

Lappeenranta University of Technology  
School of Engineering Science  
Intelligent Computing Major

Master's thesis

**Artem Zhelezniakov**

**AUTOMATIC IMAGE-BASED IDENTIFICATION OF SAIMAA  
RINGED SEALS**

Examiners: Professor Heikki Kälviäinen  
D.Sc. (Tech.) Tuomas Eerola

Supervisors: Professor Heikki Kälviäinen  
D.Sc. (Tech.) Tuomas Eerola

## ABSTRACT

Lappeenranta University of Technology  
School of Engineering Science  
Intelligent Computing Major

Artem Zhelezniakov

### **Automatic image-based identification of Saimaa ringed seals**

Master's thesis

2015

66 pages, 32 figures, 2 tables, 1 algorithm.

Examiners:       Professor Heikki Kälviäinen  
                      D.Sc. (Tech.) Tuomas Eerola

Keywords: Saimaa ringed seals, segmentation, identification, animal biometrics, computer vision, image processing

The Saimaa ringed seal is one of the most endangered seals in the world. It is a symbol of Lake Saimaa and a lot of effort have been applied to save it. Traditional methods of seal monitoring include capturing the animals and installing sensors on their bodies. These invasive methods for identifying can be painful and affect the behavior of the animals. Automatic identification of seals using computer vision provides a more humane method for the monitoring. This Master's thesis focuses on automatic image-based identification of the Saimaa ringed seals. This consists of detection and segmentation of a seal in an image, analysis of its ring patterns, and identification of the detected seal based on the features of the ring patterns. The proposed algorithm is evaluated with a dataset of 131 individual seals. Based on the experiments with 363 images, 81% of the images were successfully segmented automatically. Furthermore, a new approach for interactive identification of Saimaa ringed seals is proposed. The results of this research are a starting point for future research in the topic of seal photo-identification.

## **PREFACE**

I wish to thank my supervisors Professor Heikki Kälviäinen and D.Sc. (Tech.) Tuomas Eerola and also biologists from University of Eastern Finland which have provided the image database of the Saimaa ringed seals.

Finally, I wish to thank hospitable Lappeenranta, Finland,

May 20th, 2015

*Artem Zhelezniakov*

# CONTENTS

<b>1</b>	<b>INTRODUCTION</b>	<b>7</b>
1.1	Background . . . . .	7
1.2	Objectives and delimitations . . . . .	9
1.3	Structure of the thesis . . . . .	10
<b>2</b>	<b>WILDLIFE PHOTO IDENTIFICATION</b>	<b>11</b>
2.1	Biometric identification of animals . . . . .	11
2.1.1	Fur and feather patterns . . . . .	12
2.1.2	Fin patterns . . . . .	12
2.1.3	Nose-prints . . . . .	13
2.1.4	Retinal patterns . . . . .	13
2.1.5	Face recognition . . . . .	14
2.1.6	Ear vessels . . . . .	14
2.1.7	Movements . . . . .	15
2.1.8	Drawbacks of the biometric methods . . . . .	15
2.2	Computer vision methods . . . . .	16
2.3	Saimaa ringed seals . . . . .	19
<b>3</b>	<b>SEGMENTATION</b>	<b>21</b>
3.1	Image segmentation . . . . .	21
3.2	Proposed segmentation algorithm . . . . .	22
3.3	Unsupervised segmentation . . . . .	23
3.4	Feature extraction and selection . . . . .	26
3.4.1	SFTA . . . . .	27
3.4.2	LBP-HF . . . . .	28
3.4.3	LPQ . . . . .	29
3.4.4	Feature selection . . . . .	30
3.5	Segment classification . . . . .	30
3.5.1	Naive Bayes classifier . . . . .	31
3.5.2	KNN classifier . . . . .	32
3.5.3	SVM classifier . . . . .	33
<b>4</b>	<b>IDENTIFICATION</b>	<b>36</b>
4.1	Proposed identification algorithm . . . . .	36
4.2	Feature extraction and selection . . . . .	37
4.2.1	PCA . . . . .	38
4.3	Identification . . . . .	40

<b>5 EXPERIMENTS</b>	<b>42</b>
5.1 Data . . . . .	42
5.2 Segmentation . . . . .	44
5.2.1 Threshold finding for unsupervised segmentation . . . . .	44
5.2.2 Labeling for training of a classifier . . . . .	46
5.2.3 Feature extraction and classification . . . . .	47
5.3 Identification . . . . .	48
5.3.1 Principal component analysis . . . . .	49
5.3.2 Identification performance with 10 seals . . . . .	52
5.3.3 Identification performance with 40 seals . . . . .	54
<b>6 DISCUSSION</b>	<b>56</b>
6.1 Proposed method . . . . .	56
6.2 Results . . . . .	56
6.3 Future research . . . . .	58
<b>7 CONCLUSION</b>	<b>59</b>
<b>REFERENCES</b>	<b>59</b>

## ABBREVIATIONS

AI	Artificial Intelligence
ANN	Artificial Neural Network
CC	Correlation Coefficient
CMS	Cumulative Match Score
CV	Computer Vision
gPb	Globalized Probability of Boundary
GPS	Global Positioning System
GT	Ground Truth
HD	Hamming Distance
k-d	k-dimensional
KNN	K-Nearest Neighbour classifier
LBP	Local Binary Patterns
LBP-HF	Local Binary Pattern Histogram Fourier
LPQ	Local Phase Quantization
NB	Naive Bayes
OWT	Oriented Watershed Transform
PDE	Partial Differential Equation
QP	Quadratic Programming
RBF	Radial Basis Function
RGB	Red Green Blue
ScSPM	Sparse coding Spatial Pyramid Matching
SFTA	Scale Invariant Feature Transform
SVM	Support Vector Machines
UCM	Ultrametric Contour Map
UEF	University of Eastern Finland
WPI	Wildlife Photo Identification
WWF	World Wide Fund for Nature

# 1 INTRODUCTION

## 1.1 Background

Saimaa ringed seals, *Pusa hispida saimensis*, are among the most endangered seals in the world, only live in landlocked lake Saimaa in Finland. In the 21st century the population size has increased slowly from approximately 240 to 310 seals. Annual pup production has varied between 44 and 66 pups and estimated annual mortality between approximately 30 and 60 seals [1, 2]. The Saimaa ringed seal is a symbol of Lake Saimaa and forms the icon of the nature conservation movement of Finland [3]. Figure 1 shows a Saimaa ringed seal in nature.



**Figure 1.** Saimaa ringed seal. [4]

The Saimaa ringed seal is Finland's only endemic mammal and it is only found in the Saimaa lake district. Its main habitats include the lake areas of the national parks of Linnansaari and Kolovesi as well as Lake Joutenvesi and Lake Pihlajavesi [3]. In the early 20th century, Saimaa ringed seals were regarded as pests. From 1882 until 1948, a bounty was paid for killing them. In 1955, the Saimaa ringed seal was protected from hunting by law because its population had become too sparse. Since that, the seal population continued to decline until the early 80s when there were 180 seals [3].

Today the protection and the monitoring of the population of Saimaa ringed seals, classified as one of the critically endangered species [5], is managed by Metsähallitus (Parks & Wildlife Finland) in cooperation with Regional Environment Centres, University of

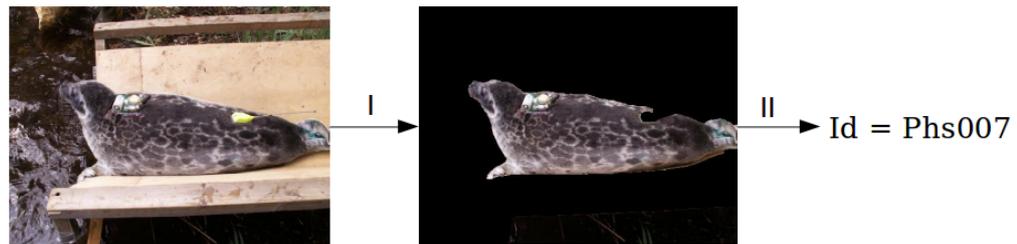
Eastern Finland (UEF), regional Employment and Economic Development Centres, and World Wide Fund for Nature (WWF). Moreover, the Saimaa ringed seal is the emblem of the Finnish Association for Nature Conservation. The protection of the seal is supervised by the Ministry of the Environment. The target is to increase the population of ringed seals in the Saimaa district to 400 individuals by the year 2020 [3].

However, the main threats to the population of Saimaa ringed seals are still presented as: net fishing, climate change related loss of snow and ice and highly fragmented population structure [1]. Because of its endangered status, the population of Saimaa ringed seals need to be closely monitored and explored. The annual monitoring procedures include the following actions: 1. Monitoring the breeding conditions and birth rate. 2. Determining causes of death. 3. Monitoring the natural state of the breeding grounds. Protective measures are being further developed through the investigation habitat requirements and the effects of various disturbances to its breeding [3].

Current monitoring and conservation methods of the Saimaa ringed seal have mostly been developed at University of Eastern Finland. Development of methods for monitoring has been focused especially on estimation of juvenile mortality and improvement of pup survival in the changing climate. In addition, applications of individual identification methods make more accurate estimates of a population size and distribution possible in the near future [6].

The current method for seal monitoring is to catch them and install sensors on their body. This method has obvious shortcomings such as the need to constantly check the sensor and to look for new individuals. Also, regular stress produced in the process of catching seals has a negative impact on lifecycles and health of seals in general.

Wildlife Photo Identification (WPI) is a technology that allows to recognize individuals and to track the movement of animal populations over time. It is based on acquiring images of animals and further identifying individuals manually or using automatic image processing methods. This approach is applicable to Saimaa ringed seals due to the presence of special patterns on the back of seals consisting of dark spots and light gray circles. This pattern is unique for each individual Saimaa ringed seal, and a sufficient number of images enables exact identification of the animal. WPI can be performed manually by experts using an image-database and experience. However, manual identification by eyes has obvious disadvantages such as slow speed and errors caused by human factor. This motivates to develop automatic methods for WPI using computer vision techniques. The approach is illustrated in Figure 2.



**Figure 2.** Main steps of the WFI algorithm: I) detection or segmentation, and II) identification.

## 1.2 Objectives and delimitations

The goal of the research is to develop an automatic image-based algorithm for a Saimaa ringed seal identification. The algorithm consists of segmenting the seal from the background on a given image to find a segment of the image containing a seal, analyzing features of seal skin patterns and identifying of an individual seal. The main idea is to present a computer vision method which can completely or partly substitute the time-consuming manual work of identifying a seal on a given picture from a large image dataset. One of the possible results can be a semimanual application which contains a seal photo as an input and suggests the most probable seal id numbers as output shows pictures of them for further manual identification by experts.

The objectives of the research are as follows:

1. Generate the manually annotated ground truth for image segmentation for a selected set of seal images.
2. Find the most suitable methods for segmentation and identification.
3. Test different features and classifiers with suitable parameters to optimize the accuracy.
4. Evaluate the performance of the selected methods.
5. Construct the complete automatic image-based algorithm for the photo-identification system.

There are several limitations imposed on the work. First, images of the seal without heavy distortions or overlapping of seals by surrounding objects are considered. Second, the identification algorithm is not obligated to be 100% accurate, but it should limit the set of possible seals to help the experts making the manual decision. The objective of the thesis is to explore, to test and to design methods that can be applied for WPI. Moreover,

the solution has to help biologists to recognise seals and to give new directions to further researches.

### **1.3 Structure of the thesis**

The rest of the thesis is structured as follows. Section 2 covers the features of the current methods used to identify animals. It introduces to the reader the main approaches applied and shows which methods can be used for identifying the Saimaa ringed seals. Section 3 discusses the segmentation, describes unsupervised methods and specifies the proposed segmentation algorithm. The justification for selecting the methods used in the thesis are presented. Section 4 considers identification, showing main approaches applied to the segmented images. Section 5 contains the results of experiments and data collection used to produce the results. Section 6 discusses the results, practical problems, and future directions of the research. Finally, Section 7 summarizes the thesis.

## **2 WILDLIFE PHOTO IDENTIFICATION**

### **2.1 Biometric identification of animals**

Since ancient times there has been need to identify animals for various purposes such as, to determine the owner of the animal, to check belonging of individual to a certain population, or to track the emergence of new species. Traditional method for these purposes was marking of specific population or each individual separately. Nowadays, not only marking but transmitters are applied by biologists to identify and track animals.

In spite of the fact that invasive methods such as marking are easy for identification purposes and have absolute accuracy of identification, these techniques can potentially disserve animals and strongly affect their behavior. Any method associated with the installation of a special marker or sensor on the body leads the stress caused by catching, processing and containment of the animal. In addition, many marking procedures, such as branding, tattooing, toe clipping, ear notching and tagging involve tissue damage and therefore cause pain and aggression. Furthermore, wearing a mark can change an appearance of the animal, social behavior, other habits and ultimately affect its survival. The ideal method of identification should work accurately, safely and without interfering with the vital functions of the animals [7].

Today, new so-called biometric technologies are gaining popularity in the tasks of identification. They are based on physical characteristics or behavioural signs of the individual. Some of these methods are used also for reliable identification of humans. An animal biometric identifier is any measurable, robust and distinctive physical, anatomical or molecular trait that can be used to uniquely identify or verify the claimed identity of an animal [8]. Therefore, a good biometric characteristic should comply with several basic rules: be readable by a sensor, not change over time, be different among all the individuals of a given population. Biometric techniques are non-invasive, do not cause suffering and aggression, and do not affect the appearance of an animal. Moreover, these methods have no effect on the behaviour and survivability of the animals, except in cases where repeated capture or handling is necessary [7]. The following paragraphs describe the most popular biometric identification methods.

### 2.1.1 Fur and feather patterns

There are a lot of animals that have exterior characteristics that are unique for each specimen. Such characteristics allow to easily distinguish between individuals of the population. Examples of such characteristics are color rings on snakes, body stripes of zebras, patches on geese' bellies and eyespots on the wings of butterflies. Today, these signs are photographed by biologists to identify animals. Problems of this method include changing light settings or background that make the identification task more difficult. However, new digital imaging techniques can sufficiently reduce these difficulties. The method is cheap and at its simplest implementation needs no more than paper and a pencil. Furthermore, the observation can be carried out at a sufficient distance to avoid to affect the life and behavior of animals [7].

The most obvious visual pattern is an external coloring of the animal. For example, zebras and tigers can be identified from their stripes, cheetahs and African penguins carry unique spot patterns, and snakes have colored rings [9]. Another study shows that individuals of lesser white-fronted geese, *Anser erythropus*, can be identified by differences in personal body patches [10]. Identification accuracy was shown to be very high, and two geese with the same pattern were not found.

### 2.1.2 Fin patterns

Photographic identification has been used since the 1970s to identify aquatic animals such as dolphins and whales. Individual bottlenose dolphins can be identified by comparing photographs of their fins which display curves, notches, nicks and tears. Whales can be distinguished by the callosity patterns on their heads [7]. Example of dorsal fins used for identification is illustrated in Figure 3.

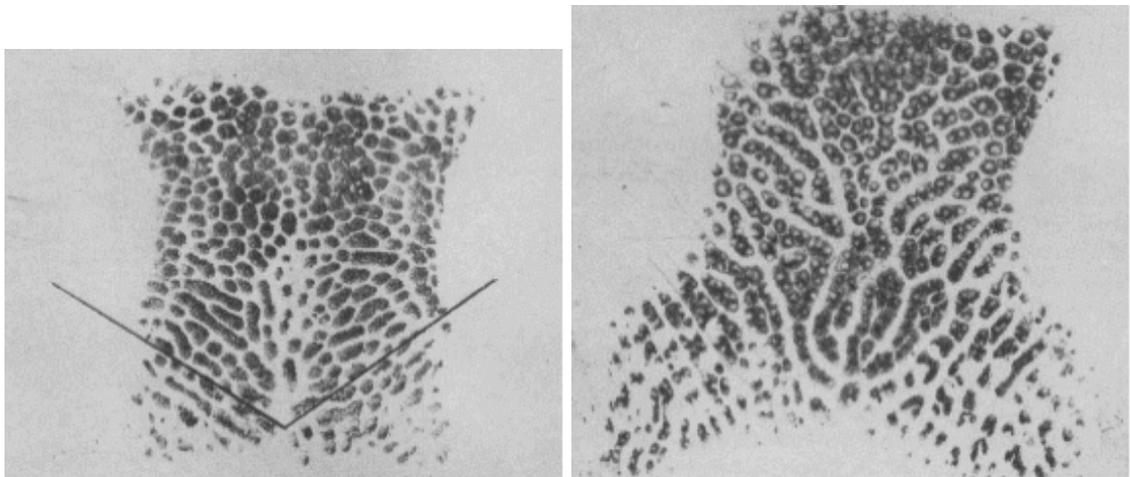


**Figure 3.** Dorsal fins of bottlenose dolphins displaying unique permanent characteristics used for their identification. [7]

### 2.1.3 Nose-prints

One of the classic methods of biometric identification of a person is fingerprinting. A similar method was used to detect cattle by Petersen [11]. However, instead of the surface of fingerprint, the nose of the animal was used. The technique was developed to avoid the potential for deception associated with traditional marking methods such as branding, tattooing, and ear tags. Nose-printing is equally well suited for the identification of sheep and cattle.

The advantages of the method is that it is relatively inexpensive and easy to use: ink is applied to the nose of the animal and then a mark is printed on the paper. On the other hand, its accuracy depends on the human factor, fingerprinting should take place with the same pressing force, and for reading and recognition of the results, a trained specialist is needed [7]. Moreover, it is recommended to use the same paper and ink to avoid possible incorrect identification of animals. Figure 4 shows examples of bovine nose prints used for identification purpose.



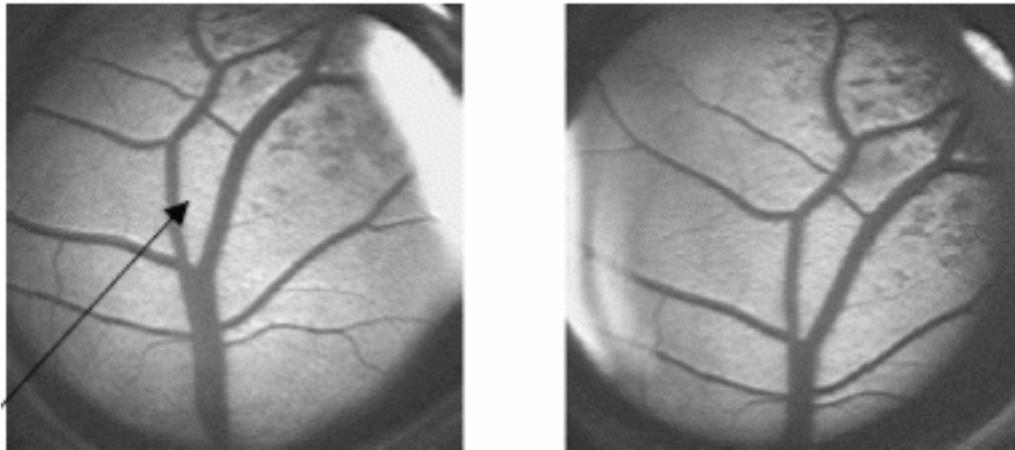
**Figure 4.** Examples of bovine nose-prints. [12]

### 2.1.4 Retinal patterns

The retina is an unique and highly precise identity of an animal. It is based upon the branching patterns of the retinal vessels which are present from birth and do not change during the animal's life [7]. The retinal pattern and Global Positioning System (GPS) coordinates can be read using a special hand-held scanner. All scans are collected in the

database and used for further processing and identification of cattle. This method is also relatively cheap. It is presented in Figure 5.

Retinal imaging and nose-prints of sheep and cattle were compared in [13]. Nose-prints are quicker to obtain than retinal scanning, but retinal scans are easier to analyze by unpractised operator [14]. A computer software for the analysis of digital pictures from both retinal scans and nose-prints makes the analysis faster, cheaper, and more reliable.



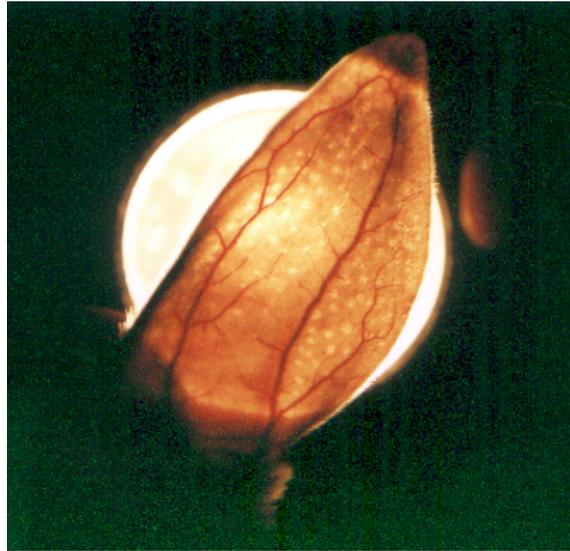
**Figure 5.** Example of matching retinal images. [13]

### 2.1.5 Face recognition

Identification of the individual by the face is a technique used by people every day. This method is also applied and adapted to identify animals, such as sheep [15]. Although individuals have quite different faces it may be difficult to create algorithms which could produce highly accurate face authentication.

### 2.1.6 Ear vessels

The unique design of the ear's circulatory system can also be considered as a unique feature that can be used to identify animals [16]. In this technology, the ear is photographed with a special backlighting that allows to capture the unique pattern and provide a detailed, high-contrast picture of the blood vessels. Node points of weaved vessels are then used for image comparison and identification of the individual. Figure 6 shows an example of ear blood vessel patterns.



**Figure 6.** An image with a torch shone through a bilby ear. [17]

### **2.1.7 Movements**

It has been suggested that aquatic animals can be identified by analyzing their movement patterns using a tri-axial accelerometry device [18]. By measuring the movements of animals in three dimensions, their movement patterns can be stored and these can be used to diagnose aberrant behavioral patterns, such as those associated with infections. Accelerometry may have the potential to be a powerful tool to produce maps for conservation purposes where animal movements can be plotted [7].

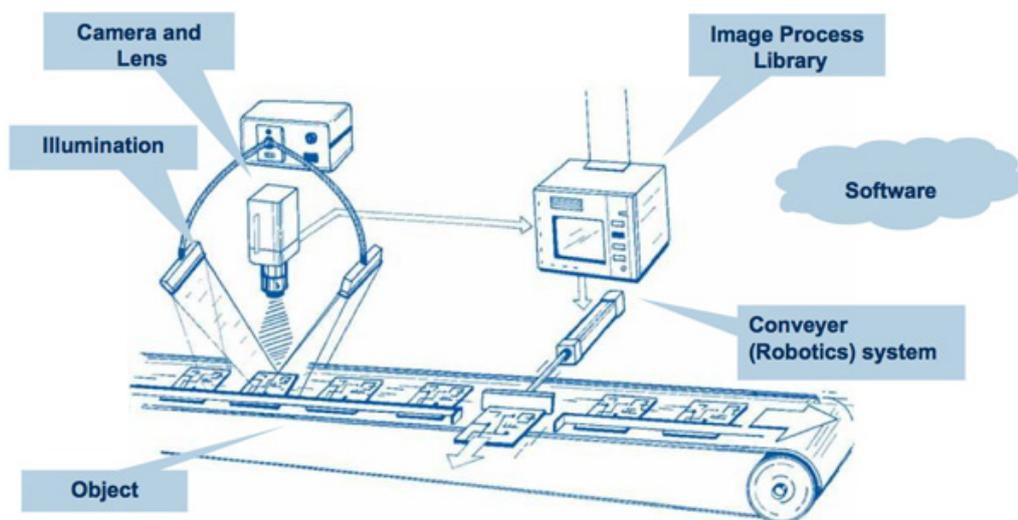
### **2.1.8 Drawbacks of the biometric methods**

Not all biometric methods can be applied for the identification of wild animals. Some of the methods require a close contact with animals. This can be rather difficult to realize for wild animals and can be more suitable for livestock or zoo populations. On the other hand, for the task of wild animals identification methods based on fur patterns and fin shapes are the most suitable approaches because of their invasiveness. Next paragraph shows how novel computer vision technologies can help to make the animal identification easier.

## 2.2 Computer vision methods

Computer vision (CV) is a field that includes methods for acquiring, processing, analyzing, and understanding images and, in general, high-dimensional data from the real world in order to produce numerical or symbolic information. As a scientific discipline, computer vision is concerned with the theory behind artificial systems that extract information from images. The image data can take many forms, such as video sequences, views from multiple cameras, or multi-dimensional data from a medical scanner. As a technological discipline, computer vision seeks to apply its theories and models to the construction of computer vision systems [19].

The organization of a computer vision system is highly application dependent. Some systems are stand-alone applications which solve a specific measurement or detection problem, while others constitute a sub-system of a larger design which, for example, also contains sub-systems for control of mechanical actuators, planning, information databases, man-machine interfaces, etc. The specific implementation of a computer vision system also depends on if its functionality is pre-specified or if some part of it can be learned or modified during operation. Many functions are unique to the application. There are, however, typical functions which are found in many computer vision systems [20]. The typical example of a computer vision system is presented in Figure 7.



**Figure 7.** An example of a real computer vision system. [21]

Firstly, image acquisition is producing of an image by one or several image sensors, which, besides various types of light-sensitive cameras, include range sensors, tomography devices, radar, ultra-sonic cameras, etc. Depending on the type of sensor, the resulting

image data is an ordinary 2D image, a 3D volume, or an image sequence. Secondly, pre-processing is a preliminary step of computer vision system which processes the data in order to assure that it satisfies certain assumptions implied by further CV methods. Next significant step of CV process is the segmentation of an image. Now the decision is made about which image points or regions of the image are relevant for further processing. For example it can be: selection of a specific set of interest points, or segmentation of one or multiple image regions which contain a specific object of interest. Further, high-level processing can be applied to segmented regions of an image. At this step the input is typically a small set of data, for example a set of points or an image region which is assumed to contain a specific object. The remaining processing deals with, for example: verification that the data satisfy model-based and application specific assumptions, estimation of application specific parameters, such as object pose or object size, image recognition - classifying a detected object into different categories, and image registration - comparing and combining two different views of the same object. Feature extraction is one of the main steps in any CV system. At this stage image features at various levels of complexity are extracted from the image data. Typical examples of such features are lines, edges, localized interest points such as corners, blobs, or points. More complex features may be related to texture, shape or motion. Finally, decision making step ends the CV procedure. At last step the final decision is made for the application, for example: pass (fail) on automatic inspection applications, or match (no-match) in recognition applications [20].

The manual visual analysis of animal photos is a time-consuming and laborious task, where an expert needs to compare each new image with a growing database. For example, if this database contains hundreds of thousands photos it is a daunting task [22]. Furthermore, this work requires qualified specialists and often provides identification errors caused by human factor. Therefore, many researchers are working to automate the process of animal identification, using computer vision techniques.

Automatic image-based animal identification methods have been successfully used in many studies. Automatic identification methods have been used for the wide variety of species, including polar bears [23], cattle [24], newts [25], giraffes [26], salamanders [27], snakes [28] as well as certain insects [29] and plants [30]. All of these studies use image processing and pattern recognition techniques in the task of individual identification.

The most of works explores identification of certain animal species or breed. For example, Halloran *et al.* [26] examined effectiveness of wild-ID [26] software in identifying individual Thornicroft's giraffes from a dataset of 552 photographs. This program uses a Scale Invariant Feature Transform (SIFT) algorithm [31] to extract and match distinctive

image features regardless of scale and orientation. Example points of commonality used for comparison are presented in Figure 8.



**Figure 8.** Two giraffe images confirmed as a match via visual analysis. The three points of commonality used are circled in white. [26]

In [25], Hoque *et al.* investigated the suitability of biometric techniques for the identification of the great crested newt. Distinctive belly patterns were used to compare images of newts with image-database. Two different methods were used for the comparison: 1) the correlation coefficient (CC) of the gray-scale pixel intensities, and 2) the Hamming distance (HD) between the binary image segments. The process of feature extraction in the newts identification algorithm is shown in Figure 9.

In [30] image analysis algorithms were applied to the identification of plant species. The proposed system, called Leafsnap identifies tree species from images of their leaves. This system relies on computer vision for several key aspects, including classifying images as leaves or not, obtaining fine-scale segmentations of leaves from their backgrounds, efficiently extracting histograms of curvatures along the contour of the leaf at multiple scales, and retrieving the most similar species matches using a k-nearest neighbors (KNN) search on a large dataset of labeled images. The great interest of people to this application shows the potential of computer vision identification of animals and plants to be implemented



**Figure 9.** Feature extraction from belly patterns in the task of newt-identification. [25]

for a wide range of users.

There has been also research efforts in order to create an unified approach applicable for identification purposes for several animal species. For example, Crall *et al.* [32] presented HotSpotter, a fast, accurate algorithm, for identifying individual animals in a labeled database. This algorithm is not species specific and has been applied to Grevy's and plains zebras, giraffes, leopards, and lionfish. HotSpotter uses viewpoint invariant descriptors and a scoring mechanism that emphasizes the most distinctiveness keypoints and descriptors. In addition, Xiaoyuan Yu *et al.* [33] developed species recognition algorithm based on sparse coding spatial pyramid matching (ScSPM). It was shown that the proposed object recognition techniques can be successfully used to identify wild animals on sequences of images taken by camera traps in nature.

### 2.3 Saimaa ringed seals

To provide scientific data to support conservation measures, free-ranging Saimaa ringed seals in the central Lake Saimaa have been studied using various tracking techniques such as audiovisual monitoring, very-high-frequency (VHF) transmitters and GPS telemetry in [34]. However, there is a need for a long-time span non-invasive method what can be used to monitor the whole population.

Wildlife photo identification is a new way to explore the seal population and contains a great potential because it has invasive essence, may affect a large number of individuals at once, and can be easily extended by installing additional cameras. It can be said that Photo-ID methods bring biologists closer to the seals and they get information about the

changes in the population and also about the individual's fitness, survival, affinities and social behavior [35]. Nowadays, biologists from UEF set cameras around seals' places of habitat and take photos of each individual or group of seals. After collecting images, scientists need to identify seals on the image manually relying on their experience. In spite of the fact that population of Saimaa ringed seals is about 310 individuals, it takes a lot of time to compare each new seal image with a large image-database.

The main objective of this master thesis is to construct a Saimaa seal identification algorithm using an existing image dataset provided by biologists from UEF. This technique should include two principal, sequential steps of image processing: segmentation and identification. First segmentation should separate a seal in the image from surrounding background and to provide seal as image segments for further identification. The purpose of identification step is to extract features from the detected seal and to find an appropriate class (seal) based on feature database. The main steps of proposed algorithm are presented in Figure 2.

## 3 SEGMENTATION

### 3.1 Image segmentation

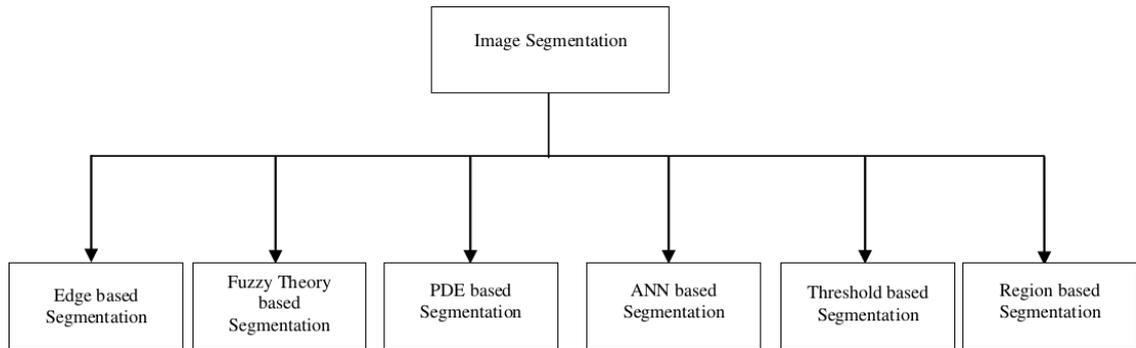
Image segmentation is one of the main steps of image processing. It usually provides the basis for further processing. In the segmentation process, an image is divided into a plurality of segments (sets of pixels, superpixels) that contain some kind of information such as color, intensity, or texture [36]. These characteristics are similar for the pixels lying inside the same segment and different for pixels from other segments. Hence, the goal of segmentation is to simplify and/or to change the representation of an image into something that is more meaningful and easier to analyze [37]. For instance, image segmentation could be performed by assigning a label to every pixel in an image such that pixels with the same label share certain characteristics. The result of image segmentation is a set of segments that collectively cover the entire image, or a set of contours extracted from the image. Image segmentation is typically used to locate objects and boundaries (lines, curves, etc.) in images [38]. Image segmentation plays an important role in various computer vision problems. The specific tasks which can be solved by segmentation include image noise removal, image retrieval, feature extraction, and object recognition [36].

It has been shown that there is not a single best method for image segmentation, due to the fact that each image has its own specific type. Since a method applied to one image may not remain successful to other type of images segmentation techniques have been divided into three types: 1) classical methods, 2) Artificial Intelligence (AI) techniques, and 3) hybrid techniques [39].

The most famous image segmentation methodologies including edge based segmentation [40], fuzzy theory based segmentation [41], partial differential equation (PDE) based segmentation [42], artificial neural network (ANN) based segmentation [43], threshold based image segmentation [44], and region based image segmentation [45] are shown in Figure 10.

The image segmentation methods can be divided into two general types: supervised and unsupervised methods. Supervised segmentation algorithms use a priori knowledge involving a manually labeled training set of images, while unsupervised algorithms train themselves online during segmentation.

In this thesis the goal is to create a supervised image segmentation algorithm for Saimaa



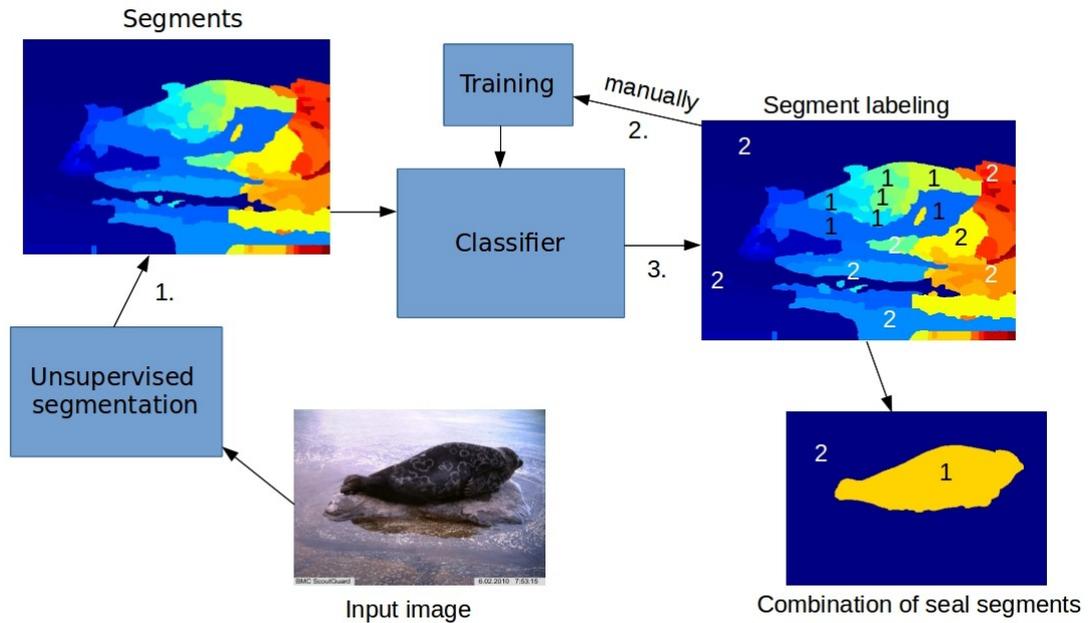
**Figure 10.** Various image segmentation techniques. [36]

ringed seals based on information collected manually with pre-selected images. However, the initial stage of this algorithm includes an unsupervised segmentation part produces a set of segments from the image that can be further classified and combined in supervised manner.

### 3.2 Proposed segmentation algorithm

Segmentation is the first image processing step of the proposed identification system. Figure 11 illustrates the parts of the proposed segmentation algorithm and their relation to each other. It can be seen that segmentation algorithm consists of three main parts: 1) unsupervised segmentation, 2) training of the classifier, and 3) classification of the segments. In the first step, the image is divided into many small segments in unsupervised manner. Each segment is assumed be a seal segment or a background segment. As it was explained in Section 3.1 methods used for this purpose are called unsupervised segmentation since that they do not need ground truth information. In the second stage, a classifier is trained using features from manually labeled segments to classify segments to seal or background. In the third step, the trained classifier is used to automatically classify segments in new test images, and to give as an output only segments containing parts of the detected seal. Ultimately, the segmentation algorithm provides a segmented image of the seal, suitable for identification.

Each segmentation step requires separate testing and selection of the parameters to receive the best result. The following subsections describe the steps in more details, and Section 5 presents the results for a selection of optimal parameters.



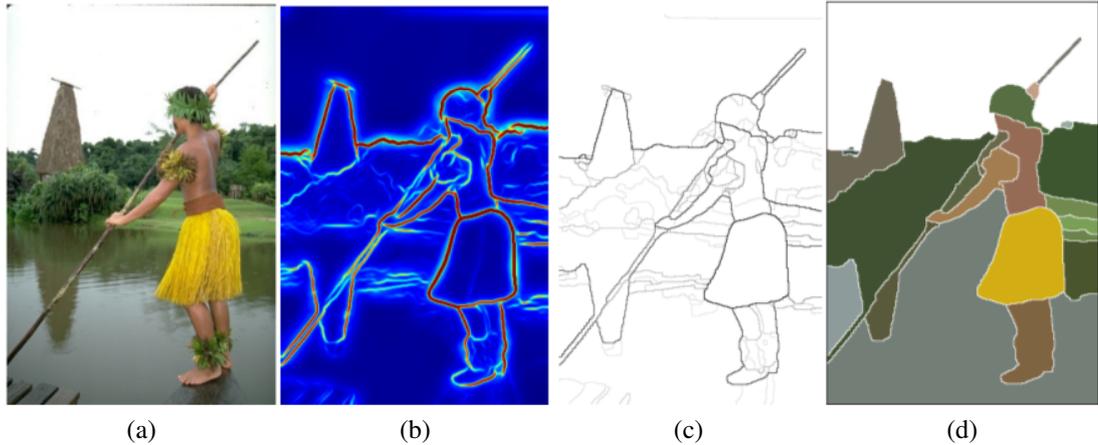
**Figure 11.** Segmentation algorithm: 1) unsupervised segmentation, 2) training, 3) classification.

### 3.3 Unsupervised segmentation

The first step of the proposed seal segmentation algorithm is unsupervised segmentation which involves dividing the image into small segments for further classification. For this purpose the Globalized Probability of Boundary – Oriented Watershed Transform – Ultrametric Contour Map (*gPb-owt-ucm*) algorithm [46, 47] was chosen which has been shown to provide the highest segmentation accuracy with the Berkeley Segmentation Benchmark database [48]. An example of the segmentation results obtained by the algorithm is shown in Figure 12. The *gPb-owt-ucm* algorithm consists of three parts: Globalized Probability of Boundary (gPb) part from [49], Oriented Watershed Transform (OWT) [47], and Ultrametric Contour Map (UCM) [47]. First, the gPb contour detector is applied to produce a probability map  $E(x, y, \theta)$  which describes the probability of an image boundary at location  $(x, y)$  and orientation  $\theta$ . Then, hierarchical regions are built by exploiting the information in the contour probabilities using a sequence of two transformations, OWT and UCM, described below [47].

#### Globalized probability of boundary contour detector

The OWT-UCM part can use any source of contours before thresholding the input  $E(x, y, \theta)$  signal. However, as it was shown in [47] the best choice is to use the *gPb* detector. The



**Figure 12.** Example of the *gPb-owt-ucm* algorithm: (a) original image; (b) maximal response of the contour detector *gPb* over orientations; (c) weighted contours resulting from the Oriented Watershed Transform - Ultrametric Contour Map (OWT-UCM) algorithm using *gPb* as input; (d) segmentation obtained by thresholding the UCM at level 0.4, with segments represented by their mean color. [47]

*gPb* detector combines multiscale local brightness, color, and texture gradients with an oriented spectral signal computed from these cues. The local cues, computed by applying oriented gradient operators at every location in the image, define an affinity matrix representing the similarity between pixels. From this matrix, a generalized eigenproblem is derived and solved for a fixed number of eigenvectors which encode contour information (Figure 12 (b)). Using a classifier to recombine this signal with the local cues, a large improvement is obtained over alternative globalization schemes built on top of similar cues [46].

### Oriented Watershed Transform

Using the contour signal the finest partition for the hierarchy is constructed. This can be understood as an over-segmentation where the segments determine the highest level of detail considered. This is done by computing, first,

$$E(x, y) = \max_{\theta} E(x, y, \theta), \quad (1)$$

i.e., as the maximal response of the contour detector over orientations. After this the regional minima of  $E(x, y)$  are selected as seed locations for homogeneous segments and the watershed transform is applied on the topographic surface defined by  $E(x, y)$ . The catchment basins of the minima, denoted by  $P_0$ , provide the regions of the finest partition and the corresponding watershed arcs,  $K_0$ , the possible locations of the boundaries [47].

In the next step, the strength of the boundaries defined by  $E(x, y, \theta)$  is transferred to the locations  $K_0$ . For this purpose, the watershed arcs with line segments are approximated, and each point in  $K_0$  is weighted by the  $E(x, y, \theta)$  value at the point in the direction  $\theta$  given by the orientation of the corresponding line segment. This procedure called Oriented Watershed Transform (OWT) enforces consistency between the strength of the boundaries of  $K_0$  and the underlying  $E(x, y, \theta)$  signal and removes artifacts of the standard watershed algorithm [47].

### Ultrametric Contour Map

One possibility to represent uncertainty of the segmentation is Ultrametric Contour Map (UCM) [50] which defines a duality between closed, nonself-intersecting weighted contours and a hierarchy of regions. Making this shift representation from a single segmentation to a nested collection of segmentations has shown to be very powerful [47].

The hierarchy is constructed by a greedy graph-based region merging algorithm. An initial graph is defined where the nodes are the regions in  $P_0$  the links join adjacent regions and are weighted by a measure of similarity between regions. The algorithm proceeds by sorting the links by similarity and iteratively merging the most similar regions. This process produces a tree of regions where the leaves are the elements of  $P_0$ , the root is the entire image domain, and the regions are ordered by the inclusion relation [47].

The similarity between two adjacent regions is defined as the average strength of their common boundary in  $K_0$ , initialized by OWT. Since this value cannot decrease during the merging process, there is guaranteed to produce an ultrametric distance on  $P_0 \times P_0$  [50]. As a consequence, the constructed region tree has the structure of an indexed hierarchy and can be described by a dendrogram where the height of each segment is the value of the similarity at which it first appears and the distance between two regions is the height of the smallest segment in the hierarchy containing them. Furthermore, the whole hierarchy can be represented as an UCM, that is the real-valued image obtained by weighting each boundary between two regions by its scale of disappearance [47].

Figure 12 shows an example of the *gPb-owt-ucm* method. The UCM is a weighted contour image that, by construction, has the remarkable property of producing a set of closed curves for any threshold. Conversely, it is a convenient representation of the region tree since the segmentation at a scale  $k$  can be easily retrieved by thresholding the UCM at level  $k$ . Since a notion of scale is the average contour strength, the UCM values reflect the contrast between neighboring regions [47].

Depending on the threshold value segments size can be changed. Therefore, it was necessary to choose the most appropriate threshold providing the lowest segmentation error. In this regard, *gPb-owt-ucm* algorithm was tested with different values of the threshold. The test results are presented in Section 5.

### 3.4 Feature extraction and selection

The next stage of the seal segmentation algorithm is to find features describing the image segments containing parts of seal or background. These features are needed to train the classifier (Figure 11, Step 2) and to classify segments in new images (Figure 11, Step 3). The main purpose of the segment description and labeling is to create an automatic supervised segmentation system which allows to detect image segments containing parts of seal and combine them into a one large segment that contains all the pixels that belong to the seal.

To describe the segments several features were considered including mean RGB colors, center distance, area, Segmentation-based Fractal Texture Analysis (SFTA) descriptor [51], Local Binary Pattern Histogram Fourier (LBP-HF) descriptor [52], Local Phase Quantization (LPQ) descriptor [53]. Table 1 shows all features used with their short description and the length of the feature vector.

**Table 1.** Features description

Feature	Description	Length
Mean colors (3)	Mean intensity of R, G, B color channels	3
Center distance	Distance between a center of the segment and a center of an image	1
Area	Area of the segment in pixels	1
SFTA	Fractal dimension, mean gray-level and size of several binary components	48
LBP-HF	Discrete Fourier transforms of local binary pattern (LBP) histograms	43
LPQ	Quantized phase of the discrete Fourier transform (DFT) computed in local image windows	256

Simple features such as average intensity of the RGB layers, the distance between the centroid of the segment and the image center, and area of the segment were selected due to the assumption that the these features describes efficiently the image segments containing parts of seals. Skin of the seals in most cases contains the similar color different from

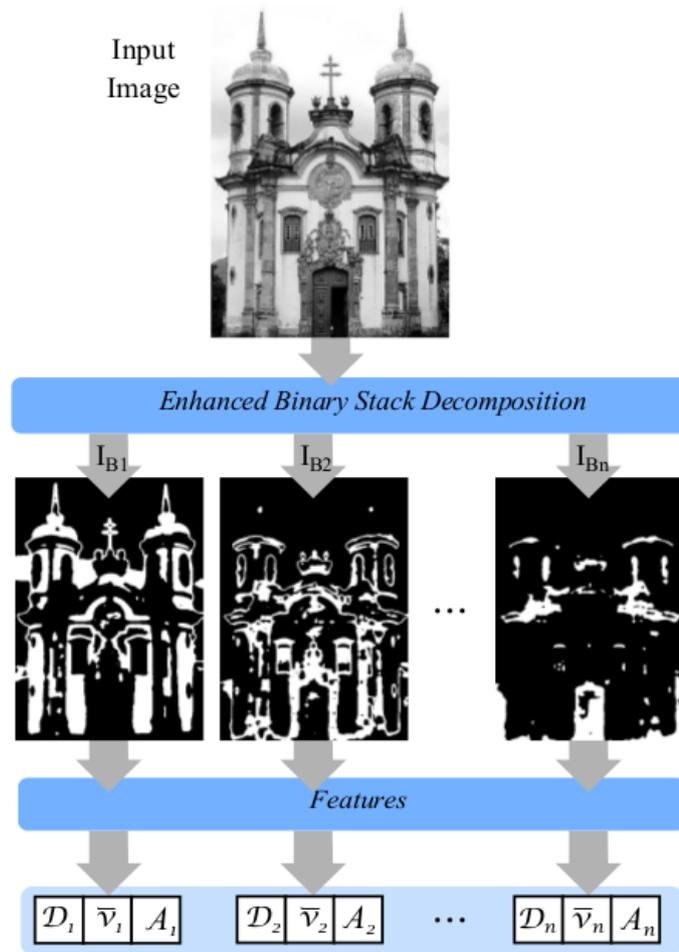
the background. Secondly, almost all pictures containing seal segments are closer to the center of the image. Thirdly, often seal segment has the largest size among the other segments. However, in spite of this hypothesis these simple features do not provide sufficient accuracy in the classification process of segments. Therefore, three independent sets of features were chosen for further study.

### 3.4.1 SFTA

Segmentation-based Fractal Texture Analysis (SFTA) is an efficient texture feature extraction method proposed in [51]. It consists of two steps: 1) decomposing of the input image into a set of binary images and 2) computation of fractal dimension from its regions boundaries for each resulting binary image. Additionally, the mean gray level and size (pixel counting) of the regions are used as features.

To decompose the input image a technique called Two-Threshold Binary Decomposition (TTBD) is employed which takes a grayscale image  $I(x, y)$  as an input and returns a set of binary images. The first step of TTBD consists of computing a set  $T$  of threshold values, where TTBD adopts a strategy that uses the input image gray level distribution information to compute the set of thresholds [51]. This is achieved by employing the multi-level Otsu algorithm [54].

After applying the TTBD to the input gray level image, the SFTA feature vector is constructed as the resulting binary image sizes, mean gray level, and fractal dimension of the boundaries. The fractal measurements are employed to describe the boundary complexity of objects and structures segmented in the input image. Thus, the SFTA feature vector dimensionality corresponds to the number of binary images obtained by TTBD multiplied by three, since the following measurements are computed from each binary image: fractal dimension, mean gray level and size [51]. Figure 13 illustrates the SFTA extraction algorithm. First the input image is decomposed into a set of binary image by the TTBD algorithm. Then, the SFTA feature vector is constructed as the resulting binary image: sizes, mean gray level, and fractal dimension of the boundaries.



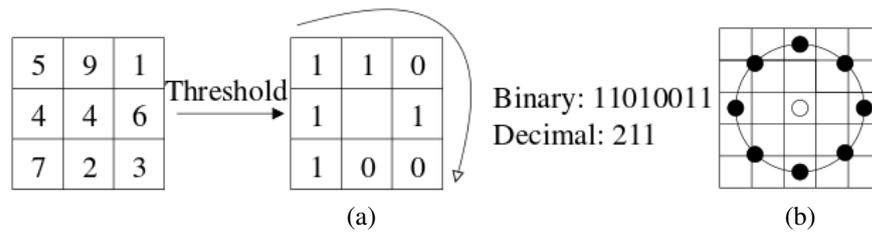
**Figure 13.** SFTA extraction algorithm diagram taking as input a grayscale image. [51]

### 3.4.2 LBP-HF

The original Local Binary Pattern (LBP) operator, introduced in [55], is a powerful means of texture description. The operator labels the pixels of an image by thresholding the  $3 \times 3$  neighbourhood of each pixel with the center value and considering the result as a binary number. Then the histogram of the labels can be used as a texture descriptor [56]. Figure 14(a) illustrates the basic LBP operator.

Later the operator was extended to use neighbourhoods of different sizes [57]. Using circular neighbourhoods and bilinearly interpolating the pixel values allow any radius and number of pixels in the neighbourhood [56]. Figure 14(b) demonstrates an example of the circular (8,2) neighbourhood.

Local Binary Pattern Histogram Fourier feature (LBP-HF) [52] is a novel rotation in-



**Figure 14.** LBP: (a) the basic LBP operator; (b) the circular (8,2) neighbourhood. [56]

variant image descriptor computed from the discrete Fourier transforms of local binary pattern (LBP) histograms. Unlike most other histogram based invariant texture descriptors which normalize rotation locally, LBP-HF invariants are constructed globally for the whole region to be described. In addition to being rotation invariant, the LBP-HF features retain the highly discriminative nature of LBP histograms.

It has been shown that rotations of the input image cause cyclic shifts of the values in the uniform LBP histogram [52]. Relying on this observation, discrete Fourier transform based features were proposed that are invariant to cyclic shifts of the input vector and, when computed from uniform LBP histograms, invariant to rotations of the input image.

LBP-HF features are computed from the histogram representing the whole region, i.e. the invariants are constructed globally instead of computing invariant independently at each pixel location. The major advantage of this approach is that the relative distribution of local orientations is not lost [52].

Another benefit of constructing invariant features globally is that invariant computation needs not to be performed at every pixel location. This allows using computationally more complex invariant functions still keeping the total computational cost reasonable. In the case of LBP-HF descriptor, the computational overhead is negligible. After computing the non-invariant LBP histogram, only  $P - 1$  Fast Fourier Transforms of  $P$  points need to be computed to construct the rotation invariant LBP-HF descriptor [52].

### 3.4.3 LPQ

Local Phase Quantization (LPQ) is a blur insensitive texture classification method, which is based on quantized phase of the discrete Fourier transform (DFT) computed in local image windows [53]. The codes produced by the LPQ operator are insensitive to centrally symmetric blur, which includes motion, out of focus, and atmospheric turbulence blur.

The LPQ operator is applied to texture identification by computing it locally at every pixel location and presenting the resulting codes as a histogram. Generation of the codes and their histograms is similar to the LBP method [57].

The LPQ method is based on the blur invariance property of the Fourier phase spectrum. It uses the local phase information extracted using the 2-D DFT or, more precisely, a short-term Fourier transform (STFT) computed over a rectangular  $M - by - M$  neighborhood  $N_x$  at each pixel position  $x$  of the image  $f(x)$  and defined as

$$F(u, x) = \sum_{y \in N_x} f(x - y) e^{-j2\pi u^T y} = w_u^T f_x \quad (2)$$

where  $w_u$  is the basis vector of the 2-D DFT at frequency  $u$ , and  $f_x$  is a vector containing all  $M_2$  image samples from  $N_x$  [53].

The phases of the four low-frequency coefficients are uniformly quantized into one of 256 hypercubes in eight-dimensional space, which results in an 8-bit code. These LPQ codes for all image pixel neighborhoods are collected into a histogram, which describes the texture and can be used for classification. The phases of the low-frequency components are shown to be ideally invariant to centrally symmetric blur. Although, the invariance is disturbed by the finite-sized image windows, the method is still very tolerant of blur. Because only phase information is used, the method is also invariant to uniform illumination changes [53].

#### 3.4.4 Feature selection

All of the above features were tested for evidence of providing the most accurate classification of segments and image segmentation in general. The results of experiments using different features and classifiers are shown in Section 5.

### 3.5 Segment classification

An important point in the development of seal image segmentation algorithm is the choice of a suitable classifier. For the classification of image segments obtained in unsupervised segmentation step and features described by texture, three widely known classifiers were

chosen: Naive Bayes Classifier (NB), k-nearest neighbor classifier (KNN) and support vector machine classifier (SVM). Descriptions of these classifiers are given in the following subsections.

### 3.5.1 Naive Bayes classifier

A Naive Bayes classifier is a simple probabilistic classifier based on applying Bayes' theorem with naive independence assumptions. In simple terms, a naive Bayes classifier assumes that the presence (or absence) of a particular feature of a class is unrelated to the presence (or absence) of any other feature [58].

Let  $x$  is a vector to classify, and  $c_k$  is a possible class. Finding the probability  $P(c_k|x)$  that the vector  $x$  belongs to the class  $c_k$ , NB classifier algorithm can be described as follows [59]:

1. Compute the probability  $P(c_k|x)$  using Bayes' rule,

$$P(c_k|x) = P(c_k) \frac{P(x|c_k)}{P(x)}, \quad (3)$$

where  $P(c_k)$  is a probability of occurrence of class  $c_k$ ,  $P(x|c_k)$  is a probability of generating instance  $x$  given class  $c_k$ ,  $P(x)$  is a probability of instance  $x$  occurring.

2. Class probability  $P(c_k)$  can be estimated from training data. However, direct estimation of  $P(c_k|x)$  is impossible in most cases because of the sparseness of training data.
3. By assuming the conditional independence of the elements of a vector,  $P(x|c_k)$  is decomposed as follows,

$$P(x|c_k) = \prod_{j=1}^d P(x_j|c_k), \quad (4)$$

where  $x_j$  is the  $j$ th element of vector  $x$  and  $P(x_j|c_k)$  is the probability of generating instance  $x_j$  given class  $c_k$ .

4. Then Equation 3 becomes

$$P(c_k|x) = P(c_k) \frac{\prod_{j=1}^d P(x_j|c_k)}{P(x)}. \quad (5)$$

With this equation,  $P(c_k|x)$  is calculated.

5. Finally,  $x$  is classified into the class with the highest  $P(c_k|x)$ .

Depending on the precise nature of the probability model, naive Bayes classifiers can be trained very efficiently in a supervised learning setting. In many practical applications, parameter estimation for naive Bayes models uses the method of maximum likelihood. In spite of their naive design and apparently over-simplified assumptions, naive Bayes classifiers have worked quite well in many complex real-world situations. Analysis of the Bayesian classification problem has shown that there are theoretical reasons for the apparently unreasonable efficacy of naive Bayes classifiers [58].

The main advantage of the naive Bayes classifier is that it only requires a small amount of training data to estimate the parameters (means and variances of the variables) necessary for classification. Because independent variables are assumed, only the variances of the variables for each class need to be determined and not the entire covariance matrix [58].

### 3.5.2 KNN classifier

The k-Nearest Neighbors (KNN) algorithm, like other instance-based algorithms, is unusual from a classification perspective in its lack of explicit model training. While a training dataset is required, it is used solely to populate a sample of the search space with instances whose class is known. No actual model or learning is performed during this phase; for this reason, these algorithms are also known as lazy learning algorithms. When an instance whose class is unknown is presented for evaluation, the algorithm computes its  $k$  closest neighbors, and the class is assigned by voting among those neighbors. Different distance metrics can be used, depending on the nature of the data. Euclidean distance is typical for continuous variables, but other metrics can be used for categorical data. Specialized metrics are often useful for specific problems, such as text classification. To prevent ties, one typically uses an odd choice of  $k$  for binary classification. For multiple classes, one can use plurality voting or majority voting. The latter can sometimes result in no class being assigned to an instance, while the former can result in classifications being made with very low support from the neighborhood. One can also weight each neighbor by an inverse function of its distance to the instance being classified [60].

The training phase for KNN consists of simply storing all known instances and their class labels. A tabular representation can be used, or a specialized structure such as a  $k$ -dimensional ( $k$ -d) tree. The  $k$ -d tree is a binary tree in which every node is a dimensional point. To tune the value of  $k$  and perform feature selection,  $n$ -fold cross-validation can be

used on the training dataset. The testing phase for a new instance  $t$ , given a known set  $I$  is as follows [60]:

1. Compute the distance between  $t$  and each instance in  $I$ .
2. Sort the distances in increasing numerical order and pick the first  $k$  elements.
3. Compute and return the most frequent class in the  $k$  nearest neighbors, optionally weighting each instance's class by the inverse of its distance to  $t$ .

### 3.5.3 SVM classifier

Support Vector Machines (SVMs) are supervised learning methods used for classification and regression tasks that originated from statistical learning theory [61]. As a classification method, SVM is a global classification model that generates non-overlapping partitions and usually employs all attributes. The entity space is partitioned in a single pass, so that flat and linear partitions are generated. SVMs are based on maximum margin linear discriminants, and are similar to probabilistic approaches, but do not consider the dependencies among attributes [62]. SVMs rely on preprocessing the data to represent patterns in a high dimension, typically much higher than the original feature space [63]. Data from two categories can always be separated by a hyperplane when an appropriate nonlinear mapping to a sufficiently high dimension is used.

Let  $D$  be a classification dataset with  $n$  points in a  $d$ -dimensional space  $D = \{(x_i, y_i)\}$ , with  $i = 1, 2, \dots, n$  and let there be only two class labels such that  $y_i$  is either  $+1$  or  $-1$ . A hyperplane  $h(x)$  gives a linear discriminant function in  $d$  dimensions and splits the original space into two half-spaces:

$$h(x) = \omega^T x + b = \omega_1 x_1 + \omega_2 x_2 + \dots + \omega_d x_d + b \quad (6)$$

where  $\omega$  is a  $d$ -dimensional weight vector and  $b$  is a scalar bias. Points on the hyperplane have  $h(x) = 0$ , i.e. the hyperplane is defined by all points for which  $\omega^T x = -b$ .

According to [61], if the dataset is linearly separable, a separating hyperplane can be found such that for all points with label  $-1$ ,  $h(x) < 0$  and for all points labeled  $+1$ ,  $h(x) > 0$ . In this case,  $h(x)$  serves as a linear classifier or linear discriminant that predicts the class for any point. Moreover, the weight vector  $\omega$  is orthogonal to the hyperplane,

therefore giving the direction that is normal to it, whereas the bias  $b$  fixes the offset of the hyperplane in the  $d$ -dimensional space.

Given a separating hyperplane  $h(x) = 0$ , it is possible to calculate the distance between each point  $x_i$  and the hyperplane by:

$$\delta_i = \frac{y_i h(x_i)}{\|\omega\|} \quad (7)$$

The margin of the linear classifier is defined as the minimum distance of all  $n$  points to the separating hyperplane.

$$\delta^* = \min_{x_i} \left\{ \frac{y_i h(x_i)}{\|\omega\|} \right\} \quad (8)$$

All points that achieve this minimum distance are called the support vectors for the linear classifier. In other words, a support vector is a point that lies precisely on the margin of the classifying hyperplane.

In a canonical representation of the hyperplane, for each support vector  $x_i^*$  with label  $y_i^*$  there is  $y_i^* h(x_i^*) = 1$ . Similarly, for any point that is not a support vector, there is  $y_i h(x_i) > 1$ , since, by definition, it must be farther from the hyperplane than a support vector. Therefore we have that  $y_i h(x_i) \geq 1, \forall x_i \in D$ .

The fundamental idea behind SVMs is to choose the hyperplane with the maximum margin, i.e. the optimal canonical hyperplane. To do this, one needs to find the weight vector  $\omega$  and the bias  $b$  that yield the maximum margin among all possible separating hyperplanes, that is, the hyperplane that maximizes  $\frac{1}{\|\omega\|}$ .

SVMs can also solve problems with non-linear decision boundaries. The main idea is to map the original  $d$ -dimensional space into a  $d'$ -dimensional space ( $d' > d$ ), where the points can be linearly separated. Given the original dataset  $D = x_i, y_i$  with  $i = 1, \dots, n$  and the transformation function  $\Phi$ , a new dataset is obtained in the transformation space  $D_\Phi = \Phi(x_i), y_i$  with  $i = 1, \dots, n$ . After the linear decision surface is found in the  $d'$ -dimensional space, it is mapped back to the non-linear surface in the original  $d$ -dimensional space [61]. To obtain  $\omega$  and  $b$ ,  $\Phi(x)$  need not be computed in isolation. The only operation required in the transformed space is the inner product  $\Phi(x_i)^T \Phi(x_j)$ , which is defined with the kernel function  $K$  between  $x_i$  and  $x_j$ . Kernels commonly used with

SVMs include:

- the polynomial kernel  $K(x_i, x_j) = (x_i^T x_j + 1)^q$ , where  $q$  is the degree of the polynomial,
- the gaussian kernel  $K(x_i, x_j) = e^{-\frac{\|x_i - x_j\|^2}{2\sigma^2}}$ , where  $\sigma$  is the spread or standard deviation,
- the gaussian radial basis function (RBF)  $K(x_i, x_j) = e^{-\gamma\|x_i - x_j\|^2}$ ,  $\gamma \geq 0$ ,

and others.

SVM were initially designed for binary (two-class) problems. When dealing with multiple classes, an appropriate multi-class method is needed. Vapnik [64] suggested comparing one class with the others taken together. This strategy generates  $n$  classifiers, where  $n$  is the number of classes. The final output is the class that corresponds to the SVM with the largest margin, as defined above. For multi-class problems one has to determine  $n$  hyperplanes. Thus, this method requires the solution of  $n$  Quadratic Programming (QP) optimisation problems, each of which separates one class from the remaining classes. This strategy can be described as "one against the rest" [65].

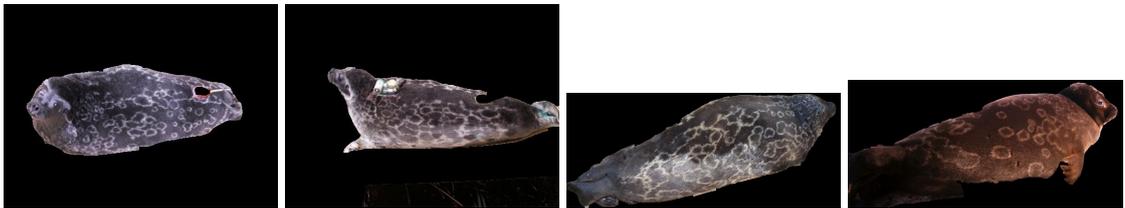
A second approach is to combine several classifiers ("one against one"). Knerr *et al.* [66] perform pair-wise comparisons between all  $n$  classes. Thus, all possible two-class classifiers are evaluated from the training set of  $n$  classes, each classifier being trained on only two out of  $n$  classes, giving a total of  $n(n-1)/2$  classifiers. Applying each classifier to the test data vectors gives one vote to the winning class. The data is assigned the label of the class with most votes [65]. The results of a recent analysis of multi-class strategies are provided by Hsu and Lin [67].

## 4 IDENTIFICATION

Identification is the second stage of the seal recognition algorithm. As mentioned in Section 2 identification is the key component and segmentation is only preparatory process for the identification. This section describes the proposed identification algorithm based on processing of the obtained seal segments.

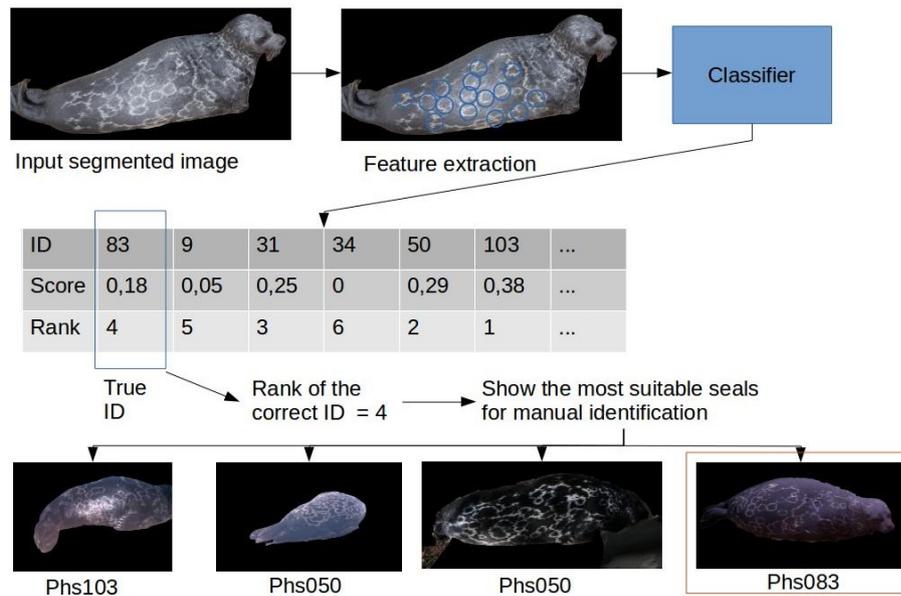
### 4.1 Proposed identification algorithm

In the second part of the work (identification of seals) processed images are used which were obtained after the segmentation process. In the images only segment with a seal is visible, and the background is colored black. Images of this type are convenient to extract features of animal skin for further recognition problem. Examples of segmented seal images are shown in Figure 15.



**Figure 15.** Images of seals for the identification step.

The proposed identification algorithm consists of several steps. First, features characterizing the texture of seal fur are extracted from the segmented image obtained using the automatic supervised segmentation. These features are given to a pre-trained classifier that computes the probabilities of the image (seal) belonging to a particular class (seal individual). As a result, the best matches of seals are found and the final decision can be made by expert using a limited amount of possible seals. Optimally the best match is correct and a specialist is not needed. The Saimaa ringed seal identification algorithm is shown in Figure 16.



**Figure 16.** Identification algorithm. The classifier uses extracted features from segmented images to obtain the most probable class for income image.

As a result of the classifier, a table of possible seal ids is produced. The score row represents probabilities of seal belonging to a certain id. Rank indicates the score of a given seal id by assigning values from one to the number of seals, where  $rank = 1$  is the seal id with the maximum probability. The cumulative match score (CMS) for a certain rank  $R$  is the percentage of seals for which the true seal is within the  $R$  best matches proposed by the algorithm. Evaluation of the identification performance using CMS allows to reduce the number of possible seal ids for the further manual identification by experts. Figure 13 shows an example where the four possible seal ids with ranks from 1 to 4 have the greatest probabilities. In this case the problem is reduced to the selection from the four individuals instead of the initially possible more seals. The following subsections describe the steps of identification in more details and Section 5 presents the results of a selection of optimal parameters.

## 4.2 Feature extraction and selection

For the purposes of the seal identification formulated as a classification task, it is necessary to select the appropriate set of features that characterize the fur covered with the ring pattern. As a features for the identification, the features were found promising in the

segmentation stage were considered, namely the SFTA, LBP, LPQ features.

Due to the fact that certain feature descriptors produce long feature vectors and not all classifiers work equally well with a large set of features the Principal Component Analysis (PCA) was applied to investigate the possibility of its application to reduce the dimension of features. Following subsection is a brief description of the PCA operation. The results of the experiments with PCA dimension reduction are presented in Section 5.

#### 4.2.1 PCA

Principal Component Analysis (PCA) is the general name for a technique which uses sophisticated underlying mathematical principles to transform a number of possibly correlated variables into a smaller number of variables called principal components. The origins of PCA lie in multivariate data analysis, however, it has a wide range of other applications. PCA has been called, "one of the most important results from applied linear algebra" [68] and perhaps its most common use is as the first step in trying to analyse large data sets. Some of the other common applications include: de-noising signals, blind source separation, and data compression.

In general terms, PCA uses a vector space transform to reduce the dimensionality of large data sets. Using mathematical projection, the original data set, which may have involved many variables, can often be interpreted in just a few variables (the principal components). It is therefore often the case that an examination of the reduced dimension data set will allow the user to spot trends, patterns and outliers in the data, far more easily than would have been possible without performing the principal component analysis [69].

The PCA algorithm consists of the following steps [70]:

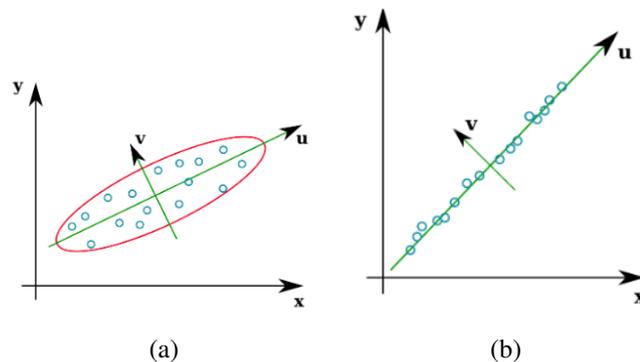
1. Take the whole dataset consisting of  $d$ -dimensional samples ignoring the class labels.
2. Compute the  $d$ -dimensional mean vector (i.e., the means for every dimension of the whole dataset).
3. Compute the scatter matrix (alternatively, the covariance matrix) of the whole data set.
4. Compute eigenvectors and corresponding eigenvalues .

5. Sort the eigenvectors by decreasing eigenvalues and choose  $k$  eigenvectors with the largest eigenvalues to form a  $d \times k$  dimensional matrix  $W$  (where every column represents an eigenvector)
6. Use this  $d \times k$  eigenvector matrix to transform the samples onto the new subspace. This can be summarized by the mathematical equation:

$$y = W^T \times x, \quad (9)$$

where  $x$  is a  $d \times 1$  -dimensional vector representing one sample, and  $y$  is the transformed  $k \times 1$  -dimensional sample in the new subspace.

For example, in Figure 17(a), there is supposed a two variable data set measured in the  $xy$  coordinate system. The principal direction in which the data varies is shown by the  $u$  axis and the second most important direction is the  $v$  axis orthogonal to it. If the  $uv$  axis system is placed at the mean of the data it gives a compact representation. If each  $(x, y)$  coordinates are transformed into its corresponding  $(u, v)$  value, the data is de-correlated, meaning that the covariance between the  $u$  and  $v$  variables is zero. For a given set of data, principal component analysis finds the axis system defined by the principal directions of variance (i.e. the  $uv$  axis system in Figure 17(a)). The directions  $u$  and  $v$  are called the principal components [71].



**Figure 17.** PCA: (a) PCA for Data Representation; (b) PCA for Dimension Reduction. [71]

If the variation in a data set is caused by some natural property, or is caused by random experimental error, then it can be expected to be normally distributed. In this case the nominal extent is shown of the normal distribution by a hyper-ellipse. The hyper ellipse encloses data points that are thought of as belonging to a class. It is drawn at a distance

beyond which the probability of a point belonging to the class is low, and can be thought of as a class boundary [71].

If the variation in the data is caused by some other relationship then PCA gives us a way of reducing the dimensionality of a data set. Consider two variables that are nearly related linearly as shown in Figure 17(b). As in figure Figure 17(a) the principal direction in which the data varies is shown by the  $u$  axis, and the secondary direction by the  $v$  axis. However in this case all the  $v$  coordinates are all very close to zero. It may be assumed, for example, that they are only non zero because of experimental noise. Thus in the  $uv$  axis system we can represent the data set by one variable  $u$  and discard  $v$ . Thus the dimensionality of the problem is reduced by one [71].

### 4.3 Identification

The basis of the seal identification system considered in this section is a classifier which able to calculate a score (e.g. probability) for a seal belonging to a certain class (seal id) determined in the step of the classifier training. The same classifiers that were used in the classification segments were considered. NB classifier is based on the calculation of probability of belonging sample to the class, therefore applying NB to the multiclass task is straightforward. KNN classifier measures distances to  $k$ -nearest neighbours and calculates the score by a majority vote of its neighbors, with the object being assigned to the class most common among its  $k$ -nearest neighbors. Although SVM classifier traditionally is used in two class problems, it can be used in multiclass case as it is described in Section 3.5.3. Then for "one-vs-all" method score of class is calculated in two steps:

1. Train  $c$  classifiers, each classifier  $i$  with the two classes 1)  $c_i$  and 2) all classes except  $c_i$ .
2. Sum-normalize all single-class-scores and find final class score  $S(c_i)$

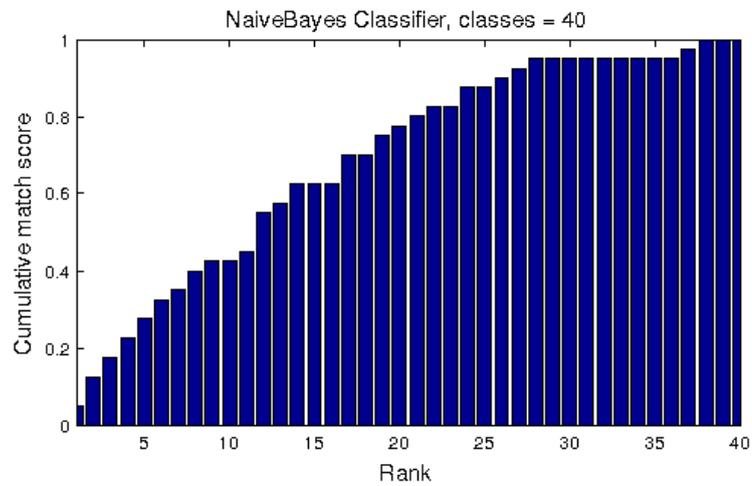
$$S(c_i) = \frac{\text{score}(c_i)}{\sum \text{score}(c_i)} \quad (10)$$

From these probabilities ranks are extracted, and then CMS histogram is formed as follows:

1. For each test image, find the rank of true seal id.

2. For each rank  $R$ , find the percentage of images where the correct seal id is within the set of  $R$  best matches.

The produced CMS histogram represents accumulative accuracy of the proposed identification system. Each bar of the histogram shows a percentage of images which can be successfully identified within the set of  $R$  seal ids. Figure 18 shows an example of CMS histogram.



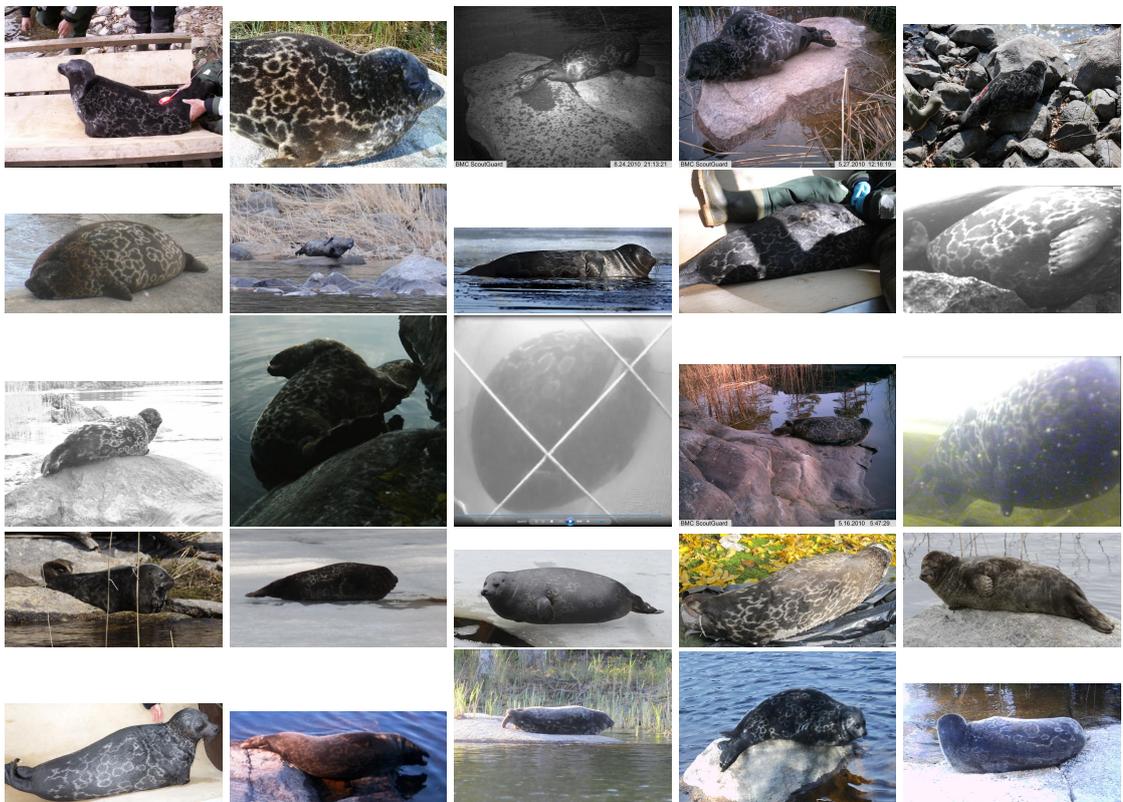
**Figure 18.** Example of the CMS histogram.

## 5 EXPERIMENTS

In this section the used datasets and experiments of segmentation and identification are presented.

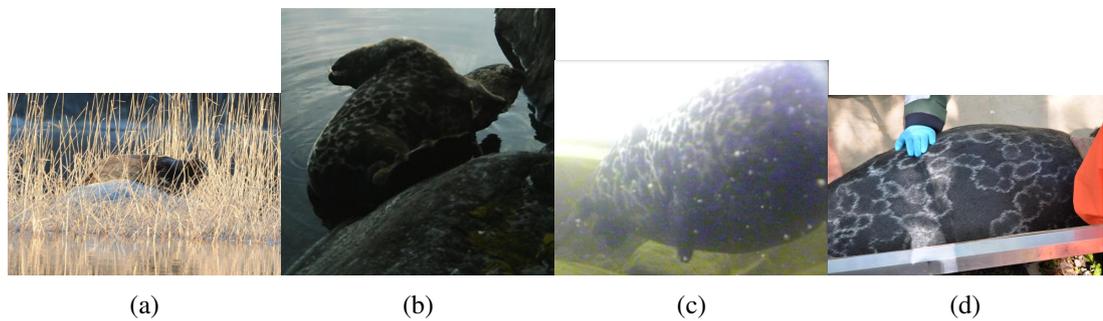
### 5.1 Data

In this research, an unique database of Saimaa ringed seal images collected by UEF was used. This database includes 131 individual seals and contains the total number of 785 images. Most of the images contain one individual Saimaa ringed seal in the nature or biological reserve. However, not all images have the same form and content. Some images contain additional objects, such as two or more individual seals, people working with animals, or a sensor mounted on the body of the animals. Sample images from the database are shown in Figure 19.



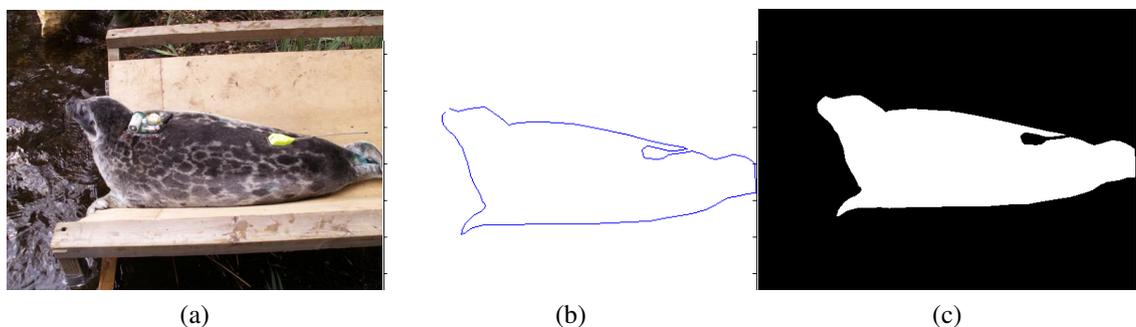
**Figure 19.** Examples from the Saimaa ringed seals database. [4]

Variations in images could make the identification task significantly more difficult or impossible. Examples of such problems are: images with objects overlaying on the top of seals, images with very low contrast, and highly overexposed images. Preliminary experiments with such images showed poor results due to the difficulty of extracting necessary features. Therefore, for further work 363 images of acceptable quality that do not contain the above disadvantages were selected. Examples of possible changes in the quality are shown in Figure 20.



**Figure 20.** Examples of image quality: (a) Covered; (b) Low contrast; (c) High brightness; (d) Good quality.

Next, the segmentation ground truth (GT) for the seal images was formed manually by the author. In these images seals were segmented from background and true contour of seal was formed. In total 121 images were segmented manually to. This information was used to compare segmentation results and calculate the accuracy of the proposed segmentation algorithm. An example of the GT creation is shown in Figure 21.



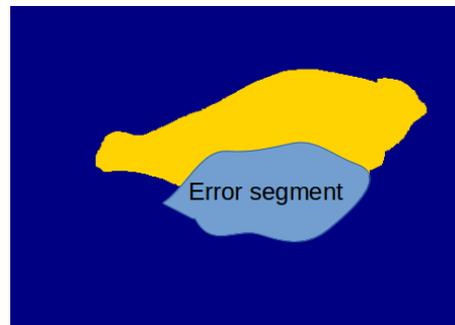
**Figure 21.** Ground truth creation: (a) Image from the database [4]; (b) Manually created contour; (c) Mask after manual annotation.

## 5.2 Segmentation

As it was shown in Section 3 (Figure 11) segmentation process consists of two main steps: unsupervised segmentation and classification of image segments. Experiments were performed for both steps.

### 5.2.1 Threshold finding for unsupervised segmentation

The UCM segmentation algorithm [47] selected for the unsupervised segmentation allows to choose the threshold value that effects on the size of the segments obtained as an output. In this regard, it was necessary to select the optimal value for the threshold. To evaluate the threshold value, two mandatory conditions were defined to make further work successful. The first condition is the minimization of the number of erroneous segments containing both seal and background. An example of an incorrect segment is shown in Figure 22.



**Figure 22.** Erroneous segment containing both a seal and background.

The second condition was to obtain the smallest possible number of segments as the output of the algorithm due to the fact that the choice of the large threshold leads to a very large number of small segments which makes it significantly more difficult to classify the segments.

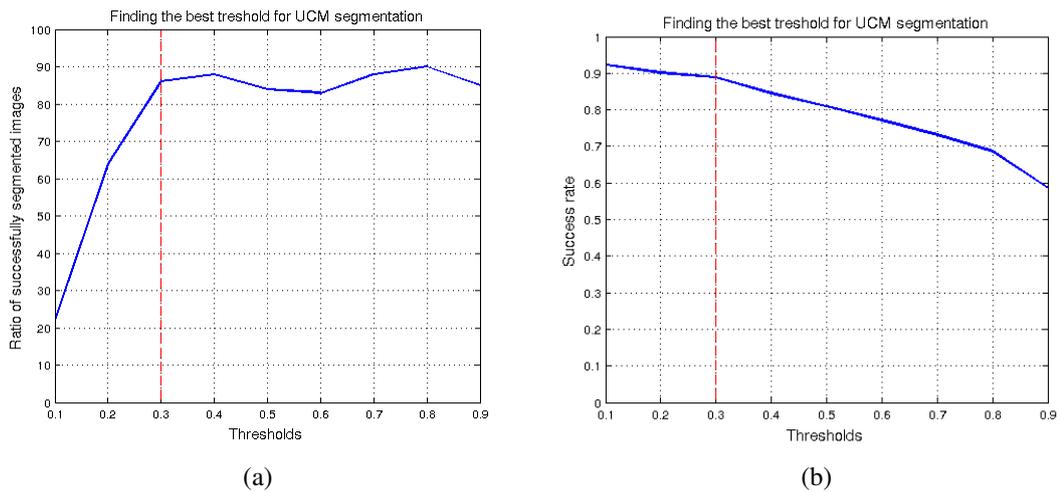
The testing of the UCM algorithm was performed for 121 images with different values of the threshold between 0.1 and 0.9. The number of erroneous segments was computed for every image using GT information formed manually as shown in Figure 21. The results of the experiment are presented in Figure 23. The ratio of successfully segmented images is calculated through following steps:

1. For each segment in all test images define:

*Entrancy* is a measure of a segment embedding in the GT seal segment and it is formed as follows: if all pixels of the segment are inside the GT segmentation,  $entrancy = 1$ ; if all pixels of the segment are outside the GT segmentation,  $entrancy = 0$ ; if the value of *entrancy* is between 0.1 and 0.9 the segment is considered as an erroneous segment.

2. For each threshold value, find the ratio of successfully segmented images as a percentage of images where there are no erroneous segments.

Figure 23(a) shows the ratio of successfully segmented images to the selected threshold value. Figure 23(b) reflects the relation of the success rate to the selected threshold value.



**Figure 23.** Experiments with thresholds: (a) Percentage of successfully segmented images; (b) Success rate.

The success rate is defined as:

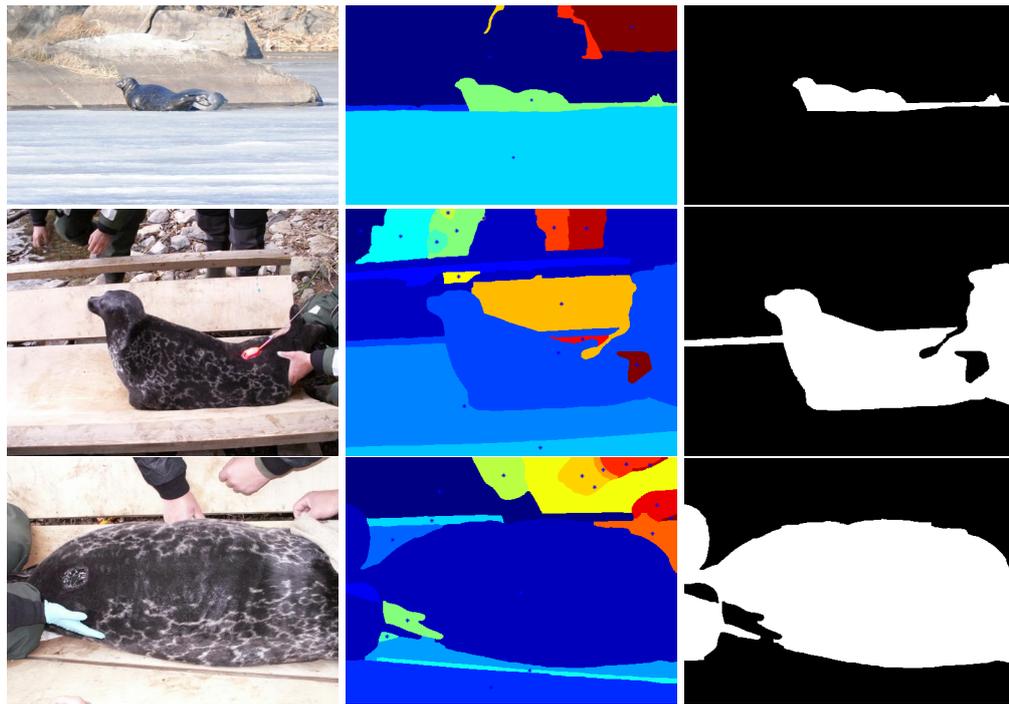
$$\text{Success rate} = \frac{\text{Number of correct segments}}{\text{Total number of segments}}, \quad (11)$$

where the correct segment is a segment with *entrancy* value in the intervals  $0 < 0.1$  and  $0.9 < 1$ .

Figure 23 shows that the optimal threshold value of the UCM algorithm is equal to 0.3 due to the fact that with this value there is a large number of correct segments and at the same time the size of segments do not become too small.

### 5.2.2 Labeling for training of a classifier

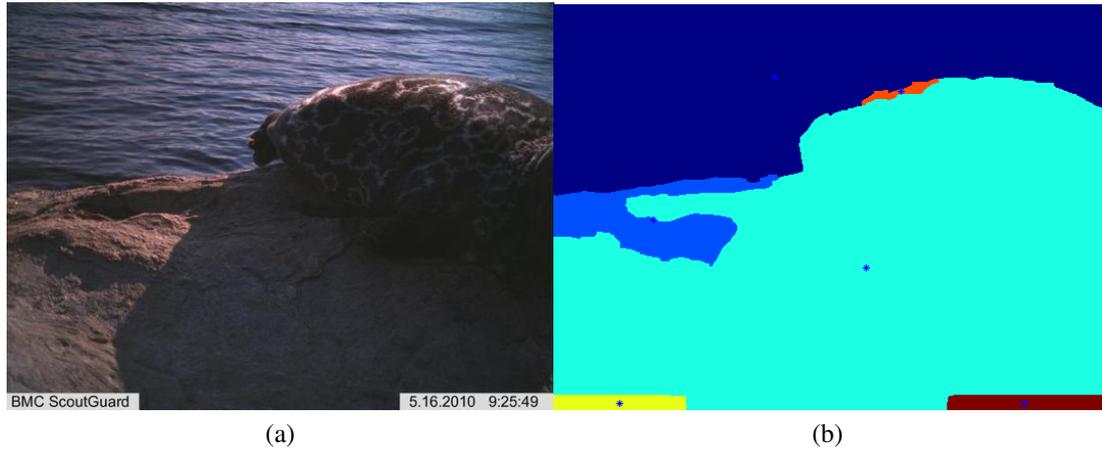
In order to train and test the segment classification methods a manual labeling of the segments into two classes (seal and background) was performed. This procedure was done by the author and provides the ground truth for training a classifier with correct classes. Example images after segmentation and labeling procedures are shown in Figure 24.



**Figure 24.** Labeling; from left to right: raw image, segmented image, labeled image.

However, in the process of manual segment labeling it was found that some of images contain too low quality to get the correct seal segment. This problem concerns many images with low contrast, very high brightness, underwater images and images where seals are close to another objects. It was decided to exclude the low quality images (images with less than 95% of pixels labeled correctly) from further work. If less than 5% of image pixels are labeled erroneously it does not significantly affect the further accuracy of the system. An example of unsupervised segmentation result with a low quality image is shown in Figure 25.

Finally, 363 images with satisfactory quality were chosen. All segments of these images were labeled manually to two classes by the author.



**Figure 25.** Example of image with low contrast which was not segmented correctly: (a) The image; (b) Segments.

### 5.2.3 Feature extraction and classification

The next step was to find the features and the classifier that provide the most accurate classification result of segments. Experiments were performed with different types of classifiers and a variety of features characterizing each segment. The features and classifiers were described in Section 3.

For the experiments 20 test images and 343 training images were randomly selected. In the each experiment accuracy of segmentation was computed

$$\text{Segmentation accuracy} = \frac{\text{Number of successfully segmented images}}{\text{All images}}. \quad (12)$$

The concept of "successfully segmented image" is determined as

$$\text{Segmentation success} = \frac{\text{Number of pixels successfully classified}}{\text{All image pixels}} > 0.95. \quad (13)$$

Table 2 shows the results of experiments.

Testing shows that the best feature is LPQ and the best classifier is SVM. With this combination, a mean accuracy equal to 0.81 was achieved. Examples of the segmentation process with the LPQ features and the SVM classifier are shown in Figure 26.

**Table 2.** Segmentation accuracy with different features, and classifiers.

Feature/classifier	K-NN (k=9)	Naive Bayes	SVM
Mean colors (3)	0.04	0.00	0.24
Center distance	0.64	0.52	0.00
Area	0.12	0.12	0.00
SFTA	0.24	0.52	0.44
LBP	0.32	0.00	0.28
LPQ	0.60	0.00	0.81

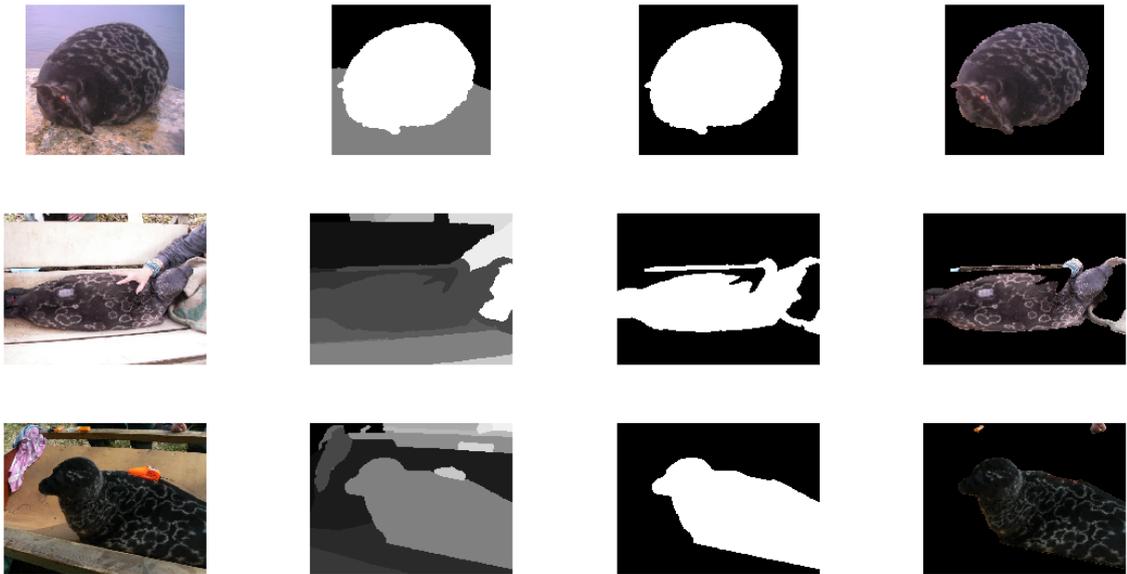
**Figure 26.** Segmentation algorithm results (From the left to the right): input image, segmented image, ground truth, the obtained result.

Figure shows that most of the segments are classified correctly and as a result a segmented image is formed comprising only the object of interest which is a seal and a small number of background pixels.

### 5.3 Identification

Next, the detected seal is classified as one of the individual seals. In connection with this, various tests were performed using different classifiers in conjunction with different sets of features. Experiments were performed with 10 and 40 seals as testsets. Initial experiments were done with 10 seal images. After this more comprehensive experiments were done using 40 seal images. In the first case, the training set consisted in total 64 images of 10 different seals. In the case of 40 seals the training set consisted of 185 images. For the experiments seals were selected containing five or more individual image

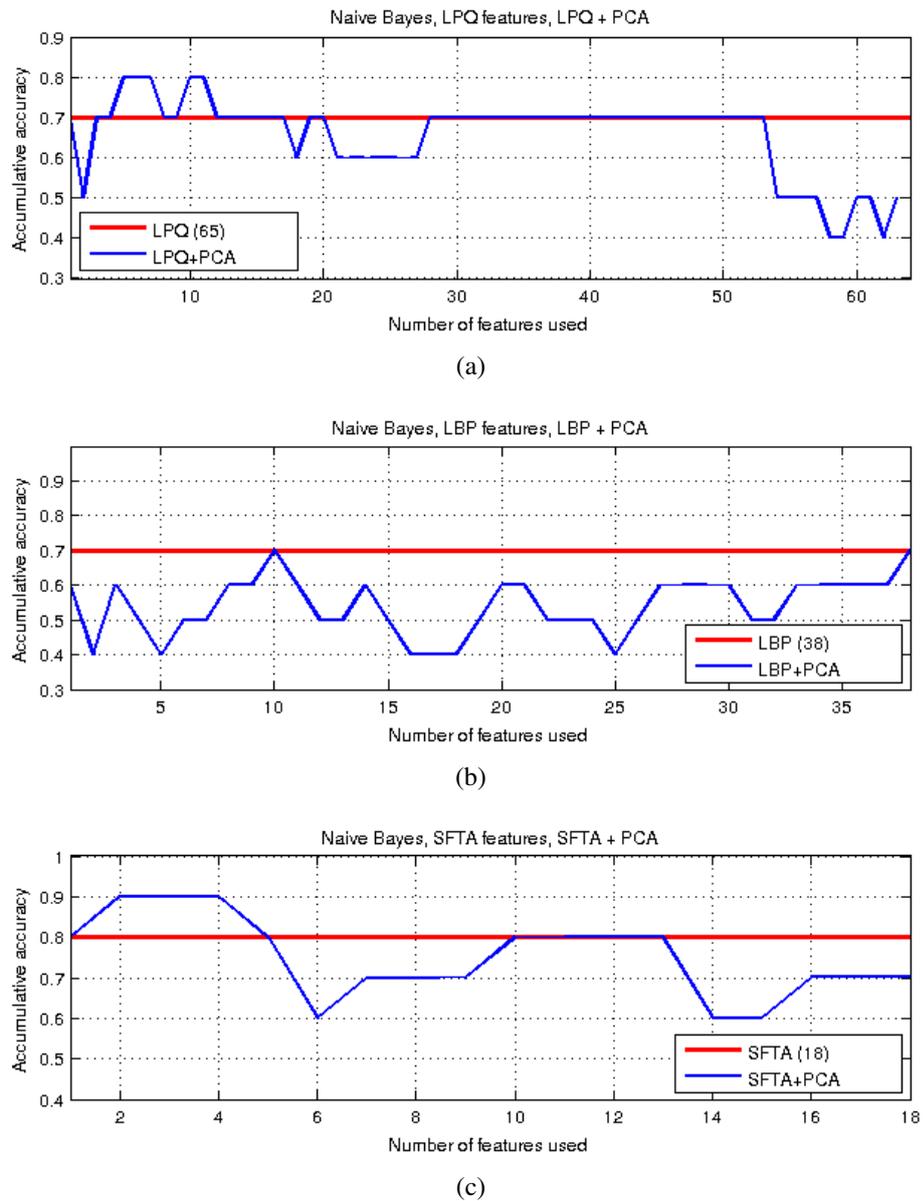
per seal. Seals with fewer than five images were excluded because they did not contain enough information. The results obtained by the experiments are shown in the following subsections.

### 5.3.1 Principal component analysis

#### 10 seals

In order to find the most suitable set of features for each classifier, an analysis of the classifier accuracy depending on the size of the feature set was made. To reduce the number of dimensions, the PCA was applied to features. Results of the experiments for different feature sets, with and without PCA are shown in Figure 27. For the raw features the whole feature vector is used and the number of features are varied only for the PCA processed features. In Figures 27(a)-(c) the  $x$ -axis represents the length of feature vector used and  $y$ -axis represents accumulative accuracy which equal to the value of fourth bar of CMS histogram. The value of the 4th bar represents a probability of true seal being among the four best matches. In the case of simple identification accuracy (how many images were correctly identified) does not change significantly when PCA is applied, accuracy of 4th bar was proposed more informative.

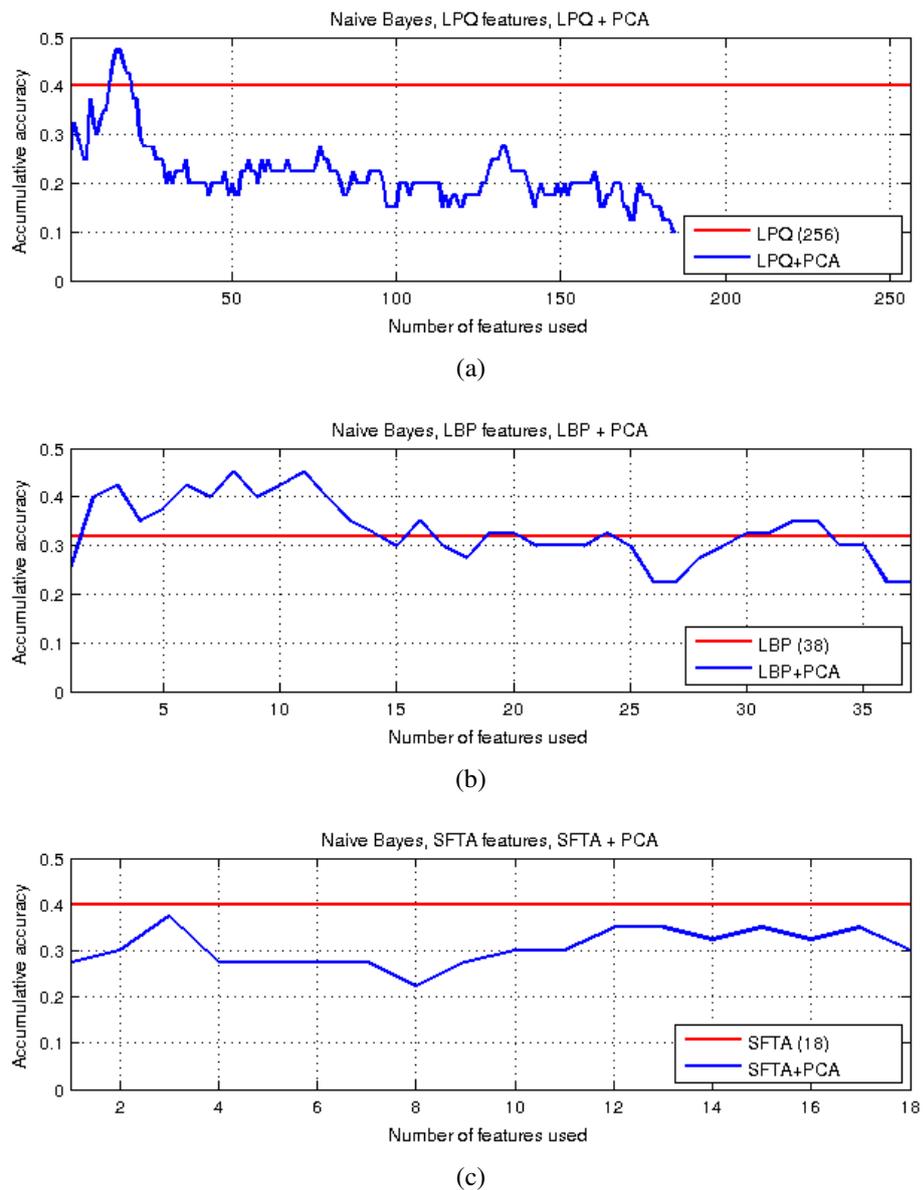
Figure 27 shows that the success of PCA depends on various parameters of the input data. In the first case (Figure 27(a)), the total number of LPQ features is large but PCA does not give significant improvements in output accuracy. In the second case (Figure 27(b)) PCA applied to the LBP features degrades the accuracy of the identification with any size of feature vector. Figure 27(c) shows that the PCA provides a significant increase in recognition accuracy for any chosen size of feature set. Thus, it is advisable to use PCA only in the case of the SFTA features, which was done in further experiments.



**Figure 27.** Comparison of feature accuracy applied with PCA and without PCA for 10 seals.

#### 40 seals

As in the previous experiments, for 40 seals PCA was applied and compared with raw features. Also for the raw features the whole feature vector is used and the number of features are varied only for the PCA processed features. Due to the fact that in this experiment a larger training set (185 images) was used, PCA of features was more successful and applicable. Results of the experiments for different feature sets, with and without PCA application are shown in Figure 28.



**Figure 28.** Comparison of feature accuracy applied with PCA and without PCA for 40 seals.

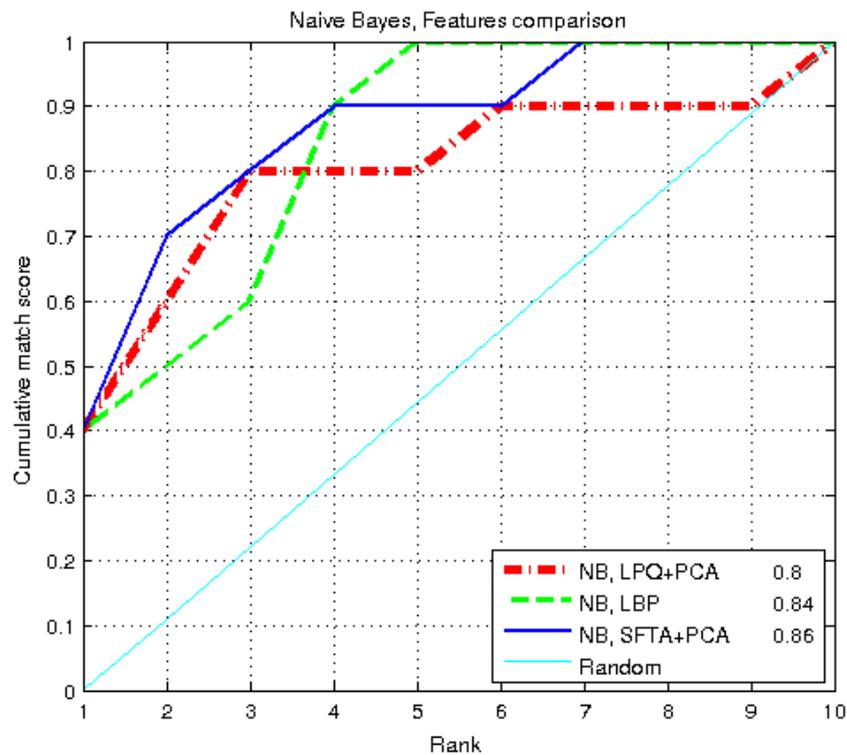
From Figure 28 it can be seen that in two out of three cases PCA shows improvement in classification accuracy. For example in Figure 28(a) it is shown that there is the increase in accuracy using PCA. Using a shorter length feature vector containing more important information NB classifier works significantly better. Figure 28(b) shows that PCA also improves the classification accuracy for LBP features. On the other hand, Figure 28(c) reflects the inapplicability of PCA for SFTA features and the reduction of accuracy in most possible configurations. Thus, the PCA was applied only for LPQ and LBP features and in the case of SFTA, raw features were used.

### 5.3.2 Identification performance with 10 seals

First, 10 random seal images of different individuals were considered as a test set. For the training set 5 - 9 images of each seal were used.

#### Features comparison

The experiments with different features were performed for each feature extraction methods and optimal parameters were chosen. Comparison of the LPQ, LBP, and SFTA features using the CMS histogram is presented in Figure 29.



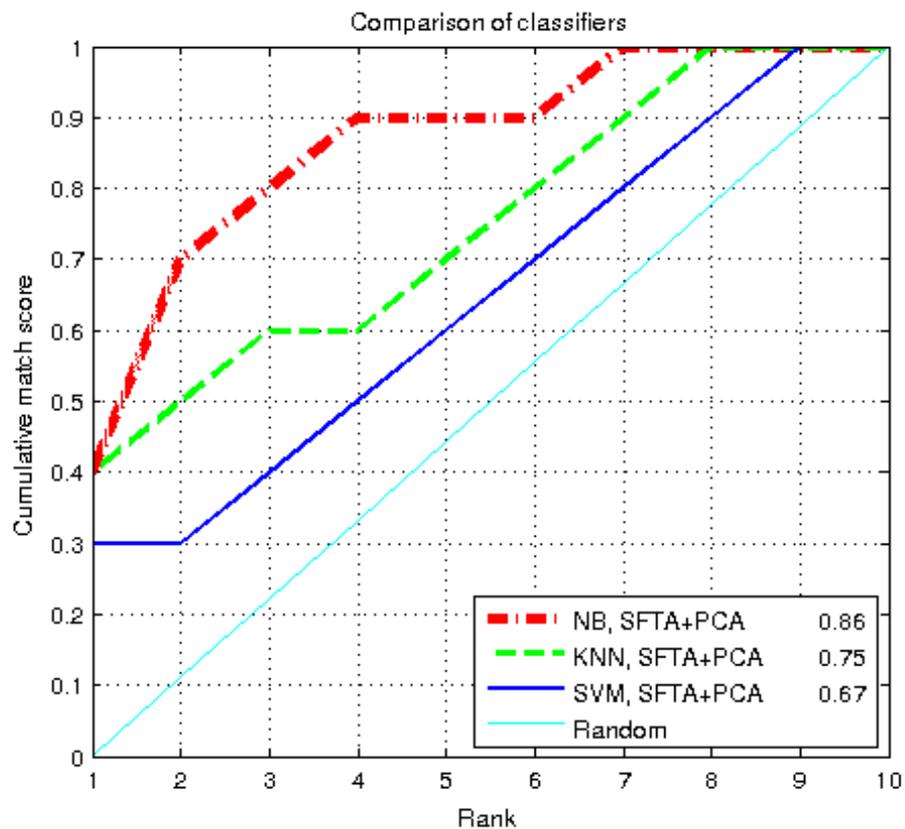
**Figure 29.** Comparison the best features applied with NB classifier.

As an estimate for the classification accuracy an accumulative accuracy was selected, which is computed as normalized area under the CMS histogram. Figure 29 shows that the highest accuracy with NB classifier was obtained using SFTA and LBP features with accuracy equal 0.86 and 0.84 correspondingly. However, LPQ features are not far from the first place with accuracy equal to 0.80. Basically the correct class is in the range of 2

- 6 best matches, that shows the necessity for further manual identification.

### Classifiers comparison

Next, a comparison of the NB, KNN, and SVM classifiers with the best set of the SFTA features was made. The experimental results are shown in Figure 30.



**Figure 30.** Comparison of different classifiers.

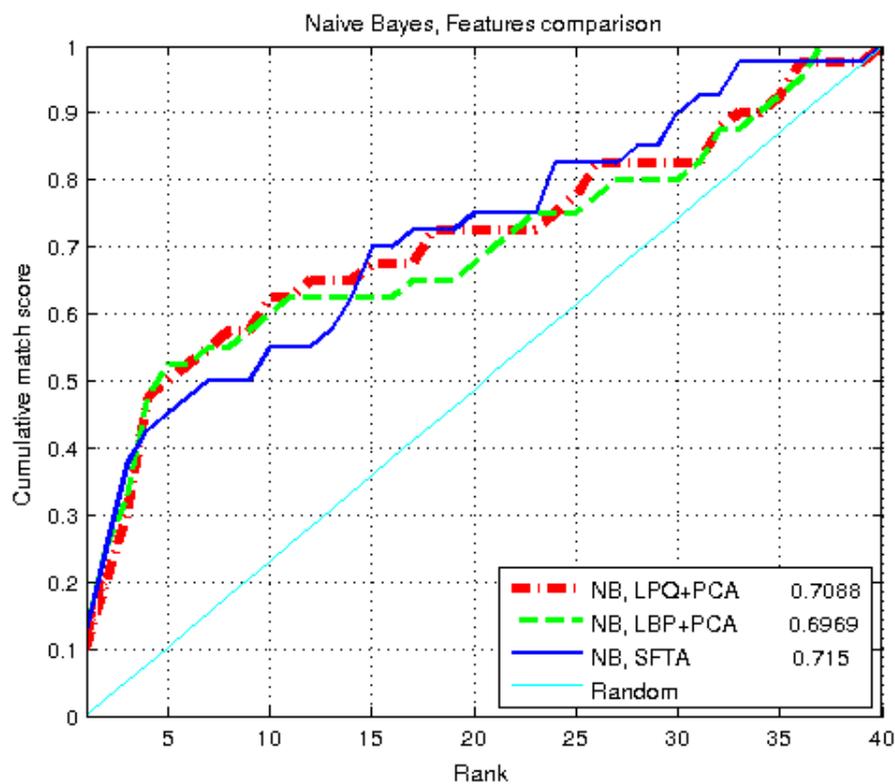
It can be seen that the NB classifier shows the highest accuracy (0.86). In contrast, other classifiers accept significant errors and can not be used for the most accurate operation. Also, the graph shows that the NB classifier can provide 90% identification accuracy with the necessity of manual selection only within four possible classes.

### 5.3.3 Identification performance with 40 seals

In the second case, 40 seals were used. The same features and classifiers were tested as with 10 seals.

#### Features comparison

Experiments with different features were performed in order to find the most accurate in the problem of identification of seals with 40 individuals. Similarly to the previous experiment the selection of parameters for each feature descriptor was done. Comparison of the best variations of the LPQ, LBP, and SFTA features is presented in Figure 31.



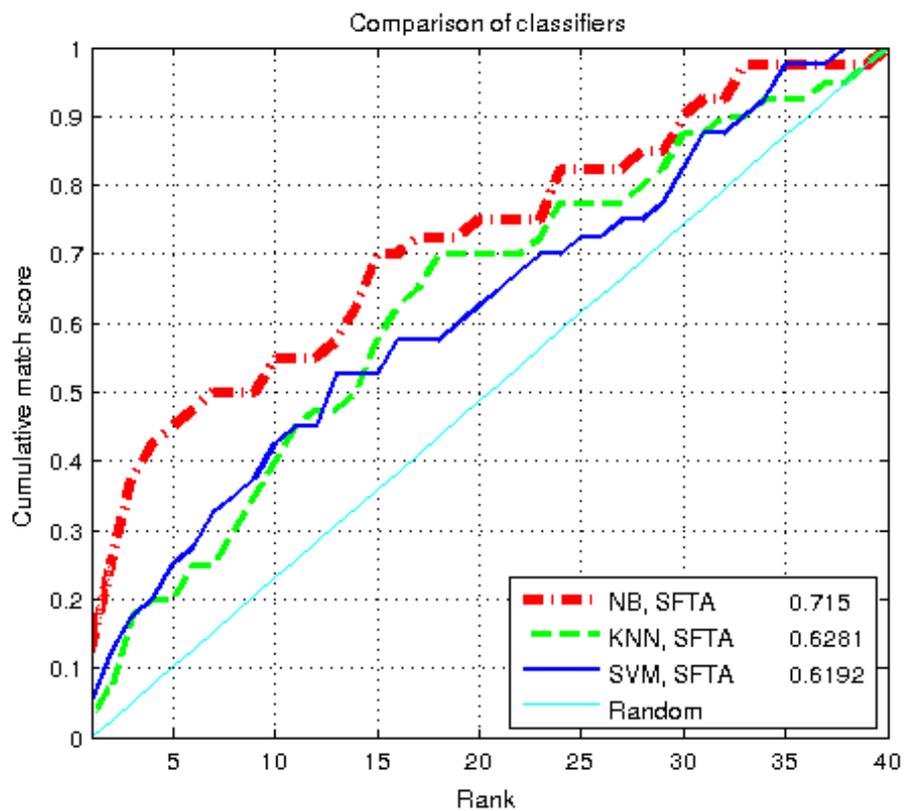
**Figure 31.** Comparison of the best features applied with NB classifier.

Figure 31 shows that the most successful in the task of seal identification are the LPQ and SFTA features with accuracy 0.7088 and 0.7150 correspondingly. The LBP features are less accurate. As it can be seen from Figure, it is possible to predict a set of 15 classes among which the true class is with 70% probability using the SFTA features.

Thus features provide a further relief to the task of manual identification from a 40 classes choice to 15 classes.

### Classifiers comparison

Next, a comparison of the NB, KNN, and SVM classifiers with the best set of SFTA features was made. The experimental results are shown in Figure 32.



**Figure 32.** Comparison of different classifiers.

The graph shows that the KNN and SVM classifiers provide almost the same accumulative accuracy. On the other hand, the NB classifier shows the best results with the total accumulative accuracy of 0.7150. Thus, it is can be concluded that the NB classifier is the most promising one. SVM classifier should be further evaluated with different kernels in the future research.

## 6 DISCUSSION

In this chapter the obtained results are discussed. The limitations of the proposed approach and possible problems with the results are discussed. Finally, thoughts for the future research topics and the future of the system in general are presented.

### 6.1 Proposed method

In this thesis, the algorithm for identification of Saimaa ringed seals was proposed. The algorithm receives an image of a seal as input and returns the most probable seal ids as output. All the steps of the proposed method are presented in Algorithm 1 showing the best methods selected based on the experiments. Based on the experiments the optimal threshold value  $t$  is 0.3.

---

**Algorithm 1** SealVision identification algorithm.

---

**Input:** Image.

**Output:** Seal id.

- 1: Perform the unsupervised  $gPb - owt - ucm$  segmentation on the input image with the threshold value  $t$ .
  - 2: Classify obtained segments to two classes (seal, background) using the SVM classifier trained with the LPQ features extracted from the training set of  $N$  seal images.
  - 3: Combine all segments labeled as a seal to the one large segment.
  - 4: Extract the SFTA features from the segmented image containing only the seal segment.
  - 5: For all seal ids, compute the score (posterior probabilities) for the image representing each individual using NB classifier.
  - 6: Sort the scores in the descending order.
  - 7: Return the set of best matches (ids with highest score).
- 

### 6.2 Results

Ground truth for image segmentation for a selected set of seal images was manually formed. The most suitable methods for segmentation and identification were found and tested with suitable parameters to optimize the accuracy.

At the stage of segmentation the SVM classifier was trained manually by segments from

363 images ( $N$ ) and could be further improved by training with a larger number of segments. Classifier is used to classify segments into two classes: background and seal, which has shown good accuracy (81%) paired with LPQ features. Based on the results, it can be concluded about the possibility of further using of these classifier and feature as the most promising and a little refinements that could show even greater accuracy.

At the stage of identification, novel seal imageset, containing 225 segmented seal images, useful for CV applications was formed using provided UEF image database. The possibility of using trained classifier to solve the problem of seal identification was investigated. For experiments two cases were considered: with 10 and 40 seals. Despite these limitations, reasonable good results were obtained with the NB classifier and a modern feature descriptors such as LBP, LPQ, and SFTA. SFTA features showed the best accumulative accuracy. Results of testing showed promising performance motivating for further study of these features and the classifier.

The results obtained from this research allow to draw conclusions about the prospects of establishing and improving the identification system of the Saimaa ringed seals, which in the future could be used in the work of biologists. This task implies the exact or at least approximate identification of the individual seal (name) from the database containing images of about 200-300 individuals. The problem of a large number of possible classes (seals) is also complicated by a number of serious image distortions of the used image database.

Results from both parts of the algorithm, segmentation and identification, are significantly affected by the quality of images. As stated in Section 5, Saimaa seals images can have large variations in illumination, contrast and the presence of foreign objects covering the body of the seal, such as rocks, grass, water, and other animals. The distortions, which are visible in images such as pictures taken at night with flash, underwater, or in cloudy weather, makes the segmentation and identification tasks more difficult.

The second challenging problem of the results accuracy is the pose of the seal on the images. For a successful identification of the image there must be clearly visible characteristic gray rings on the side of the animal's body that allow to recognize individual. However, in the images obtained by automated cameras, a seal may be located in any position relative to the camera. This means that the arrangement of the rings varies suffers from distortions and is quite different when viewed from different sides of the seal. It is assumed that this problem can be solved by the presence of a large database of images for each individual animal at various angles. However, as stated above images must be

checked for distortion.

Finally, the third possible problem is changing the number of individual seals (classes) caused by the constant birth of new seals and the death of the old ones. In this regard, images of previously unknown individuals most likely absent in the database and identifying errors occur. Over time, the database would contain an increasing number of possible seals (classes) that would complicate the work of the classifier. To solve this problem there is a need to edit a permanent base and to manually delete unnecessary classes (seals) by experts.

The results obtained in this study take into account problems described above and are limited. To test the proposed algorithm, a limited set of images was used because it was necessary to select only the images with adequate quality containing unique attributes of the animal. At the stage of segmentation significant errors were not found as the skin of seals has a similar texture visibly distinguishable from the background. However, at the stage of identification for each possible class (seals) there are existed only 5-9 images suitable for training. In connection with the above-described problems of various positions of the individual as well as fluctuations in the illumination, the results of the identification leave a lot of room for further improvement of the methods.

### **6.3 Future research**

The obtained results open the way for further research in the topic of Saimaa ringed seal identification in particular, and the identification of animals in general. In further studies the possibility to significantly improve the accuracy of identification by using a large database of seals' images should be considered. It is necessary to conduct a study of possible image preprocessing to reduce the requirements of the image quality. In turn, the results could be greatly improved by the testing of classifier with different features characterizing the fur of seals. On the other hand, creation of a system of identification based on multiple classifications and different sets of features should be considered. The most accurate identification feature could be a location of the rings on the body of Saimaa seal relative to each other, but the extraction of this information is difficult to implement and require more research. Finally, for a real system it would be necessary to develop additional an easy interaction program for manual classification by people.

## 7 CONCLUSION

The purposes of this research were to create an image-based Saimaa ringed seal identification algorithm, and to investigate the possibility of automatic seal identification. Results of experiments with different components helped to create the identification system. First, the issue was reviewed and a picture of a process with subproblems were outlined. The subproblems were studied further through the literature and possible solution to each of them were proposed. The solution studied further, compared, and algorithms were implemented to solve the problems. An useful dataset for computer vision experiments was created during the research from raw database collected by UEF. Finally, the algorithms were tested with this new seal dataset.

This study is the first in the world dedicated to the automatic identification of the Saimaa ringed seal. It has been already developed several different animal identification systems, but in each case there are their own peculiarities and identification of each type is unique. Saimaa ringed seal has a very characteristic features of its fur texture that can be used to recognize individuals.

Computer vision system could be improved and used in the future to assist biologists in the problems of tracking migration Saimaa seal with the camera. At the stage of segmentation accuracy was obtained by 81% when using SVM classifier and LPQ features. At the stage of identification the SFTA features showed the best accuracy. An attempt to identify the 10 and 40 seals showed results of 90% and 47.5%, respectively, with necessity of further manual class selection of the four possible classes.

This thesis also examined the many practical aspects such as the development of the most effective algorithm, the choice of optimal parameters of the classifier, and used features and the impact of external factors on the accuracy of identification. The developed system could be further improved by using a larger database of images and a selection of optimal parameters. This system shows promising results and practical implementation of the system is possible.

## REFERENCES

- [1] Tero Sipilä, Tuomo Kokkonen, and Jouni Koskela. The growth of the saimaa ringed seal population is unstable. Factsheet, Metsähallitus, Vantaa, 2013.
- [2] Metsähallitus Natural Heritage Services. Hyljekanta 2014, 2014. <http://www.metsa.fi/sivustot/metsa/fi/luonnonsuojelu/lajitjaluontotyypit/uhanalaisetelaimet/saimaannorppa/hyljekanta2014/sivut/default.aspx>, [Accessed 15.4.2015].
- [3] University of Eastern Finland. Saimaa ringed seal *Phoca hispida saimensis* - the most endangered seal in the world? Factsheet, Metsähallitus, Vantaa, 2009.
- [4] University of Eastern Finland. Saimaa ringed seals database, 2015. Available upon the request.
- [5] Kit M. Kovacs, Alex Aguilar, David Aurioles, Vladimir Burkanov, Claudio Campaigna, Nick Gales, Tom Gelatt, Simon D. Goldsworthy, Simon J. Goodman, Greg J. G. Hofmeyr, Tero Härkönen, Lloyd Lowry, Christian Lydersen, Jan Schipper, Tero Sipilä, Colin Southwell, Simon Stuart, Dave Thompson, and Fritz Trillmich. Global threats to pinnipeds. *Marine Mammal Science*, 28(2):414–436, 2012.
- [6] University of Eastern Finland. Development of conservation and monitoring methods, January 2015. <http://www2.uef.fi/en/norppa/suojelu>, [Accessed 3.2.2015].
- [7] Cecilie E Bugge, John Burkhardt, Kari S Dugstad, Tone Berge Enger, Monika Kasprzycka, Andrius Kleinauskas, Marit Myhre, Katja Scheffler, Susanne Ström, and Susanne Vetlesen. Biometric methods of animal identification. *Course notes*, pages 1–6, 2011.
- [8] UG Barron, F Butler, K McDonnell, S Ward, et al. The end of the identity crisis? advances in biometric markers for animal identification. *Irish Veterinary Journal*, 62(3):204–208, 2009.
- [9] Tilo Burghardt. *A general introduction to visual animal biometrics*. Technical Report, Visual Information Laboratory, University of Bristol, 2012.
- [10] Ingar Jostein Qien, Tomas Aarvak, Svein-Hékon Lorentsen, and Georg Bangjord. Use of individual differences in belly patches in population monitoring of Lesser whitefronted goose *Alastair erythropus* at a staging ground. *Fauna norvegica*, 19:69–76, 1996.

- [11] W. E. Petersen. The identification of the bovine by means of nose-prints. *Journal of Dairy Science*, 5(3):249–258, 1922.
- [12] Morris Hirsch, Edmund F. Graham, and Arthur E. Dracy. A classification for the identification of bovine noseprints. *Journal of Dairy Science*, 35(4):314 – 319, 1952.
- [13] C. P. Rusk, C. R. Blomeke, M. A. Balschweid, S. J. Elliott, and D. Baker. Evaluation of retinal imaging technology for 4-h beef and sheep identification. *Journal of Extension*, 44(5):1–33, 2006.
- [14] Brian M. Howell, Clinton P. Rusk, Christine R. Blomeke, Renee K. McKee, and Ronald P. Lemenager. Perceptions of retinal imaging technology for verifying the identity of 4-h ruminant animals. *Journal of Extension*, 46(5), 2008.
- [15] Gerard Corkery, Ursula A Gonzales-Barron, Francis Butler, Kevin McDonnell, and Shane Ward. A preliminary investigation on face recognition as a biometric identifier of sheep. *Transactions of the ASABE*, 50(1):313–320, 2007.
- [16] Nilsson K. Cameron J., Jacobson C. and R.ögnvaldsson T. Identifying laboratory rodents using earprints. *National Centre for Replacement, Refinement, and Reduction of Animals in Research (NC3Rs)*, 11:1–4, 2007.
- [17] Arid Recovery News. Keeping cool in the desert, November 2012. [http://www.aridrecovery.org.au/\\_blog/Arid\\_Recovery\\_News/post/Keeping\\_cool\\_in\\_the\\_desert](http://www.aridrecovery.org.au/_blog/Arid_Recovery_News/post/Keeping_cool_in_the_desert), [Accessed 5.4.2015].
- [18] Emily LC Shepard, Rory P Wilson, Flavio Quintana, Agustina Gomez-Laich, Nikolai Liebsch, Diego A Albareda, Lewis G Halsey, Andrian Gleiss, David T Morgan, Andrew E Myers, et al. Identification of animal movement patterns using tri-axial accelerometry. *Endangered Species Research*, 10(1):47–60, 2010.
- [19] Reinhard Klette. Concise computer vision - an introduction into theory and algorithms. pages 1–413, 2014.
- [20] E. R. Davies. *Machine Vision: Theory, Algorithms, Practicalities*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2004.
- [21] Publitek Marketing Communications. Versatile leds drive machine vision in automated manufacture, 2012. <http://www.digikey.com/en/articles/techzone/2012/jan/versatile-leds-drive-machine-vision-in-automated-manufacture>, [Accessed 6.5.2015].

- [22] Eric H. Fegraus, Kai Lin, Jorge A. Ahumada, Chaitan Baru, Sandeep Chandra, and Choonhan Youn. Data acquisition and management software for camera trap data: A case study from the {TEAM} network. *Ecological Informatics*, 6(6):345 – 353, 2011.
- [23] Carlos JR Anderson. *Individual identification of polar bears by whisker spot patterns*. PhD thesis, University of Central Florida, Orlando, Florida, 2007.
- [24] Alaa Tharwat, Tarek Gaber, AboulElla Hassanien, Hasssan A. Hassanien, and Mohamed F. Tolba. Cattle identification using muzzle print images based on texture features approach. In Pavel Kömer, Ajith Abraham, and Václav Snášel, editors, *Proceedings of the Fifth International Conference on Innovations in Bio-Inspired Computing and Applications IBICA 2014*, volume 303 of *Advances in Intelligent Systems and Computing*, pages 217–227. Springer International Publishing, 2014.
- [25] Sanaul Hoque, MA Azhar, and Farzin Deravi. Zoometrics-biometric identification of wildlife using natural body marks. *International Journal of Bio-Science and Bio-Technology*, 3(3):45–53, 2011.
- [26] Kelly M. Halloran, James D. Murdoch, and Matthew S. Becker. Applying computer-aided photo-identification to messy datasets: a case study of thornicroft’s giraffe (*giraffa camelopardalis thornicrofti*). *African Journal of Ecology*, pages 147–155, 2014.
- [27] Nathan F. Bendik, Thomas A. Morrison, Andrew G. Gluesenkamp, Mark S. Sanders, and Lisa J. O’Donnell. Computer-assisted photo identification outperforms visible implant elastomers in an endangered salamander, *Eurycea tonkawae*. *PLoS ONE*, 8(3):e59424, 03 2013.
- [28] A. Branzan Albu, G. Wiebe, P. Govindarajulu, C. Engelstoft, and K. Ovatska. Towards automatic modelbased identification of individual sharp-tailed snakes from natural body markings. In *Proceedings of ICPR Workshop on Animal and Insect Behaviour*, Tampa, FL, USA, 2008.
- [29] Lokman Kayci Yılmaz Kaya and Ramazan Tekin. A computer vision system for the automatic identification of butterfly species via gabor-filter-based texture features and extreme learning machine: Gf+ elm. *Tem Journal*, 2(1):13, 2013.
- [30] Neeraj Kumar, Peter N Belhumeur, Arijit Biswas, David W Jacobs, W John Kress, Ida C Lopez, and João VB Soares. Leafsnap: A computer vision system for automatic plant species identification. In *Computer Vision–12th European Conference on Computer Vision*, pages 502–516. Springer, 2012.

- [31] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.
- [32] J.P. Crall, C.V. Stewart, T.Y. Berger-Wolf, D.I. Rubenstein, and S.R. Sundaresan. Hotspotter - patterned species instance recognition. *IEEE Workshop on Applications of Computer Vision (WACV)*, pages 230–237, Jan 2013.
- [33] Xiaoyuan Yu, Jiangping Wang, Roland Kays, Patrick Jansen, Tianjiang Wang, and Thomas Huang. Automated identification of animal species in camera trap images. *EURASIP Journal on Image and Video Processing*, 2013(1):52, 2013.
- [34] Marja Niemi. *Behavioural ecology of the Saimaa ringed seal - implications for conservation*. PhD thesis, University of Eastern Finland, November 2013.
- [35] Meeri Koivuniemi. *E-mail communication*. PhD thesis, University of Eastern Finland Department of Biology, 2015.
- [36] Waseem Khan. Image segmentation techniques: A survey. *Journal of Image and Graphics*, 1(4):166–170, 2013.
- [37] L.G. Shapiro and G.C. Stockman. *Computer Vision*. Prentice Hall, 2001.
- [38] Wikipedia. Image segmentation, 2015. [http://en.wikipedia.org/wiki/Image\\_segmentation](http://en.wikipedia.org/wiki/Image_segmentation), [Accessed 7.5.2015].
- [39] Catalin Amza. A review on neural network–based image segmentation techniques. *Mechanical and Manufacturing Engineering*, pages 1–23, 2012.
- [40] N Senthilkumaran and R Rajesh. Edge detection techniques for image segmentation—a survey of soft computing approaches. *International journal of recent trends in engineering*, 1(2), 2009.
- [41] Hongwei Zhu and Otman Basir. Fuzzy sets theory based region merging for robust image segmentation. In Lipo Wang and Yaochu Jin, editors, *Fuzzy Systems and Knowledge Discovery*, volume 3613 of *Lecture Notes in Computer Science*, pages 426–435. Springer Berlin Heidelberg, 2005.
- [42] Joachim Weickert. Efficient image segmentation using partial differential equations and morphology. *Pattern Recognition*, 34:2001, 1998.
- [43] Chinki Chandhok. A novel approach to image segmentation using artificial neural networks and k-means clustering. *International Journal of Engineering Research and Applications (IJERA) ISSN*, 2(3):274–279, 2012.

- [44] Salem Saleh Al-Amri, Namdeo V Kalyankar, et al. Image segmentation by using threshold techniques. *arXiv preprint arXiv:1005.4020*, 2010.
- [45] Stephen Gould, Tianshi Gao, and Daphne Koller. Region-based segmentation and object detection. In *Advances in neural information processing systems*, pages 655–663, 2009.
- [46] Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(5):898–916, May 2011.
- [47] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. From contours to regions: An empirical evaluation. *IEEE Conference on Computer Vision and Pattern Recognition, 2009.*, pages 2294–2301, June 2009.
- [48] The Berkeley Segmentation Dataset and Benchmark. Boundary detection benchmark: Algorithm ranking, 2013. <http://www.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/bench/html/algorithms.html>, [Accessed 10.11.2014].
- [49] M. Maire, P. Arbelaez, C. Fowlkes, and J. Malik. Using contours to detect and localize junctions in natural images. In *IEEE Conference on Computer Vision and Pattern Recognition, 2008.*, pages 1–8, June 2008.
- [50] P. Arbelaez. Boundary extraction in natural images using ultrametric contour maps. In *Conference on Computer Vision and Pattern Recognition Workshop, 2006.*, pages 182–182, June 2006.
- [51] A.F. Costa, G. Humpire-Mamani, and A.J.M. Traina. An efficient algorithm for fractal analysis of textures. In *25th Conference on Graphics, Patterns and Images (SIBGRAPI), 2012*, pages 39–46, Aug 2012.
- [52] Timo Ahonen, Jiri Matas, Chu He, and Matti Pietikäinen. Rotation invariant image description with local binary pattern histogram fourier features. In Arnt-Borre Salberg, JonYngve Hardeberg, and Robert Jenssen, editors, *Image Analysis*, volume 5575 of *Lecture Notes in Computer Science*, pages 61–70. Springer Berlin Heidelberg, 2009.
- [53] Ville Ojansivu and Janne Heikkilä. Blur insensitive texture classification using local phase quantization. In Abderrahim Elmoataz, Olivier Lezoray, Fathallah Nouboud, and Driss Mammass, editors, *Image and Signal Processing*, volume 5099 of *Lecture Notes in Computer Science*, pages 236–243. Springer Berlin Heidelberg, 2008.

- [54] Ping-Sung Liao, Tse-Sheng Chen, and Pau-Choo Chung. A fast algorithm for multi-level thresholding. *Journal of Information Science and Engineering*, 17(5):713–727, 2001.
- [55] Timo Ojala, Matti Pietikäinen, and David Harwood. A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition*, 29(1):51–59, January 1996.
- [56] Timo Ahonen, Abdenour Hadid, and Matti Pietikäinen. Face recognition with local binary patterns. In *Computer vision-eccv 2004*, pages 469–481. Springer, 2004.
- [57] Timo Ojala, Matti Pietikainen, and Topi Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002.
- [58] Wikipedia. Naive bayes classifier, 2011. [http://en.wikipedia.org/wiki/Naive\\_Bayes\\_classifier](http://en.wikipedia.org/wiki/Naive_Bayes_classifier), [Accessed 1.5.2015].
- [59] Yoshimasa Tsuruoka and Jun’ichi Tsujii. Training a naive bayes classifier via the em algorithm with a class distribution constraint. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 127–134. Association for Computational Linguistics, 2003.
- [60] Wikibooks. Data mining algorithms in r/classification/knn, 2013. [http://en.wikibooks.org/wiki/Data\\_Mining\\_Algorithms\\_In\\_R/Classification/kNN](http://en.wikibooks.org/wiki/Data_Mining_Algorithms_In_R/Classification/kNN), [Accessed 3.2.2015].
- [61] Mohammed J. Zaki and Jr. Wagner Meira. *Fundamentals of Data Mining Algorithms*. Cambridge University Press, 2010.
- [62] Wikibooks. Data mining algorithms in r/classification/svm, 2013. [http://en.wikibooks.org/wiki/Data\\_Mining\\_Algorithms\\_In\\_R/Classification/SVM](http://en.wikibooks.org/wiki/Data_Mining_Algorithms_In_R/Classification/SVM), [Accessed 7.2.2015].
- [63] Richard O Duda, Peter E Hart, and David G Stork. *Pattern classification*. John Wiley & Sons, 2012.
- [64] Vladimir N. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag New York, Inc., New York, NY, USA, 1995.
- [65] Mahesh Pal. Multiclass approaches for support vector machine based land cover classification. *arXiv preprint arXiv:0802.2411*, 2008.

- [66] S. Knerr, L. Personnaz, and G. Dreyfus. Single-layer learning revisited: a stepwise procedure for building and training a neural network. In Françoise Fogelman Soulie and Jeanny Hérault, editors, *Neurocomputing*, volume 68 of *NATO ASI Series*, pages 41–50. Springer Berlin Heidelberg, 1990.
- [67] Chih-Wei Hsu and Chih-Jen Lin. A comparison of methods for multiclass support vector machines. *IEEE Transactions on Neural Networks*, 13(2):415–425, Mar 2002.
- [68] Jonathon Shlens. A tutorial on principal component analysis. *Computing Research Repository*, abs/1404.1100, 2014.
- [69] Mark Richardson. *Principal Component Analysis*. Faculty of Natural Sciences and Engineering, University of Ljubljana, 2009.
- [70] Sebastian Raschka. Implementing a principal component analysis (pca) in python step by step, 2014. [http://sebastianraschka.com/Articles/2014\\_pca\\_step\\_by\\_step.html#drop\\_labels](http://sebastianraschka.com/Articles/2014_pca_step_by_step.html#drop_labels), [Accessed 10.5.2015].
- [71] Duncan Gillies. Intelligent data analysis and probabilistic inference, 2014. <http://www.doc.ic.ac.uk/~dfg/ProbabilisticInference/Bayesian.html>, [Accessed 15.3.2015].