree Species Classification and NIR Quality Inspection

INNA VALIEVA

Master of Science Thesis
Stockholm, Sweden 2015
Photometric Methods for Autonomous Tree Species Classification and NIR Quality Inspection

Inna Valieva
Abstract

In this paper the brief overview of methods available for individual tree stems quality evaluation and tree species classification has been performed. The use of Near Infrared photometry based on conifer's canopy reflectance measurement in near infrared range of spectrum has been evaluated for the use in autonomous forest harvesting. Photometric method based on the image processing of the bark pattern has been proposed to perform classification between main construction timber tree species in Scandinavia: Norway spruce (*Picea abies*) and Scots Pine (*Pinus sylvestris*). Several feature extraction algorithms have been evaluated, resulting two methods selected: Statistical Analysis using gray level co-occurrence matrix and maximally stable extremal regions feature detector. Feedforward Neural Network with Backpropagation training algorithm and Support Vector Machine classifiers have been implemented and compared. The verification of the proposed algorithm has been performed by real-time testing.

Keywords: RGB imaging, NIR imaging, Artificial Neural Networks, Tree species classification, Gray Level Co-occurrence Matrix, Maximally Stable Extremal Regions.
This project has been performed in a framework of the Master Thesis project at the department of Machine Design at the Royal University of Technology (KTH) in Stockholm in collaboration with Komatsu Forest AB during the spring semester of 2015. I would like to express my gratitude to Control Systems department at Komatsu for providing the opportunity of conducting this project and gaining insight into the forestry field and the company’s workflow. I would like to give special thanks to my supervisor Peter Assarsson, who supported this project with useful information, feedback, direction and ideas throughout the whole project. Many thanks to Ulf Sellgren for academic support and supervision of this project at KTH.

Inna Valieva

Stockholm, June 7, 2015
**NOMENCLATURE**

**Notations**

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$I$</td>
<td>Image matrix, pixels</td>
</tr>
<tr>
<td>$a$</td>
<td>Length of the long axis of the ellipses approximating MSER region, pixels</td>
</tr>
<tr>
<td>$b$</td>
<td>Length of the short axis of the ellipses approximating MSER region, pixels</td>
</tr>
<tr>
<td>$\mu_{a/b}$</td>
<td>Mean value $a/b$ for all ellipses, pixels</td>
</tr>
<tr>
<td>$S_{a/b}$</td>
<td>Standard deviation of $a/b$ ratio, pixels</td>
</tr>
<tr>
<td>$N_{\text{low}}$</td>
<td>Number of ellipses with $a/b&lt;2$ per image</td>
</tr>
<tr>
<td>$N_{\text{high}}$</td>
<td>Number of ellipses with $a/b&gt;6$ per image</td>
</tr>
<tr>
<td>$N$</td>
<td>The total number of ellipses corresponding to detected MSER on the image k</td>
</tr>
<tr>
<td>$\mu_E$</td>
<td>Mean value of the eccentricity of the detected ellipses corresponding to MSERs.</td>
</tr>
<tr>
<td>$k$</td>
<td>image number</td>
</tr>
<tr>
<td>$i, j$</td>
<td>pixel number (In texture analysis with GLCM)</td>
</tr>
<tr>
<td>$i$</td>
<td>image sample (In Classifier Design)</td>
</tr>
</tbody>
</table>

**Abbreviations**

- ANN: Artificial Neural Network
- SVM: Support Vector Machine
- HOG: Histogram of Oriented Gradient
- MSER: Maximally stable extremal regions
- NIR: Near Infrared portion of the spectrum between 700 and 1000 nm
- RGB: Red green blue color model
- GLCM: Gray Level Co-occurrence matrix
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABSTRACT</td>
<td>3</td>
</tr>
<tr>
<td>FOREWORD</td>
<td>3</td>
</tr>
<tr>
<td>NOMENCLATURE</td>
<td>5</td>
</tr>
<tr>
<td>TABLE OF CONTENTS</td>
<td>6</td>
</tr>
<tr>
<td>1  INTRODUCTION</td>
<td>8</td>
</tr>
<tr>
<td>1.1 Background</td>
<td>8</td>
</tr>
<tr>
<td>1.2 Purpose</td>
<td>8</td>
</tr>
<tr>
<td>1.3 Delimitations</td>
<td>9</td>
</tr>
<tr>
<td>1.4 Method</td>
<td>10</td>
</tr>
<tr>
<td>1.4.1 Product Development Methods</td>
<td>10</td>
</tr>
<tr>
<td>1.4.2 Measurement methods and computational algorithms</td>
<td>10</td>
</tr>
<tr>
<td>1.4.3 Software packages</td>
<td>11</td>
</tr>
<tr>
<td>2.1 Wood as the material and resource. Overview</td>
<td>12</td>
</tr>
<tr>
<td>2.2 Tree species classification and tree locating</td>
<td>13</td>
</tr>
<tr>
<td>2.3 Wood Quality Inspection and Wood Properties</td>
<td>14</td>
</tr>
<tr>
<td>3  THE DESIGN PROCESS</td>
<td>17</td>
</tr>
<tr>
<td>3.1 Requirement specification</td>
<td>17</td>
</tr>
<tr>
<td>3.2 Measurement Method Selection</td>
<td>17</td>
</tr>
<tr>
<td>3.3 Algorithm Overview</td>
<td>21</td>
</tr>
<tr>
<td>3.4 Training data set</td>
<td>22</td>
</tr>
<tr>
<td>3.5 Preprocessing</td>
<td>23</td>
</tr>
<tr>
<td>3.5.1 Image Rescaling</td>
<td>23</td>
</tr>
<tr>
<td>3.5.2 RGB to gray image conversion</td>
<td>24</td>
</tr>
<tr>
<td>3.5.3 Contrast Adjustment</td>
<td>24</td>
</tr>
<tr>
<td>3.6 Feature Extraction</td>
<td>27</td>
</tr>
<tr>
<td>3.6.1 Finding Connected features using Otsu thresholding</td>
<td>27</td>
</tr>
<tr>
<td>3.6.2 Maximally Stable Extremal Regions</td>
<td>30</td>
</tr>
<tr>
<td>3.6.3 Histogram of oriented gradients</td>
<td>36</td>
</tr>
<tr>
<td>3.6.4 Texture Statistics Numerical Parameters</td>
<td>39</td>
</tr>
<tr>
<td>3.7 Classifier design</td>
<td>44</td>
</tr>
<tr>
<td>3.7.1 Fuzzy logic</td>
<td>44</td>
</tr>
</tbody>
</table>
3.7.2 Neural Network Classifier: Feedforward Network ...................................................... 49
3.7.3 Neural Network Classifier: Support Vector Machine .................................................. 52
3.8 Performance Validation ................................................................................................... 55
3.8.1 Feedforward Network with MSER .............................................................................. 56
3.8.2 SVM with MSER ......................................................................................................... 57
3.8.3 SVM with GLCM......................................................................................................... 57
3.9 System verification and testing ....................................................................................... 58
3.9.1 Experimental set up ...................................................................................................... 58
3.9.2 Software ....................................................................................................................... 58

4 RESULTS 60

5 DISCUSSION AND CONCLUSIONS 64

5.1 Discussion ....................................................................................................................... 64
5.2 Conclusions ...................................................................................................................... 65

6 RECOMMENDATIONS AND FUTURE WORK 67

6.1 Recommendations ........................................................................................................... 67
  6.1.1 Vibrations measurement ............................................................................................... 67
  6.1.2 Study of the variations in the input image on the classification results ....................... 67
  6.1.3 FEM Modelling NIR Reflection of Conifer canopy .................................................... 68
6.2 Future work ....................................................................................................................... 71

7 REFERENCES 73
INTRODUCTION

This chapter describes the background, the purpose, the limitations and the methods including the product development methods, measurement methods and computational algorithms and software packages used in this project.

1.1 Background

This report is a master thesis written as a result of a project conducted at Komatsu Forest AB by a student of the master program Machine Design at the Royal Institute of Technology, KTH.

Komatsu Forest AB is one of the world's largest manufacturers of forest machines. It is a part of Japanese Komatsu Group, which is the world's second largest manufacturer of mining, forestry and construction equipment.

Rapidly growing global market demands for wood products require advanced, efficient and environmentally friendly harvesting methods and forest machinery. In scope of this project the various methods and technologies for tree species classification have been studied and computer vision-based method for autonomous tree species classification has been proposed.

The proposed method is based on existing feature extraction and texture analysis algorithms like texture analysis using gray level co-occurrence matrix and maximally stable extremal regions feature detection applied towards RGB bark images.

Also various methods of wood quality inspection have been studied and NIR imaging (0.7 µm - 1.5 µm) has been investigated as a potentially promising method for tree health and quality inspection.

1.2 Purpose

The main purpose of this work is to propose the method for autonomous tree species classification and quality inspection.

Nowadays visual inspection still remains the most dominant method of timber quality inspection and tree species classification as it has been traditionally (Kasal, 2014). Since further use of the logs is usually predefined at the harvesting site it is important to achieve precisely quality evaluation to achieve efficient management of wood processing and usage. The average price for saw logs timber is approximately 40% higher than for the pulp wood of the same kind according to (Skogsstyrelsen, 2013).
1.3 Delimitations

Number of delimitations summarized below has been made in this project to simplify the development process:

1. It is assumed that the image is taken before the falling cut by the camera positioned to obtain the tree bark and only the tree bark in the image frame, with no any other obstacles present in the image. The camera must be positioned vertically (portrait).

2. Classification is performed between two tree species Norway spruce (*Picea abies*) and Scots Pine (*Pinus sylvestris*). This delimitation has been made because these two species are the most common tree species on the Swedish timber market, accounting 40% and 38% of current forest standing volume (Royal Swedish Academy of Agriculture and Forestry, 2009).

3. The imaging is to be performed under the natural outdoor illumination.

4. Imaging and further processing has been performed on images with no blur (according to visual inspection of the bark images).

5. Bark data collection for classifier design, and for system testing has been limited to one geographic location: Umeå, Sweden.

6. The measurement methods sensitivity to the variations in illumination changes has not been studied.

7. Young pine trees with diameter at breast height (*DBH*) below 120 mm were not studied. Figure 1 a) presents the example of the young pine stem image.

8. Trees densely (more than 40% of the bark) covered by moss were not studied. Figure 1 b) and c) present the examples of delimitation and studied cases respectively.
1.4 Method

1.4.1 Product Development Methods

V-model has been chosen as the most suitable product development approach for this project, since the proposed methods were required to be tested as early as possible to identify the promising or to be discarded. V-model with each phase of the development life cycle associated with testing and validation fits this requirement. Figure 2 below presents V-model and its main product development phases.

Ishikawa Diagrams. Method used in product development to identify the causes of a certain event, introduced in 1968 by K. Ishikawa (Ishikawa, 1968). It has been used to summarize potentially interesting technologies and methods to perform the tree species classification and the quality inspection tasks.

1.4.2 Measurement methods and computational algorithms

Red Green Blue (RGB) Photometry has been used to acquire the input data for tree species classification system.

Near Infrared (NIR) Photometry. Use of NIR Photometry has been investigated for quality evaluation of the individual tree stems based on the foliage reflectance in NIR.

Fuzzy logic is a convenient way to map an input space to an output space using rules and membership functions. (Mathworks, 2015)

Artificial Neural Networks (ANN) Neural networks as the computational method have been proposed by Warren McCulloch and Walter Pitts (1943). ANNs are defined as massively parallel distributed processors made up of simple processing units, which has natural propensity for storing experimental knowledge and making it available for use. It resembles brain in two ways:

1. Knowledge is acquired by the network through the learning process.
2. Interneuron connection strength, known as synoptic weights, are used to store the acquired knowledge.

Maximally Stable Extremal Regions (MSER) is a method of blob detection in images originally proposed by Matas et al. It has been used for feature extraction to obtain the statistically significant parameters that allow classifying between spruce and pine tree classes.
Monte Carlo Simulation is a simulation method for studying statistical confidence limits in distribution analysis. The Monte-Carlo simulation has been used to generate the large synthetic data sets based on statistical parameters of the experimental data set (Monte Carlo Simulations, 2015) to study the ANN performance depending on the training data set size.

1.4.3 Software packages

Matlab 2014b development environment has been used in this project together with extension packages briefly listed below. Image Processing toolbox has been used for image processing and feature extraction, Data Acquisition Toolbox has been used for real time image acquisition from the USB camera. Fuzzy Logic and Neural Networks toolboxes have been used for the classifier design.

Deployment Toolbox has been used to compile the stand alone executable *.exe file of the test Software Application for Windows.
2.1 Wood as the material and resource. Overview

Current global forest resources account over 4 billion hectares equivalent to 31% of total land area (Food and Agriculture Organization of the United Nations, 2010). Figure 3 presents the distribution of the 25 most common tree species\(^1\). Pinus, Quercus, Picea, Abies and Fagus make up almost a third of the global forest resources (Food and Agriculture Organization of UN, 2005).

In Sweden the total area of forest land is 28 million hectares, what accounts 70% of the total land area (Swedish Wood, 2012). Norway spruce (*Picea abies*) and Scots Pine (*Pinus sylvestris*) are the key timber market tree species, accounting 40 and 38% of the total forest standing volume (Royal Swedish Academy of Agriculture and Forestry, 2009).

The life cycle of the wood as a material and resource could be summarized briefly in the following steps: **Growth** - **Harvesting and processing** - **Use** - **Reuse** (Kolb, 2008).

During the **Growth** a tree produces carbohydrates in photosynthesis process and the oxygen (O\(_2\)) from water, carbon dioxide (CO\(_2\)) and sunlight. As a tree grows, a thin layer of cells called the cambium located under the bark generates new wood, called sapwood. (Sección Bilingüe de Tecnología en Inglés, 2015). Sapwood is softer and lighter in color than heartwood. As the tree grows the sapwood ages it, natural substances invade the sapwood and gradually convert it to heartwood. Tree stem macrostructure is described by Figure 4 below.

**Harvesting and processing** are the next steps in the wood life cycle. At this step trees are harvested, sorted into three groups: 1) construction wood: spruce; 2) construction wood: pine; 3) pulp. Then harvested logs are transported to the processing site.

**Use.** The harvested wood is further used depending on its quality and properties.

\(^1\) represent 64% of all taxa reported based on data provided by 88 countries.
Construction Wood. The wood is a high quality non homogeneous, anisotropic fiber composite material, optimally designed to resist gravity and wind loads acting on it. The wood structure is adapted to create maximum strength in stressed directions while in other directions the strength is quite low. (Thelandersson & Hans, 2003).

![Image of tree stem macrostructure]

Figure 4. Tree stem macrostructure. (Sección Bilingüe de Tecnología en Inglés, 2015).

Engineered wood products is a broad class of materials typically produced from wood that has been processed into the smaller fractions by sawing, chipping, peeling, slicing, defibratation from the pulp timber (Thelandersson & Hans, 2003).

Biomass fuel: The combustion of wood is neutral in terms of carbon dioxide. For example, the calorific value of one cubic meter of dry beech wood corresponds to about 300 liters of heating oil. It is a renewable fuel that spares the use of fossil fuels (Kolb, 2008).

Reuse. Timber building components can be recycled for their materials. This form of cascade use should be continued as long as possible. Once it is no longer reasonable to recover the materials, it is still possible to use the timber components for energy production. (Kolb, 2008).

2.2 Tree species classification and tree locating

Individual tree stems classification has received very little attention from the scientific community (Ahlem Othmani, 2013). Most of the research work related to tree classification is focused on the large scale measurements and forestry resources mapping.

Airborne Laser Scanning (ALS), (H. O. Ørka, 2009), radiometry and hyperspectral (Michele Dalponte, 2013), (John Cipar, 2004) and multispectral imaging (J. Holmgren, 2008) and (V. Heikkinen, 2010) are the most common technologies proposed for forest resource mapping. In (Baltsavias, 1999), in scope of the comparison between the laser scanning and photometry, the forests mapping is listed as one of the main application areas of ALS.

The terrestrial laser scanning LiDAR (Light Detection and Radar) has been proposed for the monitoring of carbon stocks for worldwide climate policy-making (Euronews, 2015). In (Johannes Heinzel, 2012) LiDAR data used as one of the inputs data sources for tree species classification problem in temperate forest. The tree detection based on the data from LiDAR 3d point cloud is discussed in (Tamás, 2015) as the Lidar application in forestry field.

In Cipar (2004) the possibility of classification between conifers and deciduous trees using hyperspectral reflectance images for aerial forests mapping has been investigated. The spectra show a clear separation in the NIR plateau reflectance’s between the deciduous: approximately 0.2 and coniferous forests 0.3 or greater. The shapes of the spectra for conifers and deciduous trees measured by the spectral angle are very similar. The seasonal variation in pine forests reflectance has been observed to be approximately 0.1 for summer and autumn. Individual pixel spectra have shown high variability, in the NIR plateau.
Classification of individuals stems between five tree species: hornbeam, oak, spruce, beech, pine based on the bark pattern has been performed in (Ahlem Othmani, 2013) using the data from Airborne Laser Scanning.

In (Boman, 2013) 4 texture analysis algorithms: Grey Level Co-occurrence matrix, Wavelet, Scale-Invariant Feature Transform and two classification algorithms Support vector machines (SVM) and Import vector machines (IVM) have been evaluated for tree species classification between 2 tree species: spruce, pine and the ground.

In (Ali, 2006) the attempt has been performed to detect nearby trees and estimate the distance between forest vehicle and base of trees using low cost monocular machine vision. ANN has been used for tree detection and its classification from the background. Simple heuristic distance measurement method developed has shown fairly good performance.

Automatic tree map system in harvester called Optical Tree Measurement System (OTMS) has been introduced by Ponsse. (Melkas Timo, 2011) It uses relatively low cost industrial 2D laser with an inexpensive measuring platform to produce tree, stem and map information from thinning forests. OTMS has shown 99% accuracy (100% accuracy has been achieved to locate the mature pine stands, and 97% for the young spruce and birch). Around 88% of trees could be identified before felling.

2.3 Wood Quality Inspection and Wood Properties

Mechanical behavior and properties of wood is predefined mostly its biological structure. Looking at the microstructure of the wood it could be described as the small tubes bonded together. (Thelandersson & Hans, 2003)

The chemical composition of wood consists of lignin (18–35%) and carbohydrates (65–75%): cellulose and hemicelluloses. Both are complex, polymeric materials. (Pettersen, 1984)

Wood is hydroscopic material i.e it interacts with ambient humidity, what affects the strength and stiffness characteristics. It is also nonhomogeneous and anisotropic: the properties in longitudinal direction are different from properties in transversal direction. The properties also have a high variance from log to log due to atomic natural structure.

In transversal direction the annual rings are indicator of the strength: large annual growth rings correspond to a low density and thus the lower strength. Another strength indicator is the amount and location of the knots, which are considered as the strength reducing defect caused by the tension produced in the perpendicular direction to the grain which is the weakest direction.

The compression wood is another wood strength influencing characteristic is a result of the trees reaction to the external forces: it is produced in areas with the high compression.

Strength data for the structural timber is reflecting moment, tension and compression and shear capacity of a timber element. Strength properties are determined by non-destructive testing according to standardized procedure described below in the next subchapter. Both density and annual ring width are regarded as measures of clear wood strength and stiffness.

The wood stiffness is usually expressed in terms of Modulus of elasticity what is not only predefined by wood strengths but also by its impurities and defects such as knots, slope of the grains, decays, bark pockets, top rapture and compression wood. (Thelandersson & Hans, 2003)

Quantitative characteristics used in wood quality inspection

Strength: The real strength of timber can only be measured by a destructive test. Therefore strength can only estimated from nondestructive measurements of wood characteristics (knots, density, rate of growth, modulus of elasticity, slope of grain). Several studies showed the benefit

**Modulus of Elasticity.** The stiffness can be measured almost directly by several methods: - bending machines (by bending each piece as a plank over a short span) - ultrasonic method (measuring the velocity of sound) - vibration method (measuring the natural frequency of vibration after a short impact) The ultrasonic and vibration method both need also the length and density of the piece to calculate the dynamic modulus of elasticity:

\[
MOE_{dyn} = \rho \cdot \gamma^2 = \rho \cdot (2 \cdot l \cdot f)^2
\]  

(1)

**Density** In laboratory the density according to the European standard EN 384 can be determined by two methods: - Small sample (preferred method described in ISO 3131:1975) - Whole specimen (by measurement of the mass and volume; this method is allowed only “where not all the specimens are test to failure”) The density can be measured by several methods also in a production environment: - Industrial scale: Load cells measure the mass. They are very sensitive components and it is difficult to achieve high accuracy at high feeding speed. The volume is needed to calculate the density.

**Sorting and Quality evaluation methods and technologies**

There are currently two types of grading systems:

- Visual strength grading where the evaluation is based on visual inspection of the specimen according to predefined by the grading rule number and quality of imperfections and defects.
- Machine strength grading, where the specimen is passing through machine evaluation is based on non-destructive direct or indirect measurement of relevant parameters. The most common methods used in commercially available machines are summarized below.

**Flatwise bending test** is used to measure flatwise bending stiffness (Thelandersson & Hans, 2003). The bending stiffness \( EI \) can be obtained from

\[
EI = F \frac{l^4}{48\delta}
\]  

(2)

Where \( F \) is applied force, \( l \) is a span length and \( \delta \) is deflection, \( I \) is modulus of inertia. Moe is measurement machine specific and thickness specific.

Errors sources: vibrations during the timbers passage through the machine. Main disadvantage: ends of the timber are left unevaluated.

Several commercially available wood quality inspection machines for the wood processing industries based on bending test have been found:

- Computermatic /Micromatic
- Cook Bolinder / Tecmach
- Raute Timgrader
- CLT 7200LS
- Dart
- Ersson ESG-240

**Xray or gamma rays** imaging can be used to determine the density of the wood. Commercially available wood quality inspection machines for the wood processing industries based on X-ray been found on the market:

- Euro-GreComat 702
- GoldenEye 702
- Newness XLG

Finnograder is using microwave to detect knots and gamma rays to determine wood density.
**Resonant vibration test** to determine the MOE of the timber.

The resonance frequency of a longitudinal vibration in a beam:

\[ f_{A,n} = \gamma_{A,n} \frac{n}{\rho L^2} \left( \frac{E_A}{\rho L^2} \right)^{1/2} \text{Hz} \]  

(3)

Where constant \( \gamma_{A,n} \) depends on the support conditions, \( n \) is the mode number, \( \rho \) is the density and \( L \) is the length of the wood. If both ends are not fixed and the first resonance frequency of the first axial mode is measured, MOE is expressed:

\[ E_A = 4 \rho L^2 f_{A,1}^2 \]  

(4)

Commercially available wood quality inspection machines for the wood processing industries based on resonant vibration tests have been found on the market:

- Dynagrade
- ViSCAN
- Timber grader MTG
- GradeMaster 403

The following commercially available quality grading machines for the wood processing industries based on combined measurements:

1. X-ray and flatwise bending: Euro-GreComat 704;
2. X-ray & vibration: Euro-GreComat 706, GoldenEye 706, TRU Timber grader;
3. Camera for knot and other surface defects evaluation+ flatwise bending.

Since the annual ring width is affecting the strength in (Norell, 2010) a completely automatic method for counting the number of annual rings on log end faces has been described and evaluated.
3 THE DESIGN PROCESS

3.1 Requirement specification

System Performance. The system is intended for classification between two tree species: Norway Spruce and Scots Pine, which are the key construction wood species on Scandinavian timber market. Two main classification parameters are specified:

- **Time**: Tree species classification is required to be performed when the tree is approached for harvesting or during its harvesting process from the falling cut until the second cut has to be made. Construction wood logs are cut into 6m length. The harvester head travels at 6 m/s across the tree stem cutting away the branches.

- **Classification Accuracy**: as high as possible

Operation Environment. The system is intended for outdoor operation in harsh environment noised with the wood saw dust. It should be also robust towards the outdoor noise factors: rain, snow and high variations in illumination levels.

Operation temperatures: \(-40^{\circ}C…+50^{\circ}C\)

Testing. Field testing and performance evaluation of the test results in terms of Classification accuracy are suggested to verify the system performance.

3.2 Measurement Method Selection

3.2.1 Tree species classification

Tree species classification in forest harvesting is done by the harvester operator and is based on visual inspection of the approached tree. Autonomous tree species classification is mainly used in forest mapping: terrestrial or aerial on the larger scale.

Figure 5. Tree species classification methods.
However methods currently used for the large scale mapping have been also investigated for individual tree classification. Figure 5 presents the summary of the methods available for tree species classification.

Methods in diagram above were analysed and RGB photometry of the bark pattern has been proposed for further investigation due to relative implementation simplicity and low hardware costs. Other technologies were viewed as complementary to the chosen method to increase the classification accuracy or to be used for autonomous navigation and mapping.

The spruce and pine have a characteristic bark pattern. Figure 6 presents the bark pattern images of spruce and pine.

![Bark pattern images](image)

**Figure 6.** Bark pattern images a) pine at the breast height; b) pine 4 m high from the ground; c) spruce.

The characteristic features of the spruce and pine bark:

1. Bark Crack length 5-20 cm;
2. Bark Crack width ;
3. Pattern: spruce circular fish skin like, while pine – long 5-20 cm deep 5 mm longitude grooves;
4. Spruce: uniform across the stem length, pine has dense and distinctive pattern with the longitude grooves in the bottom of the stem and more uniform and homogeneous texture few meters above the ground, see Figure 6 b).

### 3.2.2 Quality inspection

Currently wood quality inspection in forest harvesting is done by visual inspection and is based mainly on the operator’s skills and experience. More advanced quality inspection methods for individual tree stems are used only at timber processing site. The health of forest is also investigated in scope of forest mapping using multispectral (NIR and CIR) photometry.

Ishikawa diagram below summarizes methods available to evaluate timber quality, including methods summarized in the literature review above and proposed methods.
In scope of this project the possibility of conifers foliage using NIR photometry has been investigated to provide the input data related to health of the tree and timber quality. The method is ‘mature’ and widely used in aerial mapping and appeared to be relatively simple and cost effective.

In plants, carbohydrates are produced in chloroplasts cells by combining light (blue 450 nm and red light 650 nm), carbon dioxide and water in a photosynthesis process and produce glucose, oxygen and reflect light: green in visible spectrum and near infrared.

Monitoring the reflected by a plant radiation provides insights into how efficiently it is carrying on photosynthesis and so, into its general state of health. In multispectral imaging systems, the ratio of reflected near-infrared radiation to red radiation is used as an excellent indicator of plant stress. Photosynthetically-active healthy plant leaves strongly reflect (reflectance value around 0.9) radiation between 700 and 1000 nm in the near infrared portion of the spectrum. If plants are stressed, the amount of NIR that plants reflect decreases (reflectance value around 0.4).

Figure 7: Wood quality inspection methods used in forest harvesting, forest mapping and wood processing.

Figure 8. Vegetation reflectance for healthy and seek plants in VIS and NIR.
3.3.3 Pre-study NIR quality inspection

Primary evaluation of NIR quality inspection method has been performed in the field tests. IDS UI-1545LE-M-GL (1.3 Mpix) camera sensitive to both visible and NIR range of the spectrum has been used together with the visible light blocking filter. The transmission response of the IR-pass filter Schneider 31093 is described by Figure 9. Wide angle lens Schneider Kreuznach, Xenoplan 1.4/23-0512 has been used in the tests.

![Experimental setup: a) NIR sensitive IDS UI-1545LE-M-GL camera; b) IR-pass filter transmission response.](image)

Figure 9. Experimental setup: a) NIR sensitive IDS UI-1545LE-M-GL camera; b) IR-pass filter transmission response.

Gamma correction based on reference gray scale placed in front of the camera has been performed on the acquired images, example on Figure 10.

![Gamma correction](image)

Figure 10. Gamma correction

However the critical delimitation has been identified: direct sunlight when the white objects are reflecting as high as vegetation in NIR as it is shown in Figure 11 below.

![Identified delimitations.](image)

Figure 11. Identified delimitations.
Summarizing the challenges and delimitations of NIR for tree health and quality evaluation:

1. Feature extraction of the foliage of the tree under the investigation from the rest of surrounding vegetation is complex. The additional positioning method could be required to identify the target area and locate the NIR camera.
2. Snow is commonly covering the branches in winter;
3. Direct sunlight;
4. Shading;
5. High seasonal variance up to 10% in reflectance in NIR for the conifers stated in (John Cipar, 2004);
6. Variations in NIR radiation in atmosphere up to 18% due to humidity variation that affect the quality assessment method;
7. Challenge with measurements outside the daylight hours: photosynthesis is performed only under the sunlight. The extensive study is required on photosynthesis rates and NIR reflectance under the artificial illumination source with the spectrum similar to solar spectrum.
8. No clear evidence how much the stress level in the conifers identified using NIR imaging is related to the timber quality.

The shading and direct sunlight can be solved using the active systems like airborne laser scanning ALS imaging in NIR range, which are insensitive to illumination shadows. (Baltsavias, 1999). Due to the challenges related to NIR Photometric quality inspection method summarized above no further investigation of this method has been performed in the framework of this study.

### 3.3 Algorithm Overview

Figure 12 presents the schematic diagram of the proposed tree species classification algorithm. Several feature detection methods have been investigated in this paper: Connected Features Detection using Otsu thresholding, Maximally Stable Extremal Regions, Histogram of Oriented gradients and texture analysis using GLCM matrix.

3 classifier types have been compared: fuzzy logic, Feedforward Neural network with backpropagation learning algorithm and Support Vector Machine.
The algorithm implementation is done in two steps:

1. Classifier is designed based on statistically significant features extraction parameters derived from bark pattern images;
2. The ‘trained’ Classifier is used to classify in real-time the images captured by the camera. The image captured by the camera is passing through the same steps of pre-processing and feature extraction as the images in the training data set. Feature extraction parameters are fed into trained classifier, which returns the tree class.

To design the classifier the data set of bark images has been created. Images were pre-processed, i.e. rescaled, converted to gray and contrast adjustment has been performed.

Then feature extraction has been performed. Then the extracted features are used to design the rules and membership functions for the fuzzy logic classifier or used as inputs for training and testing the neural network classifier.

### 3.4 Training data set

Training data set have been composed from 900 RGB images of both tree classes spruces and pines. The large bark pattern data set should be used for the classifier design to cover the biological diversity of the bark pattern. Images were taken by Canon 700 D camera and 18-55mm lens, size 720x480 pixels in the Auto mode. Images were collected in the neighbourhood of Umeå, Sweden 63°49’N, 20°16’E. Images were acquired in four different days in various illumination conditions and at different time of the day and sorted in randomized order. Figures 13 and 14 below present the collaged image composed from the bark pattern images for pine and spruce.

![Bark pattern data](image)

**a) Pine bark**

**b) Spruce bark**

Figure 13. Examples of the bark pattern data.
3.5 Preprocessing

Image pre-processing includes Rescaling, contrast adjustment and conversion from RGB to gray image. Preprocessing steps are discussed briefly below.

3.5.1 Image Rescaling

Primary image resizing to 290x180 pixels using bilinear interpolation resampling technique has been performed to speed up the processing.

Let I to be an original image 720x480 pixels which is to be rescaled to 290x180 pixels image J.

Let \[ s_r = \frac{720}{290} \quad \text{and} \quad s_c = \frac{480}{180} \]

Let \[ r_f = r' \cdot s_r \quad \text{for} \quad r' = 1, \ldots, 290 \]

And \[ c_f = c' \cdot s_c \quad \text{for} \quad c' = 1, \ldots, 180 \]

Let \[ r = r_f \quad \text{and} \quad c = c_f \]

Let \[ \Delta r = r_f - r \quad \text{and} \quad \Delta c = c_f - c \]

\[
J(r', c') = I(r, c) \cdot (1 - \Delta r) \cdot (1 - \Delta c) + I(r + 1, c) \cdot \Delta r \cdot (1 - \Delta c) + I(r, c + 1) \cdot (1 - \Delta r) \cdot \Delta c + I(r + 1, c + 1) \cdot \Delta r \cdot \Delta c
\] (7)

The rescaling results using bilinear Interpolation are presented on Figure 15 below.

Figure 14. Bilinear Interpolation (Stackoverflow)

Let I to be an original image 720x480 pixels which is to be rescaled to 290x180 pixels image J.

Figure 15. Rescaling spruce bark image with bilinear interpolation.
3.5.2 RGB to gray image conversion

Images have been converted from RGB color space to the 8 bit grayscale using rgb2gray Matlab function. It converts RGB images to grayscale by eliminating the hue and saturation information while retaining the luminance.

rgb2gray converts RGB values to grayscale values by forming a weighted sum of the $R$, $G$, and $B$ components:

$$0.2989 * R + 0.5870 * G + 0.1140 * B$$

The conversion results are described by Figure 16 below.

![Figure 16. RGB to gray image conversion](image)

3.5.3 Contrast Adjustment

Contrast adjustment has been performed using histogram equalization method and `imhist` Matlab function. This method has been used to increase the global contrast to enhance the unique bark texture features on the images.

The histogram of the image with gray levels range $[0, L-1]$ is a discrete function $h(r_i) = n_i$,

where $r_i$ is the $i$th gray level and $n_i$ is the number of pixels in the image with the gray level $r_i$.

Consider an image with $r$ gray levels that are normalized $[0,1]$, where 0 corresponds to black and 1 corresponds to white. Then transformation function $T(r)$

$$s = T(r)$$

That produce a level $s$ for every pixel with intensity value $r$ in the original image. It has been assumed that transfer function satisfy the following conditions:

1. $T(r)$ is single-valued and monotonically increasing in the interval $0 \leq r \leq 1$. This condition is required to guarantee the existence of the inverse transformation. It preserves the increasing order from black to white in the output image.
2. $0 \leq T(r) \leq 1$ for $0 \leq r \leq 1$ to guarantee that the output gray levels will be in the same level as the input levels.

![Figure 17. Gray level single-valued and monotonically increasing transformation function.](image)

Then the inverse transformation from $s$ to $r$ is: $r = T^{-1}(s)$ $0 \leq s \leq 1$
Gray levels in the image could be considered as the random variables in the interval [0, 1]. Let \( p_r(r) \) and \( p_s(s) \) to be probability density functions of random variables \( r \) and \( s \).

Then PDF of \( s \) \( p_s(s) \) is described by Equation (8)

\[
p_s(s) = p_r(r) \left| \frac{dr}{ds} \right|
\]

Thus the PDF of transformed variable \( s \) is determined by gray-length PDF of the input image and by chosen transformation function.

\[
s = T(r) = \int_0^r p_r(w)dw,
\]

where \( w \) is integration constant. Then

\[
\frac{ds}{dr} = \frac{dT(r)}{dr} = \frac{d}{dr} \left[ \int_0^r p_r(w)dw \right] = p_r(r)
\]

Combining,

\[
p_s(s) = p_r(r) \left| \frac{dr}{ds} \right| = p_r(r) \left| \frac{1}{p_r(r)} \right| = 1 \quad 0 \leq s \leq 1
\]

For the discrete values the probabilities and summations are used instead of PDFs and integrals.

The probability of occurrence of the gray level \( r_l \) is approximated by \( p_r(r_l) = \frac{n_l}{n} \quad l=0,1,2,...,L-1 \),

The discrete version of transfer mapping function representing histogram equalization is:

\[
s_l = T(r_l) = \sum_{j=0}^{l} p(r_j) = \sum_{j=0}^{l} \frac{n_j}{n} \quad \text{for} \quad l=0,1,2,...,L-1
\]

Figures 18- 21 below show the gray level images of the spruce and pine bark together with the results of histogram equalization.

![Image a)](image_a.png) ![Image b)](image_b.png)

Figure 18. a) Original 8 bit grayscale image of the spruce bark; b) Image after histogram equalization
Figure 19. a) Histogram of the original image (Figure 18a); b) Histogram of the image after histogram equalization (Figure 18b)

Figure 20. a) Original 8 bit grayscale image of the pine bark; b) Image after histogram equalization

Figure 21. a) Histogram of the original image (Figure 20a); b) Histogram of the image after histogram equalization (Figure 20b)
3.6 Feature Extraction

Feature extraction algorithms summarized below have been investigated and compared. Matlab Image Processing toolbox with the standard filter and morphological processing functions has been used.

- Connected features detection using Otsu thresholding
- MSER: Maximally Stable Extremal Regions
- HOG: Histogram of Oriented Gradients
- Statistical Analysis using GLCM

The algorithms were primary tested on data sets of 30-70 images of the bark and the best performing algorithm is chosen for classifier design. Bark image data has been collected using Canon 700D camera. 720x480 pixels images were taken in auto mode.

3.6.1 Finding Connected features using Otsu thresholding

Otsu thresholding method proposed by N. Otsu (1979) is used to perform clustering-based image thresholding or, the gray levels reduction of a multiple gray levels image to a binary image.

The binary images resulted from finding the connected objects on the bark pattern using bwconncomp matlab function. Data set Data set of 70 images has been used.

![Image showing results of feature extraction by identification of the connected objects on the image.](image_url)

**Algorithm**

1. Image is pre-processed according to steps described above in Pre-processing.
2. Primary the pixels in the image are divided into background and foreground pixels (what is required to calculate the global threshold using Otsu method).
3. Background is estimated using morphological opening with the structuring element disk 15 pixels and 5pixels. Then the foreground is calculated by subtracting the background from the grayscale image.
4. The images were converted to the binary with the calculated threshold level by Graythresh Matlab function that computes a global threshold that can be used to convert an intensity image to a binary image with im2bw function. Level is a normalized intensity value that lies in the range $[0, 1]$. (Mathworks, 2015).

Figure 23 below presents the effect of background estimation in Otsu tresholding on the binary image that is used to find the connected features and calculate the statistics.

![Gray image of pine bark](image1)  ![Background (disk5)](image2)  ![Foreground (disk5)](image3)  ![after binary tresholding (disk5)](image4)

![Gray image of pine bark](image5)  ![Background (disk15)](image6)  ![Foreground (disk15)](image7)  ![after binary tresholding (disk15)](image8)

Figure 23. The effect of the structural element size used for morphological opening on binarization.

5. The graythresh function uses Otsu's method for global image tresholding, which chooses the threshold to minimize the intraclass variance of the black and white pixels.

6. Connected components in binary image were found using bwconncomp Matlab function.

7. Then the areas of the connected features have been calculated using the Matlab function regionprops. Area of the connected region is actual number of pixels in the region.

The Mean value and Standard Deviation have been calculated for the areas of the connected regions for the 70 images of spruce and pine bark. Figure 24 and Figures 24 and 25 present the mean and standard deviation for the case where the background has been estimated using Disk 15 pixels.
Figure 24. Mean value of the connected components identified in the image. Disk size 15 pixels.

Figure 25. Standard Deviation of the connected components identified in the image. Disk size 15 pixels.

Figure 26. Mean value of the connected components identified in the image. Disk size 5 pixels.
3.6.2 Maximally Stable Extremal Regions

MSER regions are connected areas characterized by almost uniform intensity, surrounded by contrasting background. They are constructed through a process of trying multiple thresholds. The selected regions are those that maintain unchanged shapes over a large range of thresholds.

In (K. Mikolajczyk) in scope of comparison of MSER with other region detectors as Harris–Affine, Hessian–Affine, IBR, EBR and Salient number of MSER advantages have been pointed out: It shows insensitivity to the view angle change in both textured and structured scenes, compression and variation in the light, on the other hand it shows sensitivity to blur.

Implementation details (K. Mikolajczyk):

Figure 28 below presents MSER implementation algorithm.
MSER regions identification is described briefly below. Primary pixels are sorted by pixels gray level intensity. Then pixels are marked in the image either in decreasing or increasing order and the list of growing and merging connected components and their areas is created using the union-find algorithm (Sedgewick, 1988). Further during the enumeration process, the area of each connected component is stored as a function of intensity. Then among the identified extremal regions the maximally stable are identified as the regions corresponding to thresholds were the relative area change as a function of relative change of threshold is remaining at a local minimum. In other words, the MSERs are the regions of the image where local binarization is stable over a large range of thresholds. The definition of MSER stability based on relative area change is only affine invariant both photometrically and geometrically.

Detection of MSER is related to thresholding, since every extremal region is a connected component of a thresholded image. However, no global or ‘optimal’ threshold is sought, all thresholds are tested to evaluate the stability of the connected components.

MSER regions show the following properties (J. Matas, 2002):

- Invariance to affine transformation of image intensities.
- Covariance to adjacency preserving (continuous) transformation $T : D \rightarrow D$ on the image domain.
- Stability, since only extremal regions whose support is virtually unchanged over a range of thresholds is selected.
- Multi-scale detection. Since no smoothing is involved, both very fine and very large structure is detected.
- The set of all extremal regions can be enumerated in $O(n \log \log n)$, where $n$ is the number of pixels in the image.

Figure 29 presents results obtained from MSER feature extraction algorithm applied towards the original images of spruce and pine. MSER features extracted are highlighted by different colors.

As it could be seen from Figure 29 above MERS algorithm is capable to detect circular areas on the spruce bark and elliptic areas on the pine. However it fails to detect the characteristic pattern on the pine covered by lichen (case c) images. Figure 30 presents more MSER feature extraction results for spruce and pine bark.

Figure 29. MERS feature extraction results
Identified MSER regions are approximated with ellipses with axis $a$ –long axis and $b$ is the short axis (see Figure 31 below) and the following parameters calculated for the extracted areas:

1. Mean value $a/b$ for all ellipses $\mu_{a/b}$, [pixels] corresponding to identified MSER regions in the image k. Mean value of $a/b$ is described by:

$$\mu_{a/b} = \frac{1}{N} \sum_{i=1}^{N} \left( \frac{a_i}{b_i} \right)$$  \hspace{1cm} (14)

Where $N$ is number of ellipses corresponding to MSER detected on image k.

2. Standard deviation $S_{a/b}$ of $a/b$ ratio, [pixels] is calculated by

$$S_{a/b} = \sqrt{\frac{1}{N-1} \sum_{i=1}^{N} \left( \frac{a_i}{b_i} \right)^2 - \mu_{a/b}^2}$$  \hspace{1cm} (15)

3. Number of ellipses with $a/b<2$ per image $N_{low}$;
4. Number of ellipses with $a/b > 6$ per image $N_{\text{high}}$;
5. The total number of ellipses $N$ corresponding to detected MSERs in the image $k$.
6. Mean value of the Eccentricity $\mu_E$ of identified ellipses described by:

$$\mu_E = \frac{1}{N} \sum_{i=1}^{N} E_i,$$

(16)

where $E_i$ is Eccentricity of ellipse $i$, identified in the image $k$:

$$E_i = \sqrt{1 - \frac{b_i^2}{a_i^2}}$$

(17)

7. Standard Deviation the Eccentricity of identified ellipses $S_E$, [pixels]

$$S_E = \sqrt{\frac{1}{N-1} \sum_{i=1}^{N} (E_i - \mu_E)^2}$$

(18)

8. Mean value of the Area of identified Ellipses $\mu_A$, [pixels],

$$\mu_A = \frac{1}{N} \sum_{i=1}^{N} A_i,$$

(19)

Where $A_i$ is an area of ellipse $i$ detected on the image $k$ calculated as:

$$A_i = \pi a_i b_i$$

(20)

9. Standard Deviation of the Area of identified Ellipses, [pixels]:

$$S_A = \sqrt{\frac{1}{N-1} \sum_{i=1}^{N} (A_i - \mu_A)^2}$$

(21)

Figures 32-39 below present the plots and histograms of the 9 parameters calculated for the MSER regions extracted. MSER detection area threshold range has been found by trial and error. MSER areas range [175 1600] gives the highest difference in the feature extraction parameters.
Figure 33. Number of ellipses with $a/b>6$ per image $N_{\text{high}}$

Figure 34. Number of ellipses with $a/b<2$ per image $N_{\text{low}}$

Figure 35. Mean value of the Eccentricity $\mu_E$ of identified ellipses
Figure 36. Standard deviation $S_{a/b}$ of a/b ratio

Figure 37. Mean value of the Area of identified Ellipses

Figure 38. Standard Deviation of the Area of identified Ellipses
From nine extracted parameters summarized above, the following five statistically significant parameters have been determined using statistical tail area test. **Statistical significance** (or a statistically significant result) is attained when a p-value is less than the significance level. (Krzywinski & Altman, 2013). Matlab function `ttest` has been used to perform the tail area test.

Five parameters determined from MSER feature extraction have been further used as input parameters to classifier:

1. Mean value a/b for all detected ellipses in the image;
2. Standard deviation of a/b of the identified ellipses;
3. Number of ellipses with a/b<2 per image;
4. Number of ellipses with a/b>6 per image;
5. Mean value of the Eccentricity of identified ellipses.

### 3.6.3 Histogram of oriented gradients

This feature detector technique is widely used in image processing for shape edges detection and for human detection (Navneet Dalal, 2011). It counts occurrences of gradient orientation in localized portions of an image. The object appearance and shape within an image can be described by the distribution of intensity gradients or edge directions. The image is divided into small connected regions- cells, an Example of image divided into 6 cells is shown in Figure 41.
below. Then cells are formed into blocks (on Figure 41 blocks are composed of 4 cells) Then a histogram of gradient directions is computed for the pixels within each cell. The descriptor is then is the concatenation of these histograms.

Matlab function extractHOGFeatures has been used for HOG vector extraction. HOG vectors of the size of 1x26720 have been extracted for images of spruce and pine bark. Figure 42 presents HOG extracted for 4 images of spruce and pine and Figure 43 presents the feature vectors for spruce and pine images (the 1st images from the left on Figure 42).
Figure 42. HOG vectors extracted for spruce and pine bark images.
The obtained results from HOG Feature extraction does not show the difference between the Feature vectors consisting of 26720 values extracted for spruce and pine and therefore this method has not been used further in classifier design.

### 3.6.4 Texture Statistics Numerical Parameters

The texture matrix used was the gray-level co-occurrence matrix (GLCM). In designing the GLCM for texture representation, there are three fundamental parameters that must be defined: the quantization levels of the image and the displacement and orientation values of the measurements.

**GLCM Matrix**

GLCM Matrix shows how often a pixel with certain gray-level (grayscale intensity) value \(i\) occurs horizontally adjacent to a pixel with the value \(j\). Each element \((i,j)\) in GLCM specifies the number of times that the pixel with grayscale intensity value \(i\) occurred horizontally adjacent to a pixel with value \(j\).

Figure 44 shows GLCM calculation of the 4-by-5 image I. Element (1, 1) in the GLCM contains the value 1 because there is only one instance in the image where two, horizontally adjacent pixels have the values 1 and 1. Element (1, 2) in the GLCM contains the value 2 because there are two instances in the image where two, horizontally adjacent pixels have the values 1 and 2 and etc. (Mathworks, 2015)

![Figure 44. GLCM Calculation. (Mathworks, 2015)](image_url)
To calculate GLCM matrix Matlab function graycomatrix has been used. The following statistics parameters have been calculated for the test images of spruce and pine.

**Energy** is calculated as the sum of squared elements in the GLCM. Equation (22) below presents the Energy of the GLCM calculation formula. Energy parameter has range = \([0 \ 1]\). A constant image has Energy = 1.

\[ Energy = \sum_{ij} p(i, j)^2 \]  

(22)

**Contrast** Returns a measure of the intensity contrast between a pixel and its neighbor over the whole image. Range = \([0 \ (\text{size(GLCM,1)}-1)^2]\) A constant image has Contrast=0. Contrast in GLCM has been calculated according to Equation (23)

\[ Contrast = \sum_{ij} (i-j)^2 p(i, j) \]  

(13)

**Correlation** \( Cr \) Returns a measure of how correlated a pixel is to its neighbor over the whole image. Range = \([-1 \ 1]\) . Correlation is 1 or -1 for a perfectly positively or negatively correlated image. Correlation is NaN for a constant image.

\[ Correlation = \sum_{ij} \frac{(i-\mu_i)(j-\mu_j)p(ij)}{\sigma_i \sigma_j} \]  

(24)

**Homogeneity** Returns a value that measures the closeness of the distribution of elements in the GLCM to the GLCM diagonal. Range = \([0 \ 1]\) Homogeneity is 1 for a diagonal GLCM.

\[ Homogeneity = \sum_{ij} \frac{p(ij)}{1+|i-j|} \]  

(25)

The entropy of grayscale image I is the scalar value. Entropy is a statistical measure of randomness that can be used to characterize the texture of the input image. Entropy is defined as

\[ Entropy = -\sum (p.*\log2(p)) \]

where \(p\) contains the histogram counts returned from `imhist`.

Figures 45-54 below presents the results from texture analysis on the preprocessed data set of 446 images for each tree class for two gray levels and for 8 gray levels respectively (For example for 8 gray levels graycomatrix scales the values in the image, so they are integers between 1 and 8). The number of gray-levels determines the size of the GLCM matrix.
Figure 45. Energy calculated for spruce and pine bark for 2 gray levels GLCM.

Figure 46. Contrast calculated for spruce and pine bark for 2 gray levels GLCM.

Figure 47. Correlation calculated for spruce and pine bark for 2 gray levels GLCM.
Figure 48. Homogeneity calculated for spruce and pine bark for 2 gray levels GLCM.

Figure 49. Entropy calculated for spruce and pine bark for 2 gray levels GLCM.

Figure 50. Energy calculated for spruce and pine bark for 8 gray levels GLCM.
Figure 51. Contrast calculated for spruce and pine bark for 8 gray levels GLCM.

Figure 52. Correlation calculated for spruce and pine bark for 8 gray levels GLCM.

Figure 53. Homogeneity calculated for spruce and pine bark for 8 gray levels GLCM.
Figures 45 - 54 above show high variance in extracted parameters, what is the most likely due to the variation in illumination. The data set has been acquired in different days and in different illumination and weather conditions. Four parameters from the statistical analysis on GLCM: Contrast, Correlation, Entropy and Homogeneity with 8 gray levels have been selected to be used as inputs to the classifier.

As the result of this chapter two feature extraction algorithms: 1. MSER (with five parameters: mean value of a/b, standard deviation of a/b, number of ellipses with a/b<2 per image and number of ellipses with a/b>6 per image and mean value of the Eccentricity of identified ellipses); 2. Statistical analysis on GLCM (with four parameters: Contrast, Correlation, Entropy and Homogeneity) have been selected for the inputs to the classifier.

### 3.7 Classifier design

In this chapter three types of classifiers have been designed: Fuzzy logic, Feedforward Neural Network and Support Vector Machine.

#### 3.7.1 Fuzzy logic

Fuzzy logic has been chosen as potentially promising classifier algorithm, since it allows the intuitive and knowledge-based design i.e. could be implemented based on statistically processed results from feature extraction. It also allows supervised control over the classification process through rules and membership functions adjustment. On the other hand it is purely experimental based and therefore there is a risk of getting an overstrained classifier towards the training data set.

Fuzzy logic is a convenient way to map an input space to an output space using rules and membership functions. (Mathworks, 2015)

Figure 55 shows the fuzzy logic classifier with five inputs and one output class, the classifier has been designed based on Mamdani type inference, chosen due to its intuitiveness and suitability to human input. Mamdani's fuzzy inference method is the most commonly seen fuzzy methodology. It was proposed in 1975 by E. Mamdani in (Mamdani & S. Assilian, 1975). Rules and membership functions were constructed based on feature extraction parameters from the training data set of 893 images described above.
To identify the range for the inputs to Fuzzy Logic Classifier the parameters have been mapped and the distribution of the feature extraction parameters has been used to construct membership functions. For each input parameter two membership functions corresponding to the tree class have been used (see Figures 56-60 below).

Figure 56. ‘Mean’ input to fuzzy logic classifier designed based on Mean a/b ratio.

Figure 57. ‘High’ input to fuzzy logic classifier designed based on Number of ellipses with a/b>6.
Figure 58. ‘Low’ input to fuzzy logic classifier designed based on Number of ellipses with a/b<2.

Figure 59. ‘Eccentricity’ input to fuzzy logic classifier designed based on Mean value of the Eccentricity.

Figure 60. ‘Standard Deviation’ input to fuzzy logic classifier designed based on Standard deviation of a/b ratio.

Two triangular membership functions corresponding to tree classes have been used to design the output of the fuzzy classifier (see Figure 62). Figure 61 below presents the surface visualization of the input output relation.
The set of 26 fuzzy rules presented on Figure 63 and summarized in Table 1 have been designed to guide the classification process.

Table 1. Fuzzy Logic Classifier Rules

<table>
<thead>
<tr>
<th></th>
<th>Mean</th>
<th>High</th>
<th>Low</th>
<th>Eccentricity</th>
<th>Std</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>pine</td>
<td>pine</td>
<td>pine</td>
<td>pine</td>
<td>pine</td>
<td>pine</td>
</tr>
<tr>
<td>2</td>
<td>spruce</td>
<td>pine</td>
<td>pine</td>
<td>pine</td>
<td>pine</td>
<td>pine</td>
</tr>
<tr>
<td>3</td>
<td>pine</td>
<td>spruce</td>
<td>pine</td>
<td>pine</td>
<td>pine</td>
<td>pine</td>
</tr>
<tr>
<td>4</td>
<td>pine</td>
<td>pine</td>
<td>spruce</td>
<td>pine</td>
<td>pine</td>
<td>pine</td>
</tr>
<tr>
<td>5</td>
<td>pine</td>
<td>pine</td>
<td>pine</td>
<td>spruce</td>
<td>pine</td>
<td>pine</td>
</tr>
<tr>
<td>6</td>
<td>pine</td>
<td>pine</td>
<td>pine</td>
<td>pine</td>
<td>spruce</td>
<td>pine</td>
</tr>
<tr>
<td>7</td>
<td>spruce</td>
<td>spruce</td>
<td>pine</td>
<td>pine</td>
<td>pine</td>
<td>pine</td>
</tr>
<tr>
<td>8</td>
<td>spruce</td>
<td>pine</td>
<td>spruce</td>
<td>pine</td>
<td>pine</td>
<td>pine</td>
</tr>
</tbody>
</table>
Fuzzy logic classifier performance tested on 892 images is summarized in Table 2 below. Larger experimental data set and membership functions adjustment could be suggested for performance improvement. Figure 64 illustrates the misclassified bark images.

Table 2. Fuzzy Logic Classifier performance

<table>
<thead>
<tr>
<th>Number of images</th>
<th>Absolute Error</th>
<th>Relative Classification Error, %</th>
<th>Relative Classification Accuracy, %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pine</td>
<td>446</td>
<td>198</td>
<td>44</td>
</tr>
<tr>
<td>Spruce</td>
<td>446</td>
<td>31</td>
<td>7</td>
</tr>
<tr>
<td>Total</td>
<td>892</td>
<td>229</td>
<td>26</td>
</tr>
</tbody>
</table>

Figure 64. Misclassified spruces bark images.
Due to the low classification accuracy of 76% the fuzzy logic classifier has not been further investigated in this project. The possible reasons for misclassification and performance improvement suggestions are summarized in Table 2 below.

Table 3. Performance Improvement tips

<table>
<thead>
<tr>
<th>Reasons for misclassification</th>
<th>Improvement tips</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Blurred Image</td>
<td>Deblurring:</td>
</tr>
<tr>
<td></td>
<td>Blind Deconvolution</td>
</tr>
<tr>
<td></td>
<td>Lucy-Richardson method</td>
</tr>
<tr>
<td></td>
<td>Wiener filter</td>
</tr>
<tr>
<td>2. Non-uniform illumination</td>
<td>Illumination correction</td>
</tr>
<tr>
<td></td>
<td>Multiple images</td>
</tr>
<tr>
<td>3. Bark pattern itself: Some old and large spruces D &gt;50 cm have similar bark pattern to pines.</td>
<td>Add input parameters to the Classifier</td>
</tr>
</tbody>
</table>

3.7.2 Neural Network Classifier: Feedforward Network

In this chapter the neural network classifier has been designed. Optimal network type and architecture has been determined, the effect of the training data set size on the network performance has been studied.

Neural Networks Basics

Neural network is a processing machine designed to model the biological brain activity. Neural networks perform computations through the process of learning. To achieve good performance, neural networks employ a massive interconnection of simple computing cells referred as “neurons” or “processing units” (Haykin, 1999).

The ability of neural networks to derive the function after the learning process determines particularly useful applications with the high complexity of mapping input and output data, where the mathematical derivation of mapping function is impractical. Neural networks are widely used for prediction tasks, classification problems and in control theory.

For the case of bark classification problem the supervised learning algorithms applied due to the availability of the mapped input-target data.

Supervised learning is defined as machine learning task of deriving a function from labeled training data. The training data set consist of a set of training examples. In supervised learning, each example is a pair consisting of an input object and a desired output value. A supervised learning algorithm analyzes the training data and produces an inferred function, which can be used for mapping new examples.

The input data in this case are the feature extraction parameters from the bark images and output: tree class.

Two neural network architectures has been investigated: Feedforward network, which is one of the most common and Support Vector Machines, that is widely used for classification tasks.

Feedforward network

Feedforward network (also referred as multilayer perceptron) consists of 3 layers of computational nodes: input layer, hidden, layer and output layer.
Input layer consists of five neurons corresponding to five feature extraction parameters from MSER feature extraction. The optimal size of the hidden layer has been determined by trial and error. Network performance with different hidden layer size is summarized in table below.

The output layer consists of two neurons, corresponding to the tree class. The input propagates through the network on the layer by layer basis.

**Training algorithm**

Levenberg-Marquardt backpropagation algorithm has been applied for network training. Backpropagation algorithms are based on the error correction learning rule. The backpropagation algorithm determines the minimum of the error function in weight space using the method of gradient descent. The combination of weights which solve the error function minimization problem is considered to be a solution of the learning problem.

Backpropagation learning algorithm consists of two passes: forward and backward. During the forward pass the synaptic weights are fixed and the signal propagates layer by layer, finally producing the output. During the backward pass the synaptic weights are adjusted according to error correction rule. (Haykin, 1999).

When a specified training pattern $x_i$ is fed to the input layer, the weighted sum of the input to the $j^{th}$ node in the hidden layer is given by:

$$Net_j = \sum w_{ij}x_j + \theta_j$$  \hspace{1cm} (27)

Equation (27) is used to calculate the aggregate input to the neuron, where $\theta_j$ is the weighted value from a bias node that always has an output value of 1. The bias node is a "pseudo input" to each neuron in the hidden layer and the output layer, and is used to overcome the problems where the values of an input pattern $x_i = 0$.

The neuron's output, which becomes the input value for the neurons in the next layer connected to it is determined by the resulting value from the activation function. The main requirement for the activation function is it has to be differentiable. In this case sigmoid activation function $O_j$ has been used described by:

$$O_j = x_k = \frac{1}{1+e^{-Net_j}}$$ \hspace{1cm} (28)

Equations (27) and (28) are used to determine the output value for node $k$ in the output layer. If the actual activation value of the output node, $k$, is $O_k$, and the expected target output for node $k$ is $t_k$, the difference between the actual output and the expected output is described by:

$$\Delta_k = t_k - O_k$$ \hspace{1cm} (29)

Then the error signal for node $k$ in the output layer can be calculated by:
\[
\delta_k = (t_k - O_k)O_k(1 - O_k) \tag{30}
\]

where the \(O_k(1 - O_k)\) is the derivative of the Sigmoid function. The change in the weight connecting input node \(j\) and output node \(k\) is proportional to the error at node \(k\) multiplied by the activation of node \(j\).

Equations (31) are used to modify the weight, \(w_{j,k}\), according to its change between the output node, \(k\), and the node, \(j\):

\[
\Delta w_{j,k} = l_r \delta_k x_j
\]

\[
w_{j,k} + \Delta w_{j,k}
\]

\[(31)\]

where \(l_r\) is the learning rate. It indicates the relative change in weights. If the learning rate is too low, the network will learn very slowly, and if the learning rate is too high, the network may oscillate around minimum point overshooting the lowest point with each weight adjustment, but never actually reaching it. The learning rate is usually in the range of \(l_r = 0.01\). (CSE)

To improve updating the weights, a modification to equation (31) is made:

\[
\Delta w^n_{j,k} = l_r \delta_k x_j + \Delta w^{(n-1)}_{j,k} \mu
\]

\[(32)\]

where \(\mu\) is a momentum term. The multiplication of the \((n-1)^{th}\) iteration of the weight change \(\Delta w_{j,k}\) with the momentum term is used to accelerate the learning process by "encouraging" the weight changes to continue in the same direction with larger steps. The momentum term also is used to prevent the learning process from stopping immediately in a local minimum by "over stepping" it as it is described in Figure 66. The typical range for the momentum term is between 0 and 1.

![Figure 66: Global and local minima of the error function.](image)

The error signal for \(j\) node in the hidden layer is described by:

\[
\delta_j = (t_j - O_j)O_j \sum (w_{j,k} \delta_k)
\]

\[(33)\]

Where the sum term adds the weighted error signal for all nodes in the output layer.

Formulas below are used to adjust the weights \(w_{ij}\), between the nodes \(i\) and \(j\).

\[
\Delta w^n_{i,j} = l_r \delta_j x_j + \Delta w^{(n-1)}_{i,j} \mu
\]

\[
w_{i,j} + \Delta w^n_{i,j}
\]

\[(34)\]

The following equation is used to calculate the global error for the all patterns:
The value of error $E$ for the well trained network is close to 0.

**The Data set**

Data set of 5 parameters obtained from MSER feature extraction determined from the training data set of 892 images.

The data set has been formed into $7 \times 892$ matrix, where 5 first rows correspond to feature extraction parameters listed below and columns are corresponding to the sample images.

1. Mean value $a/b$ for all ellipses $\mu_{a/b}$;
2. Number of ellipses with $a/b \geq 6$ per image $N_{\text{high}}$;
3. Number of ellipses with $a/b < 2$ per image $N_{\text{low}}$;
4. Mean value of the Eccentricity $\mu_{E}$;
5. Standard deviation $S_{a/b}$ of $a/b$ ratio.

Rows 6–7 contain binary values corresponding to the target tree class:

6. Spruce
7. Pine

The data set has been randomized and divided into:

1. Training: 50% of data
2. Testing: 25% of data
3. Validation: 25% of data;

**Monte Carlo simulation** has been applied to create the synthetic data set of 10,000 samples for every feature extraction parameter to design an optimally trained network. The random sequence for each of the feature extraction parameters from spruce and pine bark images is generated from the known distribution as described below. Further the data set is randomized.

\[
\begin{align*}
\text{Mean}_{\text{spruce}} &= \text{Mean}_{\text{min}}_{\text{spruce}} + (\text{Mean}_{\text{max}}_{\text{spruce}} - \text{Mean}_{\text{min}}_{\text{spruce}}) \cdot \text{rand}(N,1); \\
\text{Mean}_{\text{pine}} &= \text{Mean}_{\text{min}}_{\text{pine}} + (\text{Mean}_{\text{max}}_{\text{pine}} - \text{Mean}_{\text{min}}_{\text{pine}}) \cdot \text{rand}(N,1); \\
\text{High}_{\text{spruce}} &= \text{High}_{\text{min}}_{\text{spruce}} + (\text{High}_{\text{max}}_{\text{spruce}} - \text{High}_{\text{min}}_{\text{spruce}}) \cdot \text{rand}(N,1); \\
\text{High}_{\text{pine}} &= \text{High}_{\text{min}}_{\text{pine}} + (\text{High}_{\text{max}}_{\text{pine}} - \text{High}_{\text{min}}_{\text{pine}}) \cdot \text{rand}(N,1); \\
\text{Low}_{\text{spruce}} &= \text{Low}_{\text{min}}_{\text{spruce}} + (\text{Low}_{\text{max}}_{\text{spruce}} - \text{Low}_{\text{min}}_{\text{spruce}}) \cdot \text{rand}(N,1); \\
\text{Low}_{\text{pine}} &= \text{Low}_{\text{min}}_{\text{pine}} + (\text{Low}_{\text{max}}_{\text{pine}} - \text{Low}_{\text{min}}_{\text{pine}}) \cdot \text{rand}(N,1); \\
\text{Eccentricity}_{\text{spruce}} &= \text{Eccentricity}_{\text{min}}_{\text{spruce}} + (\text{Eccentricity}_{\text{max}}_{\text{spruce}} - \text{Eccentricity}_{\text{min}}_{\text{spruce}}) \cdot \text{rand}(N,1); \\
\text{Eccentricity}_{\text{pine}} &= \text{Eccentricity}_{\text{min}}_{\text{pine}} + (\text{Eccentricity}_{\text{max}}_{\text{pine}} - \text{Eccentricity}_{\text{min}}_{\text{pine}}) \cdot \text{rand}(N,1); \\
\text{Std}_{\text{mean}}_{\text{spruce}} &= \text{Std}_{\text{min}}_{\text{spruce}} + (\text{Std}_{\text{max}}_{\text{spruce}} - \text{Std}_{\text{min}}_{\text{spruce}}) \cdot \text{rand}(N,1); \\
\text{Std}_{\text{mean}}_{\text{pine}} &= \text{Std}_{\text{min}}_{\text{pine}} + (\text{Std}_{\text{max}}_{\text{pine}} - \text{Std}_{\text{min}}_{\text{pine}}) \cdot \text{rand}(N,1); \\
\end{align*}
\]

The data set size effect on the neural network performance and the results are discussed in Validation chapter below.

**3.7.3 Neural Network Classifier: Support Vector Machine**

In this chapter Support Vector Machine classifier in combination with two feature extraction inputs has been designed:

1. SVM with five input parameters from MSER feature extraction;
2. **SVM with four input parameters from GLCM.**

Support vector machine is a linear processing machine that is widely used in pattern classification. It classifies data by finding the optimal hyperplane for 2 dimensional and hyperplane for multidimensional data that separates all data points of one class from those of the other class in such a way so the margin the two classes is maximized. (Haykin, 1999)

The *support vectors* are the data points that are closest to the separating hyperplane; these points are on the boundary of the slab. Figure 67 below illustrates support vectors and optimal hyperline definitions.

![Figure 67: Support Vector Machines definitions (Mathworks, 2015)](image)

Where '+' corresponds to data of class 1, and '-' corresponds to data points of class 2.

Figure 67. Support Vector Machines definitions (Mathworks, 2015)

For building and training SVM classifier Matlab function has been used. Quadratic Optimization is used to find the optimal hyperplane due to computational efficiency. Consider the training sample \( Y = \{(x_i, d_i)\}_{i=1}^k \), where \( x_i \) is the input matrix for the \( i^{th} \) sample and \( d_i \) is a corresponding desired response, that is used to find the optimal hyperplane, subject to constraint

\[
d_i (w^T x_i + b) \geq 1 \quad \text{for } i = 1, 2, \ldots, k
\]  

This constraint combines two lines:

\[
w_0^T x_i + b_0 \geq 1 \quad \text{for } d_i = +1
\]

\[
w_0^T x_i + b_0 \leq -1 \quad \text{for } d_i = -1
\]

In scope of this optimization problem for the given training sample it is required to find the optimum values of weights vector \( w \) and bias such as they satisfy the given by (1.1) constraint and the weight vector minimizes the cost function

\[
\Phi(w) = \frac{1}{2} w^T w
\]

Where \( \frac{1}{2} \) is the scaling factor. This optimization problem is called primal problem and it satisfy two following conditions:

1. The cost function \( \Phi(w) \) is a convex function of \( w \)
2. The constraints are linear in \( w \).

This optimization problem could be solved using the method of Lagrange Multipliers as it is described below.
Primary the Lagrangian function is constructed:

\[
J(w, b, \alpha) = \frac{1}{2} w^T w - \sum_{i=1}^{k} \alpha_i \left[ d_i (w^T x_i + b) - 1 \right]
\]  

(39)

Where \( \alpha_i \) is Lagrange multiplier. The solution to our optimization problem is determined by the saddle point of the Lagrangian function \( J(w, b, \alpha) \), which has to be minimized in respect to \( w, b \) and \( \alpha \). Thus partial derivatives of \( J(w, b, \alpha) \) with respect to \( w \) and \( b \) set equal to 0 will give two conditions.

Condition 1:

\[
\frac{\partial J(w, b, \alpha)}{\partial w} = 0
\]  

(40)

Condition 2:

\[
\frac{\partial J(w, b, \alpha)}{\partial b} = 0
\]  

(41)

Application of Conditions 1 and 2 to the Lagrangian function gives:

\[
w = \sum_{i=1}^{k} \alpha_i d_i x_i
\]  

(42)

\[
\sum_{i=1}^{k} \alpha_i d_i = 0
\]  

(43)

At the saddle point, for each Lagrange multiplier \( \alpha_i \), the product of that multiplier with its corresponding constraint vanishes:

\[
\alpha_i \left[ d_i (w^T x_i + b) - 1 \right] = 0 \quad \text{for} \quad i=1,2,\ldots,k
\]  

(44)

By giving the linear constraints in the primal problem the optimization problem, another problem called dual problem is formulated. It has the same optimal value as the primal problem, but with the Lagrange multipliers giving the optimal solution.

The following duality theorem could be stated: (Bertsekas, 1995):

a. If primal problem has an optimal solution, the dual problem also has an optimal solution, and the corresponding optimal values are equal;

b. In order for the \( w_0 \) to be an optimal primal solution and \( \alpha_0 \) to be an optimal dual solution, it is necessary and sufficient that \( w_0 \) is feasible for the primal problem and

\[
\Phi(w_0) = J(w_0, b_0, \alpha_0) = \min_{w} J(w, b_0, \alpha_0)
\]

The Equation above is expanded term by term into:

\[
J(w, b, \alpha) = \frac{1}{2} w^T w - \sum_{i=1}^{k} \alpha_i d_i w^T x_i - b \sum_{i=1}^{k} \alpha_i d_i + \sum_{i=1}^{k} \alpha_i
\]  

(45)

Where \( \sum_{j=1}^{k} \alpha_j d_i = 0 \) from the Condition 2 above. From (44) we get:
\[
\mathbf{w}^\top \mathbf{w} = \sum_{i=1}^{k} \alpha_i d_i \mathbf{w}^\top \mathbf{x}_i = \sum_{i=1}^{k} \sum_{j=1}^{k} \alpha_i \alpha_j d_i d_j x_i^\top x_j \tag{46}
\]

Then the dual problem is formulated:

For the given training sample \( \mathcal{Y} = \{(x_i, d_i)\}_{i=1}^{k} \) find Lagrange multipliers that maximizes the objective function

\[
Q(\alpha) = \sum_{i=1}^{k} \alpha_i - \frac{1}{2} \sum_{i=1}^{k} \sum_{j=1}^{k} \alpha_i \alpha_j d_i d_j x_i^\top x_j , \tag{47}
\]

is subject to constraints below:

\[
\sum_{i=1}^{k} \alpha_i d_i = 0 \tag{48}
\]

\( \alpha_i \geq 0 \) for \( i=1,2,\ldots,k \)

The function \( Q(\alpha) = J(w, b, \alpha) \) to be maximized depends only on the input patterns in form of a set of the dot products, \( \{x_i^\top x_j\}_{i,j=1}^{k} \). Once the optimum Lagrange multipliers \( \alpha_{0,i} \) are determined

The optimum weight vector could be computed according to Equation (49).

\[
w_0 = \sum_{i=1}^{k} \alpha_{0,i} d_i x_i \tag{49}
\]

Obtained weights \( w_0 \) are used to compute the optimum bias \( b_0 \):

\[
b_0 = 1 - w_0^\top x^{(s)} \text{ for } d^{(s)} = 1 \tag{50}
\]

### 3.8 Performance Validation

In this chapter performance validation has been performed for the following combinations of Feature extraction algorithms and classifiers:

1. Feedforward Neural Network with five MSER input parameters;
2. SVM with five MSER input parameters;
3. SVM with four input parameters from GLCM.

Performance validation has been performed in terms of the mean value of the Relative Classification Error \( \mu_{\text{E}_{\text{tot}}} \), mean value of the Relative Classification Accuracy \( \mu_{\text{A}_{\text{tot}}} \) calculated for the \( n \) number of test runs of the randomized input validation data set. The performance stability has been evaluated in terms of standard deviation of the classification accuracy \( S_{\text{E}_{\text{tot}}} \) calculated for the \( i=1,\ldots n \) number of test runs of the randomized input validation data set. Validation parameters are described by Formulas 51-53 below.

\[
\mu_{\text{E}_{\text{tot}}} = \frac{1}{n} \sum_{i=1}^{n} E_{\text{tot}}_i \tag{51}
\]

Where \( E_{\text{tot}}_i \) is the number of misclassified images during the test run \( i \).
\[ \mu_{Atot} = \frac{1}{n} \sum_{i=1}^{n} Atot_i \]  

(52)

Where \( Atot_i \) is the number of correctly classified images during the test run \( i \).

\[ S_{Atot} = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} (Atot_i - \mu_{Atot})^2} \]  

(53)

### 3.8.1 Feedforward Network with MSER

The effect of network architecture (hidden layer size) and data set size on the classification accuracy network performance have been studied. The performance summary is provided in Table 3 below.

<table>
<thead>
<tr>
<th>N</th>
<th>Data set size</th>
<th>Number of neurons in hidden layer</th>
<th>Number of randomized test runs</th>
<th>Relative Classification Accuracy Mean,%</th>
<th>Relative Classification Error Mean, %</th>
<th>Relative Classification Accuracy Standard Deviation,%</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>892</td>
<td>12</td>
<td>24</td>
<td>86.3</td>
<td>13.7</td>
<td>2.5</td>
</tr>
<tr>
<td>2</td>
<td>10 000</td>
<td>12</td>
<td>10</td>
<td>94.1</td>
<td>5.9</td>
<td>0.6</td>
</tr>
<tr>
<td>3</td>
<td>10 000</td>
<td>16</td>
<td>5</td>
<td>95.3</td>
<td>4.7</td>
<td>0.6</td>
</tr>
<tr>
<td>4</td>
<td>10 000</td>
<td>21</td>
<td>5</td>
<td>95.2</td>
<td>4.8</td>
<td>0.3</td>
</tr>
<tr>
<td>5</td>
<td>10 000</td>
<td>30</td>
<td>5</td>
<td>95.7</td>
<td>4.3</td>
<td>0.5</td>
</tr>
<tr>
<td>6</td>
<td>10 000</td>
<td>50</td>
<td>5</td>
<td>95.0</td>
<td>5.0</td>
<td>0.6</td>
</tr>
</tbody>
</table>

Mean Square Error plot for the Feedforward network performance is presented on Figure 68.

![Error histogram for Feedforward Network with MSER](image)

Figure 68. Error histogram for Feedforward Network with MSER

The best network performance in terms of relative classification accuracy 95.7 % has been reached with 30 neurons in the hidden layer and data set size of 10 000 samples.
### 3.8.2 SVM with MSER

Support Vector Machines classifier’s performance with MSER inputs is summarized in Table 4.

<table>
<thead>
<tr>
<th>N</th>
<th>Data set size</th>
<th>Number of randomized test runs</th>
<th>Relative Classification Accuracy, Mean, %</th>
<th>Relative Classification Error, Mean, %</th>
<th>Relative Classification Accuracy, Standard Deviation, %</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>892</td>
<td>24</td>
<td>85.5</td>
<td>14</td>
<td>3.65</td>
</tr>
<tr>
<td>2</td>
<td>10000</td>
<td>5</td>
<td>90</td>
<td>10</td>
<td>1.14</td>
</tr>
</tbody>
</table>

### 3.8.3 SVM with GLCM

Support Vector Machines classifier’s performance with four inputs from GLCM is summarized in Table 4.

<table>
<thead>
<tr>
<th>N</th>
<th>Data set size</th>
<th>Number of randomized test runs</th>
<th>Relative Classification Accuracy, Mean, %</th>
<th>Relative Classification Error, Mean, %</th>
<th>Relative Classification Accuracy, Standard Deviation, %</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>892</td>
<td>24</td>
<td>99.7</td>
<td>0.29</td>
<td>0.46</td>
</tr>
</tbody>
</table>

SVM classifier with GLCM inputs has shown the best classification accuracy of 99.7 % .

![Error Histogram with 20 Bins](image)

**Figure 69.** Error histogram for SVM with GLCM.

Despite the high classification accuracy of SVM classifier with GLCM inputs, the system has shown lower classification accuracy in the field tests (see Chapter 3.9 System verification and testing), further the feature extraction parameters from the real-time testing images have been compared with the data used for SVM design and high variance in extracted parameters has been detected. It has been concluded that the extended data set of the bark pattern images acquired in various illumination conditions is required to ensure the robust performance of the algorithm on the new data.
3.9 System verification and testing

System verification has been performed by field testing of the proposed algorithm. Field testing has been performed at the harvesting site near Sörmjöle, 20 km south west of Umeå. Testing procedure, experimental set up and the results are described in this chapter.

3.9.1 Experimental set up

Experimental set up consisted of RGB Camera Creative, 5 Mpixel with the USB data input and a PC with the custom build image processing and data acquisition software. Figure 70. Experimental setup.

3.9.2 Software

The simple Software for image acquisition and logging the test data has been designed. Figure 71, Figure 72 shows the main window GUI and log file example respectively. Testing has been performed using manual image acquisition triggering.

Figure 70. Experimental setup.

Figure 71. Main window GUI of testing software.
Data logging

Once the button capture image is pressed the RGB image captured by the camera is saved into the folder.

The log file *.txt is automatically created and feature extraction data, algorithm classification result and the correct class that is entered by the user are being written into it. The created *.txt log file is saved by user in the user-specified location with the user-specified name.

Verification results are expressed in terms of:

Absolute error - the number of misclassified tree stems: $E_s$ is absolute number of misclassified spruces, $E_p$ is absolute number of misclassified pines.

Relative error for every class and total described by Equations (54) – (56) below

$$RE_s = \frac{E_s}{k_{spruces}} \times 100\%$$  \hspace{1cm} (54)

$$RE_p = \frac{E_p}{k_{pines}} \times 100\%$$  \hspace{1cm} (55)

$$RE_{total} = \frac{E_p + E_s}{k_{total}} \times 100\%$$  \hspace{1cm} (56)

where $k_{spruces}$, $k_{pines}$ and $k_{total}$ is the number of evaluated images of spruces, pines and total.

The field test results are summarized and described in the Results chapter below. The relative classification error has been calculated for the 672 bark pattern test images.
This chapter provides the summary of Feedforward Neural network classifier performance in combination with MSER input shown in the field tests. Also MSER input parameters used for classifier design have been compared to MSER inputs determined in the images acquired during the testing.

Classification results from the field tests are summarized in the Table 6 below.

<table>
<thead>
<tr>
<th></th>
<th>Number of trees evaluated</th>
<th>Absolute Error</th>
<th>Relative Error</th>
<th>Relative Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spruce</td>
<td>332</td>
<td>69</td>
<td>21</td>
<td>79</td>
</tr>
<tr>
<td>Pine</td>
<td>340</td>
<td>54</td>
<td>16</td>
<td>84</td>
</tr>
<tr>
<td>Total</td>
<td>672</td>
<td>123</td>
<td>18</td>
<td>82</td>
</tr>
</tbody>
</table>

Figures 73-77 show MSER input parameters determined for the test images and training data set that has been used for classifier design.

Figure 73. Mean value a/b for detected ellipses in the image $\mu_{a/b}$ a) for the training data set ; b) for the testing dataset.

Figure 74. Number of ellipses with a/b>6 per image $N_{high}$ a) for the training data set ; b) for the testing dataset.
Figure 75. Number of ellipses with $a/b<2$ per image $N_{low}$ a) for the training data set; b) for the testing dataset

Figure 76. Mean value of the Eccentricity of identified ellipses $\mu_E$ a) for the training data set; b) for the testing dataset

Figure 77. Standard deviation of the identified ellipses $S_{a/b}$ a) for the training data set; b) for the testing dataset
Table 7 below presents the comparison of the mean and standard deviation of the feature extraction parameters determined for the training and testing data sets.

Table 8. MSER Feature extraction parameters comparison for training and testing data sets.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Mean for training data set</th>
<th>Standard Deviation for training data set</th>
<th>Mean for testing data set</th>
<th>Standard Deviation for testing data set</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Spruce</td>
<td>Pine</td>
<td>Spruce</td>
<td>Pine</td>
</tr>
<tr>
<td>µ_{a/b}</td>
<td>2.23</td>
<td>3.27</td>
<td>0.31</td>
<td>0.70</td>
</tr>
<tr>
<td>N_{high}</td>
<td>0.75</td>
<td>4.96</td>
<td>1.27</td>
<td>3.66</td>
</tr>
<tr>
<td>N_{low}</td>
<td>29.44</td>
<td>17.13</td>
<td>9.53</td>
<td>7.83</td>
</tr>
<tr>
<td>µ_{E}</td>
<td>0.82</td>
<td>0.87</td>
<td>0.03</td>
<td>0.03</td>
</tr>
<tr>
<td>S_{a/b}</td>
<td>1.03</td>
<td>2.14</td>
<td>0.43</td>
<td>0.78</td>
</tr>
</tbody>
</table>

From Figures 73-77 above and Table 7 it is possible to conclude that three feature extraction parameters listed below show relatively small variation between test and train data set images:

- Mean value a/b for all detected ellipses in the image µ_{a/b}
- Standard deviation of a/b in the identified ellipses S_{a/b}
- Mean value of the Eccentricity of identified ellipses µ_{E}

While the number of ellipses with a/b<2 per image N_{low} and number of ellipses with a/b>6 per image N_{high} show relatively high variation (especially N_{low} parameter in test images doesn’t show clear difference between the classified tree species).

Figures 78 and 79 below show the misclassified test images of pine and spruce respectively.

![Figure 78. Misclassified images of the pine bark](image-url)
Figure 79. Misclassified images of the spruce bark

Figures 80 and 81 below shows the test images of pine and spruce bark classified correctly.

Figure 80. Correctly classified images of the pine bark

Figure 81. Correctly classified images of the spruce bark.
5 DISCUSSION AND CONCLUSIONS

5.1 Discussion

The results of this study are applicable together with number of delimitations listed in Section 1.3 of this report.

This study has been limited towards classification between spruce and pine tree species, so algorithm should be also adjusted to classify the pulp wood. Other input data: like overall tree shape, or information about the branches could be used together with the bark images.

Image acquisition for the training data set and testing has been performed using different hardware: Canon 700D (auto mode resolution 720x480 pixels) and Creative USB camera (1280x720 pixels). In both cases images have been rescaled using bilinear interpolation to 290x180 pixels to speed up the processing.

The consequences of it have been identified and partly fixed during pre-testing. The black line (see on the bottom of the image that appears in the test images in some cases is misclassified as a MSER region. This has been solved by cropping 2 bottom pixels of the acquired image in testing algorithm. The field testing has been performed after cropping implementation.

Figure 82. Spruce Bark MSER feature extraction results visualization. a) Correctly classified spruces bark; b) misclassified spruce bark.
The orientation of the extracted MSER regions should be also taken into account in classification algorithm.

The camera orientation should be specified as vertical (portrait), so the distinctive elliptic pattern of the pine bark is detected.

5.2 Conclusions

In scope of this master thesis the photometric tree species classification system based on the bark pattern has been designed. Several feature extraction techniques together with three types of classifiers: Feedforward Neural Network, Support Vector Machine and Fuzzy Logic have been studied. Statistical parameters calculation on GLCM matrix and MSER feature extraction algorithms has been proved to be the best in term of the performance.

The combination of GLCM with SVM classifier has shown the highest relative classification accuracy of 99.7 % during the validation phase. This is the most common combination of classifier and feature extraction technique found in the literature for the tasks of texture classification.

The second best validation results have been obtained for MSER with Feedforward Neural Network classifier: the relative total classification accuracy of 86.3 %. The training data set size effect on the feedforward network performance has been studied. The synthetic data set of
10 000 samples has been generated using Monte Carlo Simulations. The validation results for the network trained on the large synthetic data sets has been improved up to 95.7% classification accuracy. However the network trained on the synthetic data sets has been performing worse during the field tests verification.

MSER feature extraction together with SVM Classifier has shown slightly worse performance during validation (relative classification Accuracy of 85.5 %) than GLCM with SVM and MSER Feedforward Neural Network classifier and therefor has not been chosen for the field testing.

MSER performance together with Fuzzy logic classifier has shown the worst classification performance, giving the total relative accuracy of 74 %. No field testing of this combination has been performed.

After the validation of the investigated combinations, MSER with Feedforward Network Classifier and GLCM with SVM Classifier have been selected for the field testing.

Field testing of MSER with Feedforward Neural Network classifier has shown the relative total classification accuracy of 82 %.

Despite the highest accuracy shown during the validation, GLCM with SVM has shown significantly lower accuracy than MSER in combination with Feedforward Neural Network classifier on the 40 acquired images and no more extensive field testing of this combination has been performed. The difference between the validation and testing performance could be explained by different image acquisition hardware used for the training data set acquisition (the training data set has been divided into training and validation) and field testing.

In scope of this project the possibility using NIR photometric foliage reflectance measurements to provide the input data related to health of the tree and timber quality has been investigated. Several major challenges related to NIR Photometric quality inspection method proposed have been identified:

1. Feature extraction of the foliage of the tree under the investigation from the rest of surrounding vegetation is complex;
2. Snow is commonly covering the branches in winter;
3. Direct sunlight;
4. Shading;
5. High seasonal variance up to 10% in reflectance in NIR for the conifers stated in (John Cipar, 2004);
6. Variations in NIR radiation in atmosphere up to 18% due to humidity variation that affect the quality assessment method;
7. Challenge with measurements outside the daylight hours: photosynthesis is performed only under the sunlight. The extensive study is required on photosynthesis rates and NIR reflectance under the artificial illumination source with the spectrum similar to solar spectrum;
8. No clear evidence how much the stress level in the conifers identified using NIR imaging is related to the timber quality.

Due to the identified limitations of the proposed method no further investigation has been performed in the framework of this study.
6.1 Recommendations

Various input parameters obtain by other sensors than camera could be suggested to increase the accuracy. For example, data from the laser scanner, containing the data cloud of the overall crown shape or vibrations measurements.

6.1.1 Vibrations measurement

Another distinctively different parameter for spruce and pine is Height to crone base (see Figure 85). It could be also considered as useful input to the tree species classification algorithm.

Since the classification decision could be taken even after the falling cut (but before the second cut), it is possible to evaluate the presence of branches on the stem from the falling cut up to the second cut by analyzing data from the vibration sensor, installed on the harvester head.

6.1.2 Study of the variations in the input image on the classification results

The main controllable and uncontrollable factors are summarized briefly below. It is suggested to perform the statistical analysis of input variations on the output classification accuracy.

Controllable factors are mainly related to the camera and its settings and also illumination (in case of the controlled illumination):

- ISO;
- Shutter speed;
- Aperture,
- White Balance;
- Light metering mode;
- Illumination.

Uncontrollable factors are mainly related to the site location, standing density, and the tree parameters:

- Site Location and variation in the bark pattern itself;
- Stem Diameter;
- Standing Density;
- Soil;
- Shadowing;
6.1.3 FEM Modelling NIR Reflection of Conifer canopy

The amount of NIR radiation (both beam and diffuse) falling on the canopy surface is varying depending on atmospheric conditions, air mass, location, variations in extraterrestrial radiation. It is required to estimate the variation of the NIR radiation to propagate this variation to the variation in reflectance of NIR radiation by the canopy.

Assumptions:

1. Sun is assumed to be an isotropic light source with the spectrum given by Figure 86.

   Solar spectrum (Duffie, 2013) below:

![Solar spectrum (Duffie, 2013)](image)

2. Sky has been assumed to be isotropic in FEM modelling

3. Assumed no shading of the tree under investigation

4. Variations in extraterrestrial radiation has been neglected. The studies show that there are less than 1.5% due to variations in solar activities and 3.3% due to variation of sun to Earth distance (Duffie, 2013).

5. Multiple reflectance from the canopy was not considered in analytical calculations.

Boundary conditions:

1. Air Mass (ratio of the mass of atmosphere through the beam radiation passes to the mass it would pass through if the sun is zenith.

2. Water vapor content in the atmospheric air.

3. Tree geometry Cad model

4. Cloudy or clear sky

5. Site location (altitude and latitude), day of the year

6. Ground albedo: snow, conifer trees brunches,

Analytical solution

Analytical solution includes modelling of the Spectral irradiance on the ground to be performed according to the method proposed in

1. Calculation of the global spectral irradiance $I_{\lambda}(t)$ on an inclined surface is represented by Equation (1.1) :

\[ I_{\lambda}(t) = \frac{G_{\lambda}(t)}{1 + \frac{1}{10} \frac{H}{D}} \]

\[ I_{\lambda}(t) = \frac{G_{\lambda}(t)}{1 + \frac{1}{10} \frac{H}{D}} \]
\[
I_{\lambda}(t) = I_{d\lambda} \cdot \cos(\theta) + I_{a\lambda} \left\{ \frac{I_{d\lambda} \cdot \cos(\theta)}{H_{0\lambda} D \cdot \cos(Z)} \right\} + 0.5 \left[ 1 + \cos(t) \right] \left[ 1 - I_{d\lambda} / H_{0\lambda} D \right] \]  

(1.1)

Where \( I_{d\lambda} \) is the direct irradiation on a horizontal surface;

\( H_{0\lambda} \) is the irradiance in the upper atmosphere of Earth-Sun distance an average wavelength \( \lambda \);

D is the correction factor for Earth-Sun distance;

\( Z \) is solar zenith angle;

\( \theta \) is the angle of incidence of direct beam on an inclined surface;

\( t \) is the angle of the inclined surface.

The NIR radiation signal reflected from canopy (needle) of the Norway spruce could be calculated from energy balance.

\[
a_{\lambda} I_{\lambda} = I_{\lambda} - r_{\lambda} I_{\lambda} - \tau_{\lambda} I_{\lambda} \]  

(1.2)

The \( r_{\lambda} \) reflectance, \( a_{\lambda} \) absorbance and \( \tau_{\lambda} \) transmittance are found from

The radiation in NIR recorded by the sensor for the channel \( \lambda \) is described by Equation (1.2):

\[
I_{\lambda} = \int I_{\lambda} \tau_{\text{atm}} \cos(\theta) \frac{1}{\pi} + I_{\lambda} \cos(\theta) \rho_{\lambda} \delta_{\lambda} \]  

(1.3)

Where:

- \( I_{\lambda} \) is radiation reflected from the surface of canopy;
- \( \tau_{\text{atm}} \) is total spectral transmittance;
- \( L_{\lambda} \) is radiation calculated for the channel \( \lambda \);
- \( \Delta \lambda \) is the spectral band of the channel;
- \( \delta_{\lambda} \) is sensitivity of the channel.

COMSOL modelling

COMSOL modeling performed in Geometrical Optics Toolbox (Time dependent study) and Electromagnetic waves (with applied scattering boundary condition and defined wavelength and equation)) includes two main tasks:

1. Modelling and simulation of the wave propagation of the NIR rage waves from the sun to the ground level for the boundary conditions described above and for different cases of water vapor content in the atmosphere and different Air Masses (AM1, AM2, AM3, AM5).

2. Modelling the absorption, transmittance and the reflectance of NIR by the canopy of the Norway spruce. Modelling and simulation of the foliage geometry is a very complicated task. (Mätzler, 1994, 32) Three different approaches have been proposed for canopy modelling and discussed below by increasing level of complexity:

1. Simplified model includes the study of NIR reflectance from the single needle of the tree located at different tilt angles (0-90°) to horizontal surface. In (M. G. Trojelker, 2007) the geometry of needles of Norway spruce are defined as 25-30 mm long and about 1 mm wide. Model geometry (Case 1) is presented in Appendix.

2. More detailed modelling of the pine canopy reflectance can be performed by modelling of the outline of the tree canopy as a dielectric sphere and then modifying the material properties by the factor of needle/air coverage density. Model geometry (Case 2) is presented in Appendix.
3. The elaborate model of tree geometry has been proposed to see the effect of multiple reflectance of the NIR light from the needles of the spruce. This approach is complex and extremely demanding in terms of computational costs (the import of 3d geometry below to comsol took around 1 hour). Model geometry (Case 3) is presented below.
6.2 Future work

More extensive statistical data from different geographical locations, different forest types and standing density (see suggestions summarized in Chapter 6.1.2 above) set of images has to be collected to improve the classifier’s performance.

Proposed classifier design with the Feedforward network could be implemented as a dynamically driven of recurrent network that keeps ‘learning’ in the testing process. The example of such network architecture is NARX (Non-liner autoregressive with exogenous input) described in (Haykin, 1999).

The proposed classification method is intended to be used together with positioning and tree identification. Simple photometric monocular method proposed in (Ali, 2006) or low cost optical laser (Melkas Timo, 2011) or radar positioning could be implemented to perform the positioning task.


Melkas, H. J. (n.d.). TREE MAPPING AS A NOVEL TOOL FOR WOOD HARVESTING.


SIMS (Forest-Industry-Market Studies). (n.d.). Life crown characteristics in young stands of Scots pine, Pinus sylvestris L.


