NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

MAP CLASSIFICATION IN IMAGE DATA

by

Frank Fiebiger

September 2015

Thesis Advisor:	Mathias N. Kölsch
Second Reader:	Samuel E. Buttrey

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704–0188

1. AGENCY USE ONLY *(Leave Blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	09-25-2015	Master's Thesis 10-01-2014 to 09-25-2015

4. TITLE AND SUBTITLE

MAP CLASSIFICATION IN IMAGE DATA

5. FUNDING NUMBERS

6. AUTHOR(S)

Frank Fiebiger

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Postgraduate School
Monterey, CA 93943

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

N/A

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

The views expressed in this document are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol Number: N/A.

12a. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(maximum 200 words)*

The digital era has led to an unprecedented increase in the amount of information available, of which an essential part is represented by visual data. The data forensics community asks for machine solutions to face the proliferation of image data. This thesis addresses the specific problem of distinguishing two-dimensional map images from other image content by examining two computational methods: Convolutional Neural Networks (CNNs) and Bag of Words (BOW). No information about current automated solutions for the mentioned task is available. The CNN used in this research consists of 60 million parameters and 650,000 neurons in eight weighted layers, is pre-trained on 1,000 classes, and provides an immense learning capacity. The BOW method uses a visual vocabulary, constructed by clustering higher-level image information, to classify unknown images by comparing their contained visual words with a content-specific vocabulary of a classifier. Both methods are evaluated in terms of recall and precision, or percentage of correctly and incorrectly classified images. The data collection consists of 1,200 map images called positives, subdivided into four sub-classes, and an additional 1,200 images without map content, called negatives. Results with a recall up to 99.17% and corresponding precision up to 97.01% support the idea of implementing CNN and BOW as the backbone of a computer-based classification application.

14. SUBJECT TERMS	15. NUMBER OF PAGES
map, classification, bag of words, BOW, convolutional neural network, CNN	77
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UU

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2–89)
Prescribed by ANSI Std. 239–18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

MAP CLASSIFICATION IN IMAGE DATA

Frank Fiebiger
Major, German Army
Diplom - Ingenieur, Bundeswehr University Munich, 2004

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
September 2015

Author: Frank Fiebiger

Approved by: Mathias N. Kölsch, Ph.D.
Thesis Advisor

Samuel E. Buttrey, Ph.D.
Second Reader

Patricia A. Jacobs, Ph.D.
Chair, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The digital era has led to an unprecedented increase in the amount of information available, of which an essential part is represented by visual data. The data forensics community asks for machine solutions to face the proliferation of image data. This thesis addresses the specific problem of distinguishing two-dimensional map images from other image content by examining two computational methods: Convolutional Neural Networks (CNNs) and Bag of Words (BOW). No information about current automated solutions for the mentioned task is available. The CNN used in this research consists of 60 million parameters and 650,000 neurons in eight weighted layers, is pre-trained on 1,000 classes, and provides an immense learning capacity. The BOW method uses a visual vocabulary, constructed by clustering higher-level image information, to classify unknown images by comparing their contained visual words with a content-specific vocabulary of a classifier. Both methods are evaluated in terms of recall and precision, or percentage of correctly and incorrectly classified images. The data collection consists of 1,200 map images called positives, subdivided into four sub-classes, and an additional 1,200 images without map content, called negatives. Results with a recall up to 99.17% and corresponding precision up to 97.01% support the idea of implementing CNN and BOW as the backbone of a computer-based classification application.

THIS PAGE INTENTIONALLY LEFT BLANK

Table of Contents

1 Introduction **1**
1.1 Background 1
1.2 Scope of This Thesis 5

2 Literature Review **7**
2.1 Classification Methods: CNNs and BOW 9
2.2 Interest Point Detectors and Feature Descriptors 10

3 Methodology **13**
3.1 Dataset 13
3.2 Experiments 19

4 Evaluation of the Results **27**
4.1 BOW with MATLAB and Two Classes: Positives and Negatives 27
4.2 BOW with MATLAB, Four Map Sub-Classes, and Negatives 30
4.3 BOW with EasyCV 37
4.4 Convolutional Neural Network 39
4.5 Insights for the Research Questions 43

5 Conclusions and Future Work **47**
5.1 Future Work 47
5.2 Conclusions 48

List of References **51**

Initial Distribution List **59**

THIS PAGE INTENTIONALLY LEFT BLANK

List of Figures

Figure 1.1 Minute-by-minute Internet data traffic. 3

Figure 1.2 A map of words, showing words related to the initial term “class.” 4

Figure 3.1 Examples of chosen map sub-classes. 15

Figure 3.2 Architecture of a deep CNN. 24

Figure 4.1 Examples of misclassifications. 35

Figure 4.2 Comparison of the largest sketch image with the largest pilotage chart image. 36

Figure 4.3 Examples of misclassified negatives. 38

Figure 4.4 ROC curve for the best performing visual word constellations. . . 38

THIS PAGE INTENTIONALLY LEFT BLANK

List of Tables

Table 3.1 Table of chosen categories from ImageNet. 18

Table 3.2 Experimental setup for 40 separate runs to examine MATLAB's BOW implementation. 21

Table 3.3 Overview of the hyper-parameter space of detector-descriptor constellations for BOW. 22

Table 3.4 Experimental setup for four runs to examine the deep CNN BVLC Reference CaffeNet. 25

Table 4.1 Results for BOW with MATLAB on two classes: positives and negatives. 29

Table 4.2 Results for 10 runs with 250 visual words. 29

Table 4.3 Confusion matrices of BOW on four sub-classes and the negatives class, using 500 visual words. 33

Table 4.4 Confusion matrices of BOW on four sub-classes and the negatives class, using 1000 visual words. 34

Table 4.5 Confusion matrix of a run without the sub-class sketches. 34

Table 4.6 Results for BOW with EasyCV on two classes: positives and negatives. 37

Table 4.7 Runs with detector/descriptor combinations using EasyCV. 39

Table 4.8 CNN with imbalanced training set–Confusion matrices of four sub-classes and negatives. 41

Table 4.9 CNN with balanced training set–Confusion matrices of four sub-classes and negatives. 42

Table 4.10 Overview of the best results for the different classification approaches. 45

THIS PAGE INTENTIONALLY LEFT BLANK

List of Acronyms and Abbreviations

BOW Bag of Words

BRIEF Binary Robust Independent Elementary Features

BRISK Binary Robust Invariant Scalable Keypoints

BVLC Berkeley Vision and Learning Center

CNN Convolutional Neural Network

CVAC Computer Vision Algorithm Collection

FAST Features from Accelerated Segment Test

HOG Histograms of Oriented Gradients

NPS Naval Postgraduate School

ORB Oriented Fast and Rotated BRIEF

SIFT Scale Invariant Feature Transform

SURF Speeded Up Robust Features

SVM Support Vector Machine

THIS PAGE INTENTIONALLY LEFT BLANK

Executive Summary

The digital era has led to an unprecedented increase in the amount of information available. Over the past decades, digital data has begun to replace–or at least to complement–traditional systems of information exchange and storage. Interpreting and analyzing this data has become a challenge to the individuals, companies, and institutions who could benefit. Today, an essential part of this increase of digital data is represented by visual data, produced by traditional digital cameras or cameras integrated into smartphones, tablets, and computers. The proliferation of visual data is a special challenge for the data forensics community, which therefore seeks machine solutions. This thesis addresses the specific problem of distinguishing two-dimensional map images from other image content by examining two computational, state-of-the-art methods: Convolutional Neural Networks (CNNs) and the Bag of Words (BOW) method.

The idea of BOW is introduced for text classification problems by Harris (1954) and later adapted to computational methods by, for example, Joachims (1998). By counting the occurrence of words in a document, the resulting vocabulary histogram can give significant evidence about the content. A visual vocabulary can be constructed by computing robust descriptors on detected interest points in all training images and clustering the descriptors; the clusters become the visual words. A classifier, trained with labeled input images and describing them with the visual vocabulary, can then evaluate the similarity between a new, unknown image and the trained classes, by examining the descriptor histogram of the new image.

CNNs, the second method examined, find access to computer applications during the early 1990s for tasks like word and character recognition (Bengio et al., 1994; LeCun & Bengio, 1994). They are introduced to image content detection and classification (LeCun & Bengio, 1995; Nowlan & Platt, 1995) and used for face detection and recognition. With the increase of computational power they become the focus of a great deal of research for classification tasks in huge image datasets containing millions of pictures. Consisting of 60 million parameters and 650,000 neurons in eight weighted layers and pre-trained on 1,000 classes, the CNN used in this research offers an immense learning capacity.

The data collection for this evaluation consists of 1,200 map images called positives, divided into sub-classes including basic maps, pilotage charts, web maps, and sketches. Additionally, 1,200 images without map content, called negatives, are chosen from ImageNet (Krizhevsky et al., 2012) to complete the dataset. The performance of CNN and BOW is evaluated either by the observed rates for correct and incorrect classifications or by recall and precision. Recall is defined by the proportion of correctly classified positives (true positives) in comparison to the overall number of positives. Precision shows the proportion of true positives among all images classified as positives, which provides information about the number of negatives within the predicted positive classifications.

Both methods produce recalls up to 99.17% on tasks with two classes. BOW's best precision of 97.01% tops the CNNs by over 3.5%. Trained on sub-classes, the results using a CNN are either biased in favor of overrepresented sub-classes, if the training set is imbalanced, or the low number of training images in the balanced scenario trains generally less reliable classifiers. A possible solution is a balanced dataset with substantially more training data, which can be gathered by either augmenting the underrepresented sub-classes, or raising the number of sub-class images by intensifying the online-search for images. With up to 100% correct classifications, BOW can generate a remarkable result on one of the underrepresented sub-classes. Instead, on image quantity, the performance of BOW depends much more on the specific, descriptor-related image information and on intra-class variability. Faced with web maps, a sub-class with high intra-class variability, and sketches, mainly images with low descriptive content, BOW produces misclassification rates up to 22.5%.

For the specific task of classifying the map image space as spanned by the sub-classes described in this work, both approaches can become the backbone of helpful applications, especially if the map search is comparable to looking for a needle in a haystack. A CNN provides an immense learning capacity and can be trained on a huge class and sub-class spectrum. BOW can become a useful alternative, whenever training data is rare.

References

Harris, Z. S. (1954). Distributional structure. In *Word*. New York: Linguistic Circle of New York.

Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. In *Proceedings of the 10th European Conference on Machine Learning* (ECML-98), 137–142.

CHAPTER 1:
Introduction

1.1 Background

This section discusses the problem of distinguishing two-dimensional map images from other image types as a result of a tremendous increase in today's information data. The impact of the human visual system on this increase of data is illuminated and the field of computer vision with its efforts to reproduce the human's sensory abilities for machine applications is introduced. Finally, the meaning of the term *map* is defined because of its specific interest for this work.

The digital era has led to an unprecedented increase in the amount of information available. Interpreting and analyzing this data has become a challenge to the individuals, companies, and institutions who could benefit. Where human abilities are overwhelmed, machines must assist in the process. The proliferation of image data is a special challenge. *Data forensics*, in particular, is overly reliant on manpower for image classification and interpretation, and seeks machine solutions. Image processing and understanding have been classic tasks in the field of computer vision, and much progress has been made in recent years. Research has come closer to bridging the semantic gap (Dorai & Venkatesh, 2003) between low-level image features like color and shape and the human visual system by improving image features and methods. This thesis addresses the specific problem of distinguishing two-dimensional map images from other image content by examining two state-of-the-art methods: *Convolutional Neural Networks (CNNs)* and the *Bag of Words (BOW)* method. The results support the idea of implementing CNNs and BOW as the backbone of a computer-based classification application, but they also point to weaknesses and necessary additional research.

The term *information explosion* was introduced by Maron (1961) to describe the increasing spread of published data. Over the past decades, digital data has begun to replace–or at least to complete–traditional systems of information exchange and storage like books, photographic films, and sound and video tapes. The new possibilities of generating digital

data worldwide, instantaneously, and without special training through the use of multimedia devices has led to an unprecedented, emerging documentation of mankind's action and the universal environment (PennyStocks.la, 2015; Sweeney, 2001). This tremendous increase in the volume of available digital data makes information detection, classification, and evaluation a costly and time-consuming process.

This massive growth of data could not only be seen as a problem for today's standard computer user, for whom it might be annoying to waste time searching for a three-week-old social network message or a selfie made last year. It is even more of a challenge for companies, organizations, and institutions that rely on some kind of data analysis. Facing their user's expectations of the outcome of search engines or legal requirements for provided data content, their failure could mean the loss of money or have even more serious consequences. If the security of individuals, equipment, infrastructure, or even organizations and social systems depends on critical information, finding indications of sensitive intelligence in digital data could lead to saving lives. Analyzing the data to simply determine whether the needed information is available can easily become a challenging task. For example, it can take two weeks to plan and conduct a drug search. But it could take an analyst a month to inspect terabytes of confiscated digital data, with no guarantee that the data has been interpreted correctly. This opens up discussion about a reliable automated computational alternative for data inspection.

Seeing is an essential skill for most creatures (Horridge, 1987) and it is the most important human sense for information gathering (San Roque et al., 2015). Therefore, it is not surprising that an essential part of the mentioned increase of digital data is represented by visual data, as shown in Figure 1.1. This kind of data is easily stored and shared via the Internet and produced by using traditional digital cameras or cameras integrated in smartphones, tablets, and computers. With respect to the human ability to see, learning by and training through the visual observation of one's environment are lifelong processes, given healthy reception and mental skills. The perception system allows a person to filter unimportant content and to interpret the remaining information by identifying objects and analyzing their relations and behavior. In addition, it allows the prediction of the future of the observed system, all in the context of the observer's position and actions. This process does not rely on an actual, real-world experience, but can also be realized by looking at a

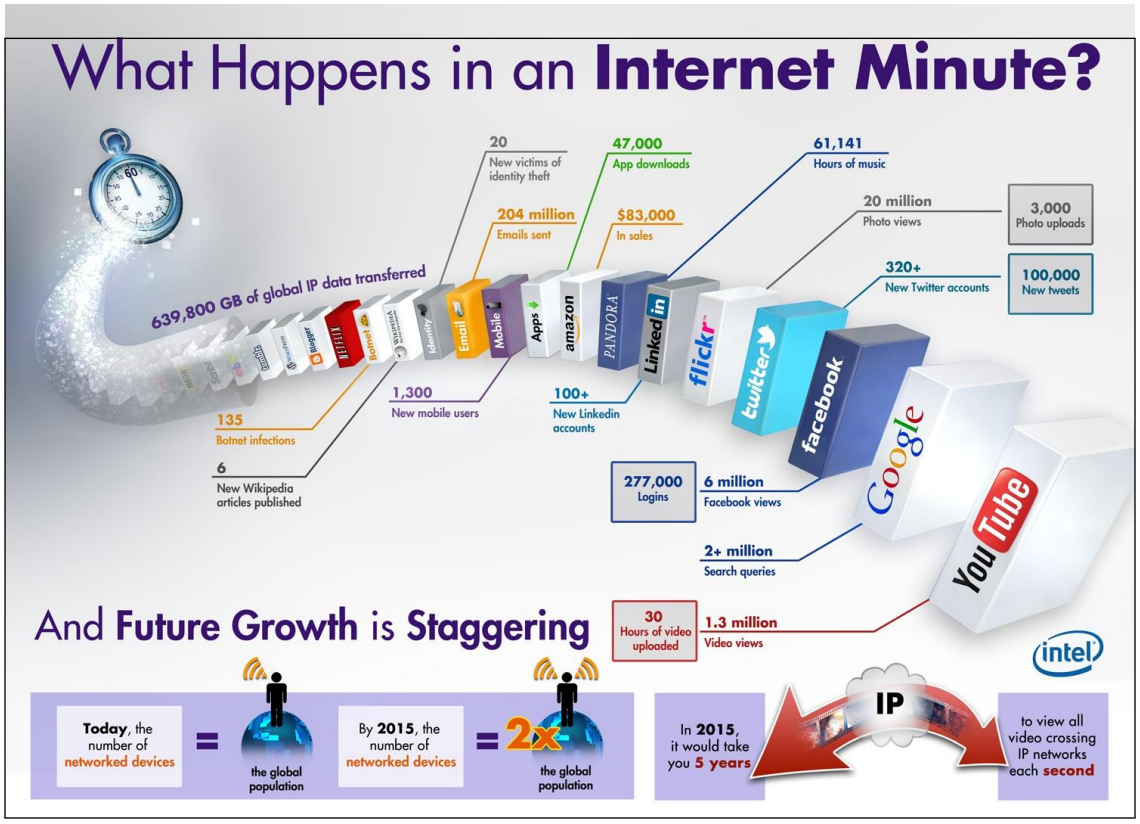

Figure 1.1: Minute-by-minute Internet data traffic from December 2013 showing the significant portion of image and video data transfers via Youtube, Facebook, and Flickr as primary platforms from Infographic (2015)

digital image. The proverb "a picture is worth a thousand words" points to the amount of information a human being can extract from a single visual scene.

It is the goal of the field of computer vision to reproduce, with the exception of predicting the future, the sensory abilities described here by, for example, adapting scientific findings about human perception from fields like psychology or art (Machajdik & Hanbury, 2010) to mathematical algorithms and computational methods in the context of artificial receptors. Reliable solutions exist for individual problems in the field of machine vision (Yammen & Muneesawang, 2014), but failures are still an existing challenge for computer vision systems as discussed, for example, by Zhang et al. (2014). Therefore, an universal "artificial image analyst" seems to be still beyond the horizon of even state-of-the-art methods. Facing the problem of the data expansion described previously, it is essential to invent new methods as shown in Divvala et al. (2014) and to improve existing ones, to provide the nec-

essary automated computational solutions for existing large image collections (see, e.g., in Szeliski (2010), pp. 719-720, table 14.1, “Image databases for recognition” and table 14.2, “Image databases for detection and localization”) .

The word *map* is an umbrella term for an abstract representation of a specific spatial alignment of objects or data. The usage of this word can be found in several fields of science and areas of life. It is not limited to something drawn but can also be represented by words, as shown in Figure 1.2. The meaning of a specific map is defined by its thematic context, the most common being the geography of the Earth with respect to the physical characteristics, especially the surface features, of an area. But maps also exist for the human body, star systems, the Moon, business relations, sociology, religious distributions or historical events, and processes like battles and wars. The variety of maps is remarkable (About.com, 2015) and they have served mankind for thousands of years (Wolkenhauer, 1895) to do what they were originally invented for: to be helpful tools for orientation and planning.

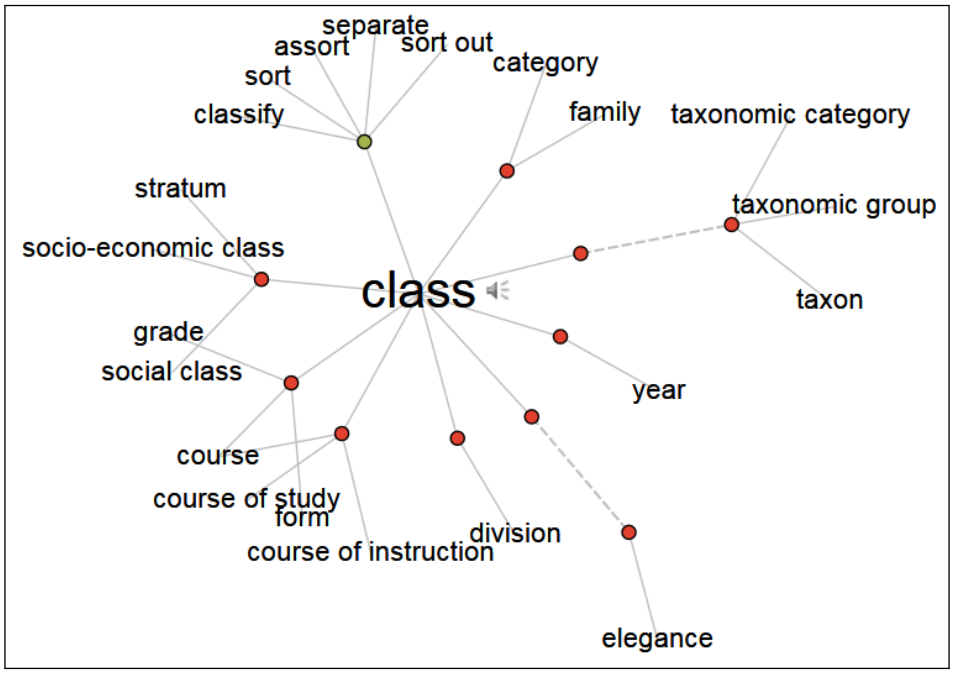

Figure 1.2: A map of words, showing related words to the initial term “class.” Visual Thesaurus connects over 145,000 English words and 115,000 meanings. Clicking on a related word brings this word to the new center of the graph and produces the specific relations. Infographic, D.(2015)

1.2 Scope of This Thesis

The research objective of this thesis is the performance evaluation of two automated computer-based methods given the task of classifying *geographical map images* in a dataset. This work limits geographical map images to content defined by topographical features of a given region. Two state-of-the-art recognition and classification approaches are examined, BOW and CNN. This work is focused on recall and precision as a performance parameters and therefore, reliable evidence about the image content is more important than computational speed.

This thesis excludes all map-related sub-classes that have no geographical reference (e.g., fantasy maps from games) or that focus on very specific content like weather or industry. The training and evaluation dataset is created based on publicly available data from digital sources and with respect to the relevance of a sub-class in accordance with the research objective in context of data forensics. By defining the relevant map sub-classes, collecting the data, and evaluating the two classification methods on the chosen dataset, this thesis considers the following research questions:

- What kinds of maps are relevant to data forensics?
- What sub-classes of maps must be contained in the dataset?
- Are there any special types of sub-classes that significantly weaken a classifier in its performance?
- Which of the two chosen classification methods performs better with respect to recall and precision?
- What factors influence the detection outcome?
- Can the chosen methods become useful applications in the field of data forensics?
- What challenges were found and what is their impact on future work?

The presented work is ordered as follows: Chapter 2 places this thesis in respect to past and related work. Chapter 3 explains the composition of the dataset and delivers insights into the chosen classification methods. Chapter 4 presents the classification results and discusses the influence of parameters and possible challenges. Finally, conclusions are provided in Chapter 5, which also offers ideas for future work.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 2:
Literature Review

A tremendous amount of literature exists focused on image classification tasks, often in the context of object detection and recognition. Without claiming to be comprehensive, this chapter briefly discusses the evolution of this literature, which is closely connected to improvements in the fields of computer science, pattern recognition, and machine learning. After introducing the image-interpreting work with geographical references, the literature related to CNNs and BOW is reviewed. Finally, the work on the underlying algorithms for interest point detection and feature description is discussed.

From the 1970s up to the early 2000s, image retrieval–the reliable access to images in large databases–was the driving force behind the development of methods for mapping images to classes. Text-based image retrieval approaches (Chang & Hsu, 1992; Kato, 1992) refer to image classes by manually or even semi-supervised automatically annotated labels (Guillaumin et al., 2010). By contrast, content-based image retrieval is focused on low-level or higher-level features to gain the necessary classification information (Rui et al. (1999); Vailaya et al. (2001); Serrano et al. (2004)). Jain & Vailaya (1996) introduce an "image retrieval system which is insensitive to large variations in image scale, rotation, and translation" (p. 3) for 400 trademark images. They can clearly show that a combination of color histograms and normalized histograms of edge directions gives a better retrieval accuracy than the individual usage of these methods. Deb & Zhang (2004) discuss the development and future challenges in this research field and give an overview about several content-based image retrieval systems.

The majority of geography-related research of the past decades addresses geographic pattern recognition tasks based on low-level image features like shape and geometric relations (Baltsavias, 2004; Jang et al., 1997; Janssen et al., 1993). Ganpatrao & Ghosh (2014) give a remarkable summary of previous work in this field. For their own approach of symbol and toponym (place name) extraction out of scanned maps, the authors use k-means (Zhang et al., 1997) for color segmentation followed by outline detection and shape matching to recognize letters, symbols, and numbers. As opposed to detection and recognition tasks of

map objects within a map image, this thesis addresses the classification of an image into map or not map. Low-level approaches as mentioned above are often tailored to very specific classification tasks. For example, Vailaya et al. (1998) try to classify city images and landscapes by selecting features like color coherence vectors or edge direction histograms based on their discriminative power. The highest accuracy is reached by combinations of color and shape features whereas they observe that the best feature combinations differ between city images and landscapes.

The increasing amount and complexity of data motivates the need for research to bridge the semantic gap between low-level image features and image content interpretation by humans. Although focused on image retrieval, Liu et al. (2007) provide a good idea of the bridging task and its challenges. A significant number of scientific papers shows the steady increase of performance in precision and speed over the last 15 years, especially solving object classification tasks as demonstrated by Li et al. (2010) or Bilen et al. (2014). Karpathy & Fei-Fei (2014), for example, find a way to connect segments of sentences with image regions that they describe. Using several instances of neural networks, they are able to generate detailed descriptions of classified objects like *wooden office desk*.

With CNN and BOW, two promising methods are chosen to examine their robustness on the map classification task. A commonality of the two methods is their so-called *global* approach, which means that a single feature representation instance is the input to the classifier. Even though BOW starts with local features, they are combined into a single, global histogram of "visual word" frequencies as the image's feature representation. By contrast, a *component-based* approach unites several local elements using a model defining their relationship as in Felzenszwalb et al. (2010).

Although they were addressing the task of face recognition, Heisele et al. (2003) provide a good comparison of these ideas in context of detection followed by recognition. Based on *support vector machines* (SVM), which were introduced by Vapnik & Chervonenkis (1964), their two-level component-based approach detects 14 different parts of a face like eyes or the mouth independently on the first level and then detects the whole face by using a geometrical configuration classifier on the second level. The face recognition classifier is finally trained on single feature vectors build out of the ten normalized parts of each training image. Heisele et al. also introduce two global approaches, where in a first stage both

are trained on 3-D head models. They generate "2,457 face images of size 58x58 pixels" (p. 10) by rotating the heads "between -30 degree and 30 degree in depth" (p. 10). During detection a "sliding window" approach is used; a window of size 58x58 pixels slides over a pyramid of preprocessed images. Two SVMs are then trained for the recognition task on images of individuals, whereas the second approach is introduced to address challenges with intra-class variations caused by changes in the head pose. They show that the component-based approach outperforms the global methods.

2.1 Classification Methods: CNNs and BOW

The first of the chosen methods for this thesis, BOW, was originally developed for text classification problems. By counting the occurrence of words in a document, the resulting vocabulary histogram can give significant evidence about the content. Csurka et al. (2004) introduce with *bag of keypoints* a generic approach for a visual vocabulary. After transferring affine regions around interest points to circular areas for normalization, scale invariant descriptors are computed on these regions. The visual vocabulary is then constructed by clustering these descriptors. The resulting classifier uses histograms of this visual vocabulary, computed for each trained class, to compare them to the histogram generated in the same fashion from the test image. Bag of keypoints is also known as *bag of features* or, with respect to its origin, often called *bag of words*, as in this work. It is widely used in computer vision today (Jégou et al., 2010; Nowak et al., 2006; Yang et al., 2007).

CNNs, as the second method examined, found its access to computer applications during the early 1990s for tasks like word and character recognition (Bengio et al., 1994; LeCun & Bengio, 1994), closely followed by its introduction to image content detection and classification (LeCun & Bengio, 1995; Nowlan & Platt, 1995). After recognizing and examining the usability of CNNs, especially for face detection and recognition tasks (Garcia & Delakis, 2004; Lawrence et al., 1997), CNNs came into research focus with the rise of huge image datasets like ImageNet with millions of pictures (Krizhevsky et al., 2012). Details about the structure of CNNs and the deep learning process are discussed by Arel et al. (2010).

2.2 Interest Point Detectors and Feature Descriptors

An elementary step for a proper learning and classification process is the selection of interest points and their descriptions, a common task for humans and computer based approaches. The key is to find a method that is robust against variations which allows a focus on similar characteristics during the classification step, given (visual) data of the same class.

Scale Invariant Feature Transform (SIFT) (Lowe, 1999) finds key locations (interest points) at local maxima and minima from the difference-of-Gaussians in scale space. Therefore an image is smoothed twice with the Gaussian function and the result of each smoothing iteration is saved. After calculating the difference of the second result from the first smoothing outcome, maxima and minima are detected by comparing each pixel with its eight neighbors. Scale invariance is achieved by rejecting key locations that cannot transport this maxima or minima property to the next lower or upper level of the image pyramid. The orientation of the remaining interest points is "determined by the peak in a histogram of local image gradient orientations" (p. 3) and robustness against intensity changes is achieved "by thresholding the gradient magnitudes at a value of 0.1 times the maximum possible gradient value" (p. 3). A feature vector is then generated to describe the area around the key locations by gradient orientation histograms of eight orientation planes.

In their *Oriented Fast and Rotated BRIEF (ORB)*, Rublee et al. (2011) present a method for interest point detection and description with low computational costs and quality properties like SIFT. Their approach starts with detecting *Features from Accelerated Segment Test (FAST)* points (Rosten & Drummond, 2005) at each level of the image pyramid by generating the intensity threshold between the center pixel and the pixels around the center. An orientation is generated by the offset of a corner's intensity to its, center, the so-called *intensity centroid* (Rosin, 1999). The resulting detector is called *oFAST* (oriented FAST). *Binary Robust Independent Elementary Features (BRIEF)* (Calonder et al., 2010) is used as a descriptor and, by constructing a steered version (*steered BRIEF*), allowed to be invariant to in-plane rotations. As steered BRIEF has a lower variance than BRIEF, its discrimination is worse. In addition, the correlation between the binary tests leads to less contribution of these tests to the result. Rublee et al. (2011) use a *greedy search* algorithm "for a set of uncorrelated tests with means near 0.5" (p. 4). This significantly raises the variance and

improves the correlation. Their final descriptor, which integrates this learning method in steered BRIEF, is called *rBRIEF* (rotated BRIEF).

With *Binary Robust Invariant Scalable Keypoints (BRISK)* (Binary Robust Invariant Scalable Keypoints), Leutenegger et al. (2011) claim to bridge the gap between quality with respect to invariance of the detector to image transformations and distinctiveness of the descriptor and the computational speed needed for real-time applications. An interest point or keypoint is identified by analyzing the saliency scores of the neighbor pixels in the original scale level "as well as in the immediately-neighboring layers above and below." (Leutenegger et al., 2011, p. 3) Such a potential interest point needs to fulfill a maximum condition. This means that the center pixel is either the brightest or darkest one among its neighbors in the particular layer and within the corresponding patch in the layers above and below (which means that this interest point is in general traceable if the scale for an image changes). The BRISK descriptor is a binary string and built by retrieving gray values from a sampling pattern which is defined by several sampling locations on "circles concentric with the keypoint" (p. 3). The characteristic direction of the interest point is determined by local intensity gradients and the descriptor is finally "assembled via brightness comparison" (p. 4).

Dalal & Triggs (2005) find with *Histograms of Oriented Gradients (HOG)* an innovative descriptor computed on a grid of overlapping cells that are (with the exception of object orientation) invariant to geometric and photometric transformations. Bay et al. (2006) introduce a scale and rotation-invariant descriptor called *Speeded Up Robust Features (SURF)* that combines and simplifies integral images for image convolution with existing state-of-the-art descriptors and detectors. This interest point detector and feature descriptor is the backbone of the image analysis with BOW and explained in more detail in the next chapter.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 3:
Methodology

This chapter illuminates the data collection process in context with the relevance to data forensics. It also describes the experimental setup of the chosen methods for classifying geographic map images in a dataset.

3.1 Dataset

The following section narrows the definition of maps with respect to the research objective. It explains the data collection process for building an image dataset as a foundation for the training and evaluation phases with BOW and CNN.

3.1.1 Definition and Delimitation

The research objective is the performance evaluation of two automated computer-based methods with classifying geographic map images in a dataset. More specifically, this work is interested in classifying those images as maps which contain:

> Detailed, accurate graphic representations of features that appear on the Earth's surface. These features include:
>
> - cultural: roads, buildings, urban development, railways, airports, names of places and geographic features, administrative boundaries, state and international borders, reserves
> - hydrography: lakes, rivers, streams, swamps, coastal flats
> - relief: mountains, valleys, slopes, depressions
> - vegetation: wooded and cleared areas, vineyards and orchards. (Geoscience Australia, 2015)

Geographical maps include a broad spectrum of sub-classes, differentiated by vegetation and climate zones and by variation among surface structures like mountains, plains, and water. This means that different techniques and styles are required to produce readable maps. In addition, there exists no standardized set of rules for map design; therefore,

symbols, geometric shapes, and colors for the same sector of the Earth's surface may differ among the producers of maps. This research's specific interest in data forensics to find sensible intelligence information leads to a focus on map products which are appropriate for orientation and planning tasks. This implies a delimitation of the sub-classes. Excluding sub-classes such as 3-D visualizations, historical maps, nautical charts, or ocean floor maps, this research focusses on four sub-classes (examples of which are shown in Figure 3.1):

- *basic maps*, containing pure topographical information, no specific thematic topic is allowed, no limitations to the kind of terrain
- *pilotage charts*, which can differ from basic maps in particular by added navigation corridors and signs, aerodromes, vertical obstructions and areas of special use airspace
- *web maps*, which are Internet-based map applications, as designed for example by Google Maps (Google, 2015b) or Bing Maps (Microsoft, 2015) and
- *sketches*

Sketches are of special interest in data forensics. This sub-class does not match with the former description of a map as "detailed, accurate graphic representations" and defines no specific rules for the creation process. Usually, sketches condense the amount of available geographic information to a minimum, focusing on the main features to achieve the purpose they are created for. Therefore, a lot of information is missing (e.g., mostly no color, low consistency for shapes of the same objects), and, additionally, unusual information is added, such as drawings of side views or a text passage. But sketches are easy to create with no more than a piece of paper and a pen and can even serve as a helpful planning tool for finding your friend's house–but also for criminals and terrorists.

3.1.2 Data Collection and Dataset Setup

The access to map images for this work is limited to publicly available data from digital sources. The dataset has to provide a proper mix of sub-class members to guarantee training descriptors with robustness to variety within a sub-class. Map images in general are not enthusiastically shared online and some deeper searches are necessary to find the desired data. A huge number of maps representing the defined geographical space is not accessible, and most countries do not provide their official map series online for free access. Often,

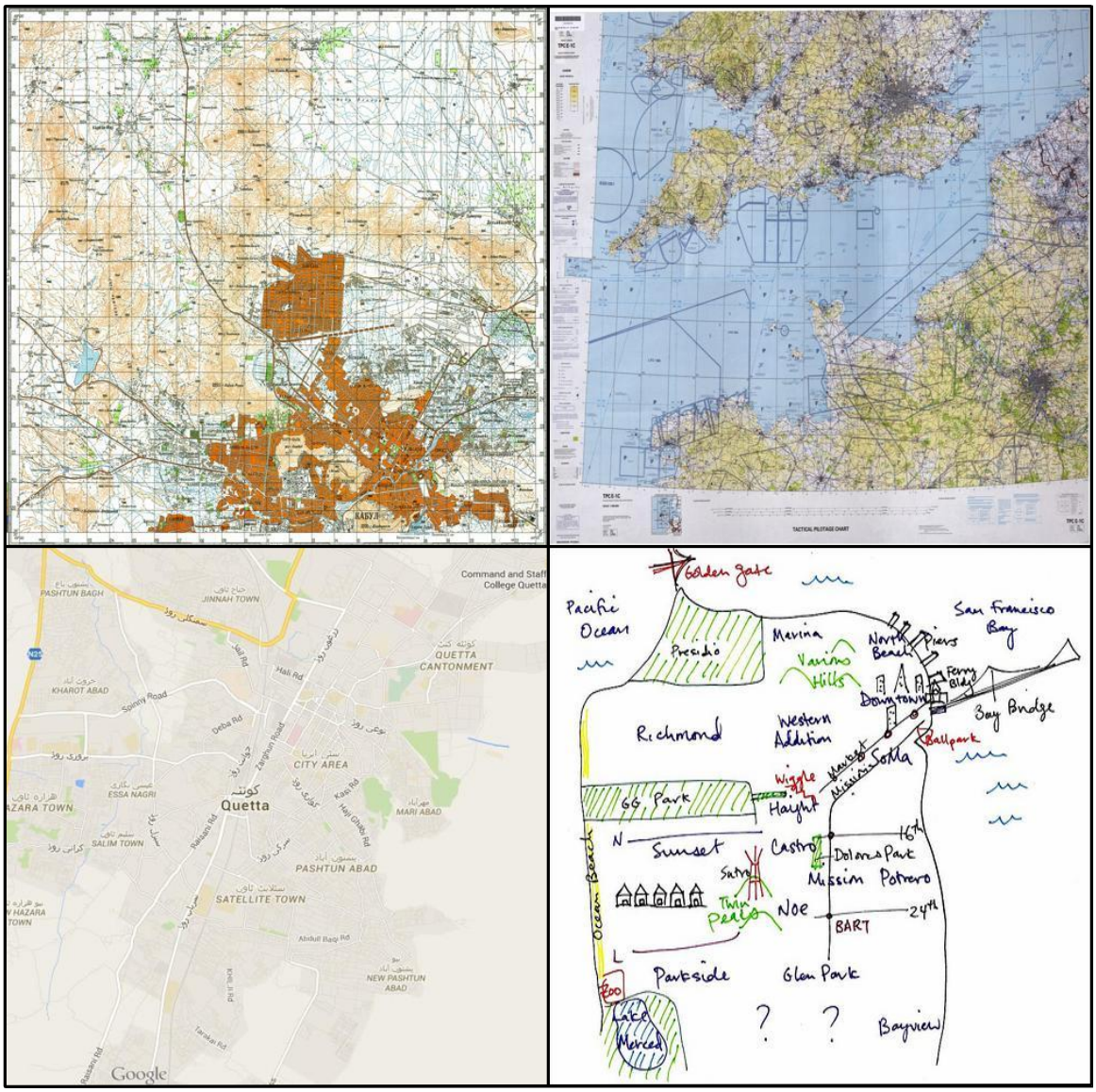

Figure 3.1: Examples of chosen map sub-classes; upper left: *basic maps*, geographical standard map with no additional information, upper right: *pilotage charts*, containing navigation corridors and special air space zones, lower left: *web map*, Internet-based map applications, as designed for example by Google Maps or Bing Maps, lower right: *sketch*, condensed amount of available topographic information.

websites of individuals, libraries, and archives are the only available sources.

In general, it is possible to generate a data collection with tens of thousands of map images, based on publicly available data, given months to search and collect them. Alternatively, this effort could be crowd-sourced as was done for the ImageNet collection which employed thousands of Amazon Mechanical Turk workers (Fei-Fei, 2010). The University of Texas (2015), for example, provides online access to over 40,000 maps; however, many of them are historical or have a specific subject matter such as climate or resources. With a limited amount of time to examine an adequate quantity of maps, the data collection process settled on 1,200 map images called *positives*. The sub-classes are represented as follows:

- 400 basic maps–sources: University of Texas (2015) at Austin and Petro Vlasenko (Vlasenko, 2015)
- 150 pilotage charts–source: University of Texas (2015) at Austin
- 500 web maps–source: Bing Maps (Microsoft, 2015) and Google Maps (Google, 2015b)
- 150 sketches–sources: Google's image search (Google, 2015a), with the following search terms: "area sketch", "geographic sketch," "orientation sketch," "paths sketch," "topographic sketch."

Additionally, images without map content, called *negatives*, are chosen from *ImageNet* (Deng et al., 2009). Although ImageNet provides easy access to roughly 15 million pictures in 22,000 categories, the number of training and evaluation negatives for this research is limited to 1,200 in order to balance the dataset. The selection process partially focuses on categories, which are expected to make the classification process harder because of their geometric properties. The idea is to examine the classification quality in cases of poor map content, like rough sketches.

The final negative set contains a composition of classes as shown in Table 3.1. The complete dataset with 1,200 positives and 1,200 negatives is always divided into two sections: 70% for training purposes (840 negatives and 840 positives) and 30% for the evaluation phase (360 positives and 360 negatives). The impact of hyper-parameter settings is ob-

served on the same dataset split for training and evaluation. A shuffle of the dataset gives evidence about the variability of the effectiveness via repetition, given the specific classification approach with a fixed hyper-parameter setting.

ImageNet Index	Description	Number Images
n02823848	beer hall	50
n03010473	chassis	50
n0352415	hockey stick	50
n03509843	heat-seeking missile	50
n03171635	defibrillator	50
n07863374	pasta	50
n07862244	bacon and eggs	50
n07840219	caper sauce	50
n07954211	book, rule book	50
n13354021	pocketbook	50
n13863771	line	50
n13864153	convex shape, convexity	50
n13865298	cylinder	50
n13874073	equator	50
n13869788	envelope	50
n13876561	helix, spiral	50
n14785065	bricks and mortar	50
n14820180	concrete	50
n14844693	soil, dirt	50
n14915184	ice, water ice	50
n14975598	papyrus	50
n09287968	geological formation, formation	50
n09344324	lunar crater	50
n09302616	highland, upland	50

Table 3.1: Chosen categories from ImageNet to create the negative set. Categories like *line* or *convex shape* were selected because of their assumed closeness to objects occurring in maps and their possible influence on the classification outcome. None of these images shows a geographic map.

3.2 Experiments

This section describes the experimental setup and the process of evaluation of the chosen BOW and CNN approaches. The detailed work flow is presented for both methods.

3.2.1 Process and Experimental Design–BOW

This subsection describes the generation of a bag of visual words as a basis for the experiments. Four computational steps precede the classification on the evaluation set. Interest points have to be found in the training set images, and descriptors have to be calculated on these interest points. The descriptors have to be clustered to form a bag of visual words and, with these visual words, a classifier has to be trained for each image class based on the frequency of visual words in the corresponding labeled training images. Two available implementations, one in MATLAB and the other one in the *Computer Vision Algorithm Collection (CVAC)*, are examined by this work.

BOW with MATLAB

MATLAB's CV toolbox (MATLAB, 2014) contains a *bagOfFeatures* function. It is parameterized by:

- the dictionary size (number of visual words)
- the selection method for interest points (either by a detector or by using a grid step option)
- an option for defining a fixed grid step size
- the orientation of the SURF feature descriptor, which can be fixed upright (so called U-SURF) or estimated on the feature vector ("rotation-invariant") and
- the sizes of areas (block width) to calculate U-SURF feature descriptors on a multi-scale.

The exact experimental setup for training and classification runs with MATLAB can be seen in Table 3.2. The dataset is examined with different dictionary sizes. From the two point selection methods, only the grid step option is used for the examination of BOW as the alternative SURF Detector did not produce usable outcomes on the dataset. The grid step option has to be specified by a step size within an image (default setting [8 8]), which reflects the distance in the x and y direction between horizontal and vertical grid lines. The

intersections of this lines are set as interest points and descriptors are calculated on these intersections. If not explicitly changed by a user, rotation-variant U-SURF descriptors are used. Even a rotation of the maps in the images is not expected, this rotation-variance of the descriptor can effect the visual vocabulary. Local image artifacts such as road intersections, houses, contour lines etc can occur in any orientation and, by chance, on grid intersections which define interest points. As one would need only one rotation-invariant descriptor for all orientations of, e.g., similar-looking road intersections, using U-SURF would lead to many variant descriptors to match the rotated intersections. So a bigger visual vocabulary is needed to describe road intersections. As the dictionary size is fix, a rotation-invariant descriptor would allow more visual words to be used for descriptors of other interest points.

When using the selection method "Detector" for interest points, these points are detected by using differently scaled box filters which approximate the Gaussian second order derivatives of the covered image region. In MATLAB, a threshold and the filter sizes can be defined to decide about the occurrence of an interest point. The calculation of a SURF descriptor starts after extracting a reproducible orientation around such an interest point. The dominant orientation can be determined by the response of *Haar wavelets* (Haar, 1910) in x and y direction in a circular area (radius 6 times the interest point detection scale) around the interest point (Bay et al., 2006). This finally defines the orientation of the square region that the descriptor is calculated on. The grid step option with U-SURF avoids these pre-processing steps.

Once descriptors are calculated for all interest points, *k-means* (Lloyd, 1982) is used to cluster them with respect to the defined amount of visual words. K-means is an iterative clustering process that minimizes the sum of distances between all objects and their cluster centroids (measured in squared Euclidean metric) over all cluster centers. It starts with k cluster centers (where k represents the defined dictionary size) at random positions and allocates each object to its nearest cluster center. After this, the cluster centroid is recalculated with respect to the cluster members. Allocation and recalculation of cluster centroids is repeated until the sum of all distances is minimized or the defined maximum number of iterations is reached.

After the construction of this visual vocabulary, a *Support Vector Machine (SVM)* classifier for each image class is trained on the labeled training data using the feature vector of each

image. Simply stated the occurrence of visual words out of the just-clustered bag of words in a training image of a specific class is observed and used for the training of the related classifier. Once the classifiers are trained, the evaluation set can be classified.

MATLAB was used to generate confusion tables for the validation of a user-defined percentage of training data, just as for the classification of the evaluation set. It is expanded by a routine that writes, in a case of misclassification, the name of the image, its original class, and the classification decision into a separate file. This allows an examination of the probable reasons for the specific misclassification.

Dictionary Size	Grid Step Size	Number of Classifiers	Comment
25, 50, 100, 250, 500, 1,000	[16 16], [32 32], [64 64]	2	
250	[64 64]	2	10 runs with image shuffling between training set and evaluation set
500, 1,000	[16 16], [32 32], [64 64]	5	

Table 3.2: Experimental setup for 40 separate runs to examine MATLAB's BOW implementation for classifying map images. The number of classifiers determines whether the training and testing is conducted on two classes (positives and negatives) or the negatives class with four map subclasses.

BOW with CVAC

The second BOW implementation is accessible via the NPSVisionLab (2015) of the Naval Postgraduate School (NPS). The *Easy Computer Vision Project (EasyCV)* "provides access to algorithms in the CVAC through well-defined interfaces, it links annotation tools (LabelMe [Torralba et al. (2010)], VATIC [Vondrick et al. (2013)]) to algorithms, and it permits creation of new detectors and their performance evaluation." (NPSVisionLab, 2015) In particular, its linkage to libraries like *OpenCV* (Bradski, 2000), allows uncomplicated setup switches with respect to algorithms and methods.

CVAC contains a *Python* (Python Software Foundation, 2014) BOW demo script which is used for this work and only has to be adapted to the collected dataset directory. The provided code is changed with respect to the detector and descriptor settings and completed with an evaluation and storage procedure for the results. To reduce the variability in the results for a specific dictionary size, nine repetitions of BOW are run with shuffled data, using ORB for interest point detection and description with the following dictionary sizes: 5, 10, 25, 50, 100, 250, 500, and 1,000.

Additionally, BOW is evaluated with different detector/descriptor combinations; Table 3.3 provides an overview of the hyper parameter space. Because of time constraints, these runs are performed on a reduced dataset with 250 visual words, 300 positives, 300 negatives, and 180 test images, and the explored combinations are not exhaustive as EasyCV offers other detectors and descriptors, too. It is, for example, possible to combine all detectors with a grid option that searches for interest points in defined grids. This method also allows a researcher to limit the number of returned features.

Interest Point Detector	Descriptor
ORB, SURF, SIFT, BRISK	ORB, SURF, SIFT, BRISK

Table 3.3: Overview of the hyper-parameter space of detector-descriptor constellations for map image classification with BOW. Training and testing is conducted on a reduced dataset with two classes (250 visual words, 300 positives, 300 negatives, and 180 test images).

3.2.2 Process and Experimental Design–CNN

This subsection describes a classification approach based on a pre-trained deep CNN. The so-called *BVLC Reference CaffeNet* is built with the *Caffe deep learning framework* (Jia et al., 2014) provided by the Berkeley Vision and Learning Center (BVLC). For this work, the deep CNN is additionally trained with transfer learning on the introduced positives and negatives and then tested on the evaluation set using the EasyCV library.

BVLC Reference CaffeNet is trained on the Large Scale Visual Recognition Challenge 2012 training set (*ILSVRC 2012*), (Russakovsky et al., 2015)) which includes 1,000 categories with 1.2 million images. The ILSVRC 2012 data is a part of ImageNet, an image database which contains roughly 15 million images in 22,000 categories.

Besides two minor differences, BVLC Reference CaffeNet is similar in its architecture to the deep CNN trained by Krizhevsky et al. (2012) on *ILSVRC 2010*. It consists of eight weighted layers where layers one to five are convolutional and the last three layers are fully connected. Layers one, two, and five are followed by max-pooling layers, and the max-pooling layers after layer one and layer two are followed by response-normalization layers. This is different than the network of Krizhevsky et al. (2012) where pooling is done after the response-normalization layers. Finally a 1,000-way softmax distributes its input over the 1,000 categories. A special characteristic of this deep CNN is that two GPUs share the work. They run on different parts of the layers and only communicate with each other between specific layers (three to four, so four gets all input from both parts of layer three, and between all fully connected layers). Figure 3.2 shows this architecture in detail. The resulting network consists of 60 million parameters and 650,000 neurons.

By enlarging the dataset with label-preserving transformations, Krizhevsky et al. (2012) try to reduce the over-fitting. On the one hand, they generate image translations by randomly extracting 1,028 224x224 pixel patches out of the down-sampled 256x256 pixels training set images and their horizontal reflections. The training set is therefore expanded by factor 2,048. On the other hand, Krizhevsky et al. (2012) try to combat over-fitting by altering the intensities of the RGB channels in the training images. This is the second difference from the BVLC Reference CaffeNet as this deep CNN is trained without the relighting data augmentation. The publicly available model is a snapshot at iteration 310,000, whereas the

best performance was reported by iteration 313,000.

This deep CNN is additionally trained (transfer learning) on the 840 positives and 840 negatives; 20% of the samples are randomly chosen and used for validation after training. The training images are downsized to 256x256 pixels from which 227x227 pixel patches are cropped, with or without reflection. The hyper parameters for the transfer learning are adjusted as follows in comparison to the original BVLC Reference CaffeNet settings: the learning rate is reduced from 0.01 to 0.001, and the maximum number of iterations is reduced from 450,000 to 1,000. The weight decay (0.0005) and the step size (100,000) are left unchanged. All convolutional and all fully connected layers are involved in the transfer learning. Table 3.4 gives an overview of the conducted evaluation tasks using BVLC Reference CaffeNet. CVAC contains a Python Caffe demo file, which is modified to train and evaluate on the collected dataset. The result is stored as a list of images by name, either classified as 0 for positive or 1 for negative. The unique ImageNet naming convention allows the quick parsing of false positives and false negatives.

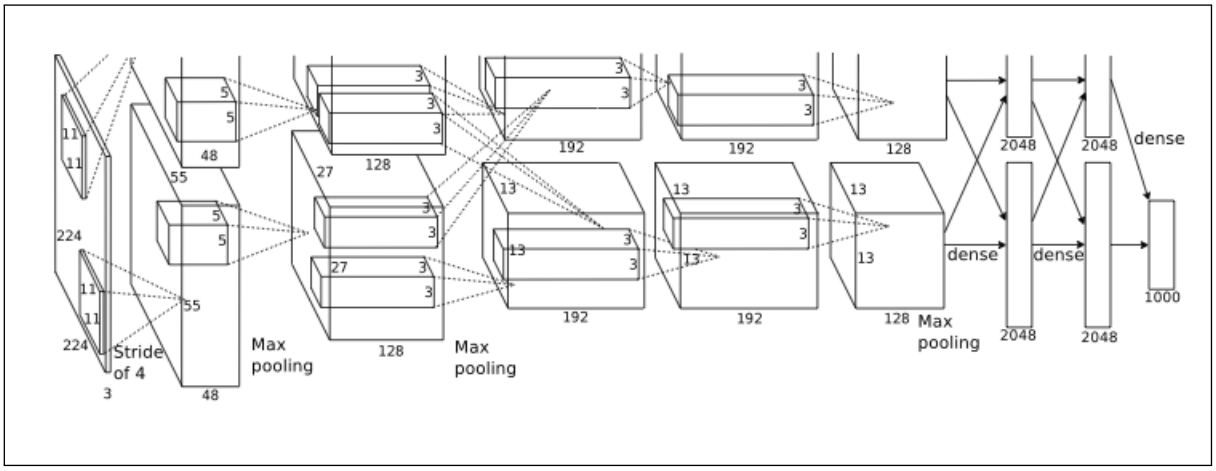

Figure 3.2: Architecture of a deep CNN by (Krizhevsky et al., 2012), with five convolutional and three fully connected layers. Two GPUs work on separate parts of the layers and communicate only after layer three and after each fully connected layer.

Number of Classifiers	Expansion of training data by reflection of 227x277 patches?
2	NO
2	YES
5	NO
5	YES

Table 3.4: Experimental setup for four runs to examine the deep CNN BVLC Reference CaffeNet. The 227x227 pixels patches are generated out of the down-sized 256x256 pixels input images.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 4:
Evaluation of the Results

This chapter introduces and discusses the results of the experiments with BOW and the CNN. It shows, for both methods, remarkable results, either in terms of recall (the proportion of classified true positives in comparison to all positives) and precision (the proportion of classified true positives in comparison to all classified positives) for classification tasks with two classes, or as percentages of correctly and incorrectly classified images for tasks with sub-classes. It can be shown that BOW can become a robust solution when limited training data allows the generation of strong descriptors. All computation was done on the NPS Hamming cluster computer.

4.1 BOW with MATLAB and Two Classes: Positives and Negatives

This section examines the BOW results of the MATLAB-based approach with positives and negatives. It shows that the performance does not require a large dictionary. For simplification, an experiment with, for example, 250 visual words with grid step option and step size [16 16] is written as 250/16.

4.1.1 Results

The results of 18 experiments are presented as recall and precision in Table 4.1. The 18 parameter configurations are generated by varying the distance between grid intersection (grid step-size) from 16 over 32 to 64 pixels for each dictionary size between 25 and 1,000. The lowest recall with 91.94% occurs for 25/64, and the highest recall with 99.17% is observable for the runs 100/16, 250/16, 500/16, and 1,000/16. Run 100/16 shows the highest precision (97.01%) among all parametrizations. Given a constant grid step size, the best precision always occurs on runs with 100 visual words. The recall improves with an increasing dictionary size. An exception is the setup with step size [16 16] where from 100 up to 1,000 visual words, no improvement is observable.

4.1.2 Discussion

The issue of interest for these 18 runs is the fact that the best precision is always reached with 100 visual words. As with an increasing dictionary size the recall either remains at a high level or increases; it is the increasing number of false positives that reduces the precision for larger dictionary size. The model for the classifier becomes more complex if the dictionary size rises. This can lead to overfitting, which means that the classifier becomes fragile for new, unseen data. So, in the presented 18 runs, the model is either not overfitted for positives, or the data from the evaluation set looks pretty similar to the training data. But this is not the case for the negatives, where the increase of visual words of a number higher than 100 leads to an increasing number of misclassifications.

The constant recall of 99.17% over the four highest numbers of visual words seems to be an upper bound for the chosen training set/evaluation set constellation; three images remain misclassified. Just including them in the training set would not necessarily produce a better classifier with a constant 100% recall. Changing the training input means to change the resulting cluster centers. The visual vocabulary would differ in its appearance, which can result in a different classification outcome on the evaluation set. Table 4.2 shows the variability of recall and precision (10 runs with 250 visual words and step size [64 64]), given a different composition of images in the training set and the evaluation set for each run.

	Number of Visual Words					
	25	50	100	250	500	1,000	Step Size
Recall	0.9722	0.9806	0.9917	0.9917	0.9917	0.9917	[16 16]
Precision	0.9162	0.9671	0.9701	0.9597	0.9520	0.9520	
Recall	0.9361	0.9611	0.9611	0.9778	0.9833	0.9889	[32 32]
Precision	0.8963	0.9153	0.9505	0.9337	0.9243	0.9152	
Recall	0.9194	0.9361	0.9694	0.9639	0.9750	0.9861	[64 64]
Precision	0.8922	0.9133	0.9458	0.9455	0.9360	0.9318	

Table 4.1: Results for BOW with MATLAB on two classes: positives and negatives. The results of 18 different computational runs is shown in this table. For these runs the dataset was divided into a fixed training set and evaluation set. Run 100/16 shows the highest recall and also the highest precision. For larger grid steps the recall finds no upper bound with the rising dictionary size; the highest precision is always observable for a bag size of 100 visual words.

	Run 1	Run 2	Run 3	Run 4	Run 5
RECALL	0.9667	0.9750	0.9694	0.9639	0.9694
PRECISION	0.9380	0.9564	0.9562	0.9507	0.9641
	Run 6	Run 7	Run 8	Run 9	Run 10
RECALL	0.9694	0.9639	0.9639	0.9778	0.9694
PRECISION	0.9432	0.9533	0.9378	0.9462	0.9562

Table 4.2: Results for 10 runs with 250 visual words and step size [64 64]. Recall and precision show some variability if training and evaluation are based on different compositions of the images.

4.2 BOW with MATLAB, Four Map Sub-Classes, and Negatives

This section examines the BOW results of the MATLAB-based approach with map sub-classes and negatives, addressing in detail the following research question: Are there any special types of sub-classes that significantly weaken the classifier in its performance? With sketches and web maps, two of such sub-classes are found. The research shows that it is not the number of sub-class members, but rather the content quality of the images that leads to higher misclassification rates.

4.2.1 Results

The classification results of six runs on the Hamming computer cluster is examined with respect to the step sizes [16 16], [32 32], and [64 64] , using 500 and 1,000 visual words. These experiments are performed parallel to the runs with two classes and the corresponding results are not incorporated with respect to the definition of the dictionary size. Correct and incorrect classification percentages are shown in Table 4.3 and Table 4.4. The lowest misclassification rate for basic maps is 4.67%, in the parameter constellation 1,000/32. The highest number of correct classifications can be observed for pilotage charts, independently of the setup. For all six runs it can be observed that the sub-classes sketches and web maps have the highest misclassification rates, pending between 11.11% and 22.5%. The best misclassification rate for negatives is 6.11% for the constellations 500/64, 1000/32, and 1,000/64. Comparing 500 with 1,000 visual words, no approach can be preferred. For example, basic maps shows better results with 1,000 visual words, and pilotage charts performs best with 500 visual words. Additionally, there is no clear choice for a special grid step size. The intuition that a lower grid step size would perform better because it generates more feature descriptors is not supported by the results; web maps, for example, always perform best with step size [64 64].

4.2.2 Discussion

Sketches and web maps are two sub-classes that are found to weaken the performance of the classifier. If the detection of map sub-classes is required by a potential analyst using this method as the backbone of an application, their impact is significant, as these sub-classes

are up to 20 percentage points worse in correct classifications in comparison to the other sub-classes.

Most of the misclassified web maps were classified as sketches. In many cases, the content of sketches is limited to the most important information that has to be communicated via the sketch. The resulting lack of visual features also occurs in web maps with very little geographic information (see Figure 4.1) and is related to geographical density. This density in a map image is a result of map scale and surface objects of the covered area. In a geographical sense, the large scale contains maps in a scale range of 1:1 to 1:600,000 (Scale (map). (n.d.)., 2015). There is definitely sufficient geographic information in a *city map* of scale 1:100,000 or even 1:5,000 but maybe not in a map with scale 1:50. And a scale of 1:5,000 might provide sufficient geographic density for feature extraction in a city map but not in a map showing a desert region. The option of excluding web maps with very little geographic information from the training set brings disadvantages. It would most likely exclude sketches from being classified as web maps, but this would also exclude the mentioned web maps with very little geographic information from this sub-class. A possible way to reach an improvement is by subdividing web maps in additional sub-classes with respect to the geographical density.

The second-worst-performing sub-class is sketches. One could argue that this is related to the low number of training and evaluation images for this sub-class. But the number of training and evaluation images for pilotage charts is as low as for sketches–105 for training and 45 for evaluation. And pilotage charts shows the overall highest percentage of correct classifications. So why is the pilotage charts sub-class so robust and not sketches? There is an observable difference in the previously discussed density between the two sub-classes (see Figure 4.2). In addition, the pilotage charts images have the largest size within the dataset. With up to 17,601x12,741 pixels resolution they represent a total of 2.6 gigabytes of data. The largest sketches image has a resolution of 3,599x3,474 pixels, but all 150 sketch images are represented by a data size of 15.3 megabyte; many of these images have low resolution. Generating interest points with MATLAB's grid method and the step size [64 64], the difference is 294,424 interest points for sketches in comparison to 28,857,392 interest points for pilotage charts. The computation proceeds with the strongest interest points of each sub-class/class but limited for each to an amount of only 80% of the number

of interest point of the sub-class/class with the overall fewest interest points. This means that 235,539 descriptors are calculated for sketches (as sketches is the sub-class with the overall lowest number of interest points). Also, 235,539 descriptors are calculated for pilotage charts.

Limited to the absolutely necessary information, it is the nature of most sketches that the areas for computing the descriptor around a grid-defined interest point often contain marginal changes of brightness or poor shape information. The usage of a grid to generate feature descriptors will generate many such descriptors; they are calculated over areas of constant color intensities like just white or other monotone-colored image areas. For sketches, the classifier is therefore heavily trained on visual words that fit to these areas. So a sketch-responding part of a classifier will finally expect to see descriptors representing monotone areas. This behavior can also be observed by looking at images of the negatives class, misclassified as sketches. The negatives class shows a true positive percentage between 92.78% and 93.89% and a constant portion of 5.56 to 6.67 percentage points misclassification votes for sketches. As opposed to the assumption made during the data collection (ImageNet categories like *line* or *convex shape* have an influence on the evaluation process because of their geometric similarities in comparison to sketches), it is the absence of variance in the areas a descriptor is calculated on, that leads to wrong votes. Some images misclassified as sketches are shown in Figure 4.3. In comparison to sketches, the strongest descriptors for pilotage charts cover much more variance and allow a more distinct description by visual words.

But can the classifier be improved by rejecting sketches from the training? A run was performed with setting 1,000/64; Table 4.5 shows the result. The correct classification rates within the sub-classes show that the improvement is significant. Except pilotage charts, which remains on its high level of correct classifications, every sub-class/class reduces the number of misclassifications. For web maps, the portion of correct classifications increases by 13.33 percentage points. A further observation is that, by training the classifier on one fewer sub-class than before, a better description of the remaining sub-classes/class with the given visual words becomes possible. For example, basic maps is improved in its number of correct classifications not only by assigning the former sketch misclassifications to its own sub-class, but also by identifying former assumed pilotage charts as basic maps. In

terms of accuracy, a rise from 91.80% (best accuracy for 1,000 visual words and step size [32 32]) to 96.59% can be observed. So, rejecting a sub-class or class can be a helpful alternative to improve the classification results. By doing so, one thing must be kept in mind. There is a reason that the excluded sub-class sketches was part of the classifier: somebody wanted to find them.

500/16	Basic Maps	Pilotage Charts	Web Maps	Sketches	Negatives
Basic Maps	**92.67**	4.67	0.00	2.67	0.00
Pilotage Charts	0.00	**97.78**	0.00	2.22	0.00
Web Maps	2.50	4.17	**77.50**	15.83	0.00
Sketches	4.44	6.67	11.11	**77.78**	0.00
Negatives	0.28	0.00	0.56	6.11	**93.06**
500/32	Basic Maps	Pilotage Charts	Web Maps	Sketches	Negatives
Basic Maps	**92.00**	5.33	0.00	2.67	0.00
Pilotage Charts	0.00	**97.78**	0.00	2.22	0.00
Web Maps	0.00	3.33	**80.83**	15.83	0.00
Sketches	4.44	6.67	6.67	**82.22**	0.00
Negatives	0.28	0.00	0.28	6.67	**92.78**
500/64	Basic Maps	Pilotage Charts	Web Maps	Sketches	Negatives
Basic Maps	**93.33**	4.67	0.00	2.00	0.00
Pilotage Charts	0.00	**100.00**	0.00	0.00	0.00
Web Maps	0.83	4.17	**82.50**	12.50	0.00
Sketches	11.11	2.22	4.44	**82.22**	0.00
Negatives	0.00	0.00	0.00	6.11	**93.89**

Table 4.3: Confusion matrices for BOW on four sub-classes and the negatives class, computed on 500 visual words with step sizes [16 16], [32 32], and [64 64]. Web maps and sketches are the weak sub-classes. Although pilotage charts has the same small number of training images as sketches, its performance is much better, based on the information available in the images.

1000/16	Basic Maps	Pilotage Charts	Web Maps	Sketches	Negatives
Basic Maps	**94.67**	2.67	0.00	2.67	0.00
Pilotage Charts	0.00	**97.78**	0.00	2.22	0.00
Web Maps	0.83	0.83	**79.17**	19.17	0.00
Sketches	4.44	2.22	4.44	**88.89**	0.00
Negatives	0.00	0.00	0.56	6.11	**93.33**
1000/32	Basic Maps	Pilotage Charts	Web Maps	Sketches	Negatives
Basic Maps	**95.33**	2.00	0.00	2.67	0.00
Pilotage Charts	0.00	**97.78**	0.00	2.22	0.00
Web Maps	0.83	2.50	**80.83**	15.83	0.00
Sketches	4.44	4.44	4.44	**86.67**	0.00
Negatives	0.00	0.00	0.28	5.83	**93.89**
1000/64	Basic Maps	Pilotage Charts	Web Maps	Sketches	Negatives
Basic Maps	**93.33**	4.00	0.00	2.67	0.00
Pilotage Charts	0.00	**97.78**	0.00	2.22	0.00
Web Maps	0.83	2.50	**82.50**	14.17	0.00
Sketches	8.89	4.44	4.44	**82.22**	0.00
Negatives	0.00	0.00	0.56	5.56	**93.89**

Table 4.4: Confusion matrices of BOW on four sub-classes and the negatives class, computed on 1,000 visual words with step sizes [16 16], [32 32], and [64 64]. The same observations can be made as for the results with 500 visual words.

1000/16	Basic Maps	Pilotage Charts	Web Maps	Negatives
Basic Maps	**99.33**	0.67	0.00	0.00
Web Maps	0.00	**97.78**	2.22	0.00
Pilotage Charts	1.67	2.50	**95.83**	0.00
Negatives	0.56	0.28	3.61	**95.56**

Table 4.5: Confusion matrix of a run without the sub-class sketches.

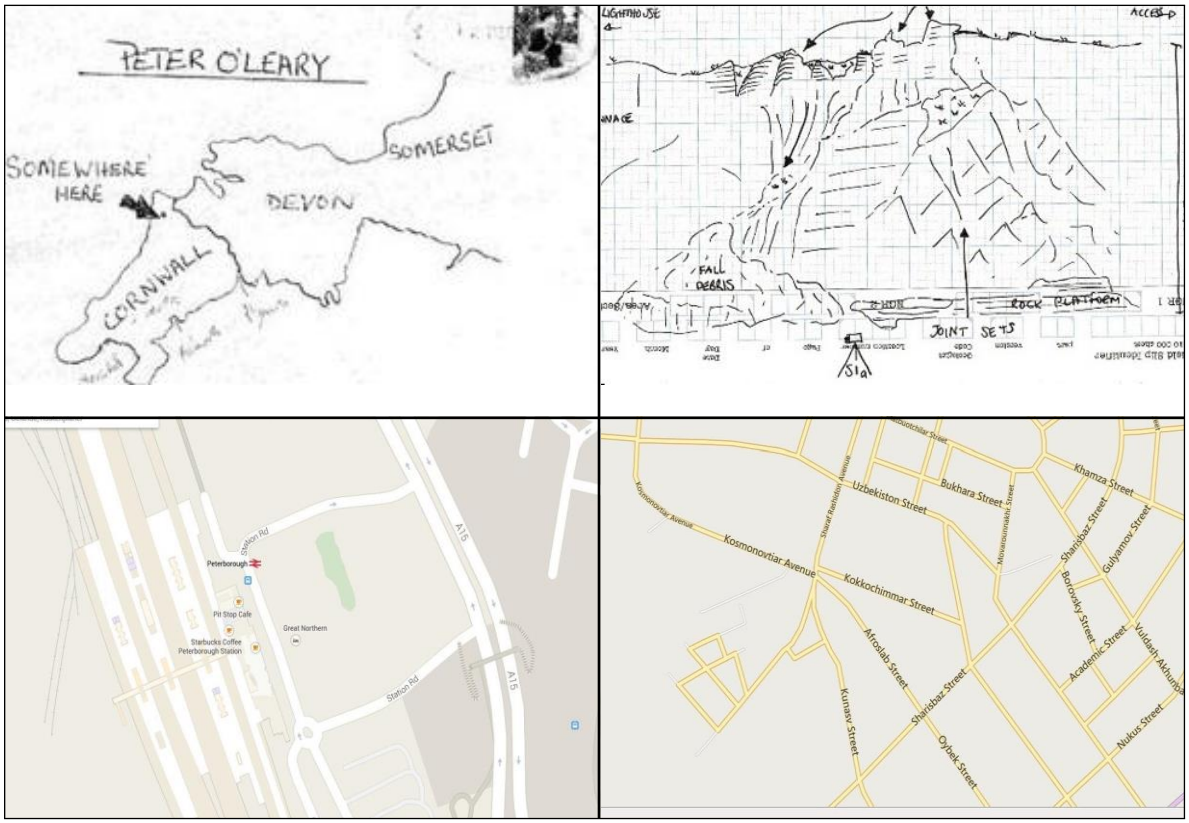

Figure 4.1: Examples of misclassifications. The upper sketches are classified as web maps; the lower web maps are classified as sketches.

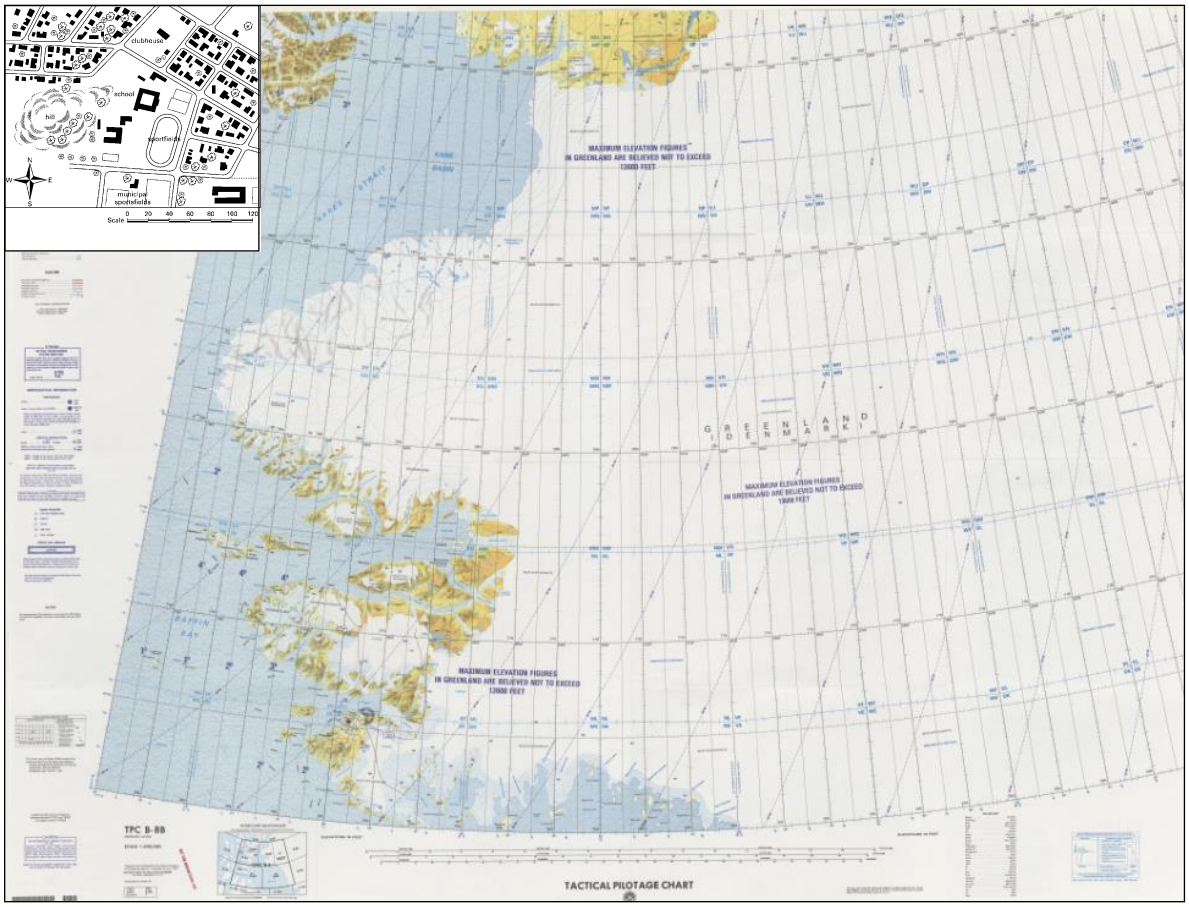

Figure 4.2: Comparison of the largest sketch image (upper left, 3,599x3,474 pixels) with the largest pilotage chart image (17,601x12,741 pixels). Although the large ice fields in the Greenland pilotage chart generate a lot of weak descriptors by using a grid approach to find interest points, this image provides many additional possibilities for strong, more descriminative interest points. The resulting descriptors will remain with much more image information than the selected strongest feature descriptors from the sketch image. This generates an advantage for training and evaluation which could be observed during this work.

4.3 BOW with EasyCV

This section shows the result for BOW using ORB as a detector for experiments between five and 1,000 visual words with exceptional recall and precision for five and 10 visual words.

4.3.1 Results

Recall and precision are calculated for nine repetitions on shifted training set and evaluation set with numbers of visual words between five and 1,000. The data assignment for the training set and the evaluation set is made by hand to exclude a similar training set and evaluation set setup between repetitions by chance. Table 4.6 shows the mean recall and precision, Figure 4.4 shows the corresponding *receiver operating characteristics* (ROC) curve. With ORB as interest point detector and descriptor, the setup with 500 visual words shows the highest recall with 99.14%, but also the lowest precision (92.68%). The highest precision can be observed for 50 visual words (93.82%); the lowest recall shows up for 10 visual words (97.93%). An exhaustive examination of possible detector/descriptor combinations on the Hamming cluster cannot be solved in time as this would take several weeks. Table 4.7 shows recall and precision of the runs on the cluster. The combination of the *Binary Robust Invariant Scalable Keypoints*-detector (BRISK) and the SURF descriptor results in the best recall (98.89%), together with the highest precision (89.90%).

Number Visual Words	5	10	25	50	100	250	500	1000
Recall	0.9796	0.9793	0.9858	0.9836	0.9861	0.9889	0.9914	0.9907
Precision	0.9360	0.9335	0.9299	0.9382	0.9335	0.9282	0.9268	0.9292

Table 4.6: Results for BOW with EasyCV on two classes: positives and negatives. The average of nine repeated computational runs for each dictionary size is shown in this table. The good performance with just five or 10 visual words is exceptional. The interest point detector and descriptor used is ORB.

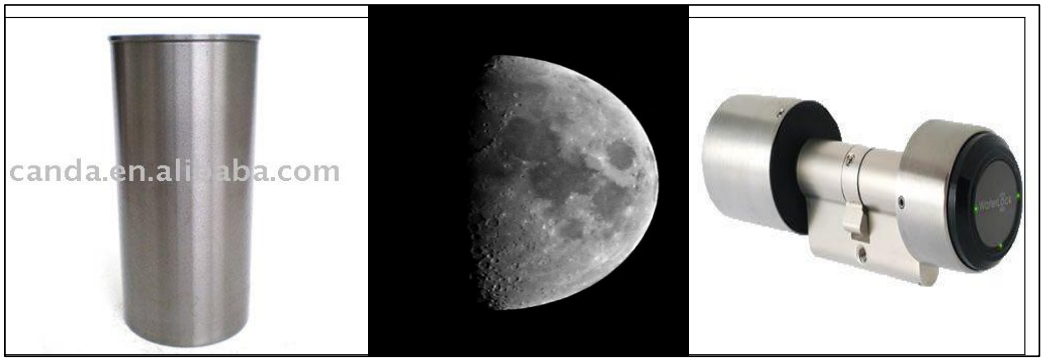

Figure 4.3: Examples of misclassified negatives after a run with 1,000 visual words and interest point detection on grid interceptions with step size [64 64]. Many descriptors are calculated on homogeneously colored areas (in these cases, black or white) and these images are finally classified as sketches.

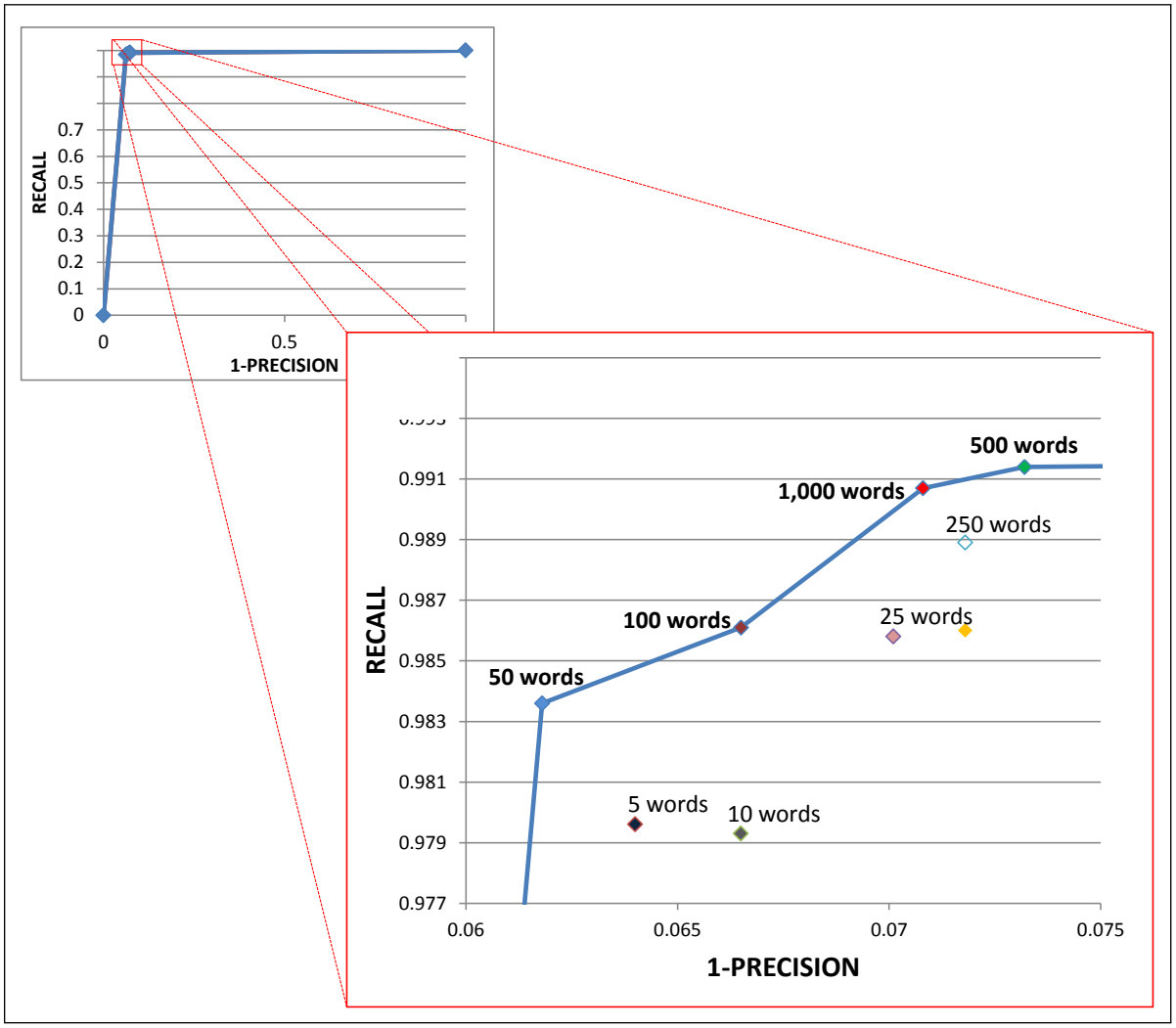

Figure 4.4: ROC curve for the best performing visual word constellations. The highest precision can be reached by using 50 visual words, the highest recall with 1,000 visual words.

Detector	ORB	ORB	BRISK	BRISK	SIFT	ORB
Descriptor	SIFT	SURF	SIFT	SURF	SURF	ORB
Recall	0.9222	0.9889	0.9889	0.9889	0.9778	0.9914
Precision	0.8557	0.8812	0.8396	0.8990	0.8889	0.9268

Table 4.7: Runs with detector/descriptor combinations using EasyCV. The combination of ORB as an detector and descriptor outperforms the other hyper-parameter settings.

4.3.2 Discussion

The result for five and 10 visual words, using BOW with ORB interest point detector and descriptor, is surprising as a recall of almost 0.98% and a precision of over 93.00% is an unexpectedly good result for this low dictionary size. In comparison, MATLAB with five visual words and step size [16 16] generates a recall of 42.50% and a 80.10% precision. For 10 visual words, these numbers rise to a recall of 93.06% and a precision of 94.10%. This, together with the variability in the results of the detector/descriptor combination-runs on 250 visual words, show the need for a deeper examination of BOW's hyper-parameter space, if one eventually wants to find the best setting for map images.

4.4 Convolutional Neural Network

This section examines the CNN results after training on two classes, positives and negatives, and after training with four map sub-classes and negatives. The last approach, again, addresses the following research question: Are there any special types of sub-classes that significantly weaken the classifiers in their performance? The results shows an obvious dependency between quantity of training input and resulting performance. Data augmentation, the generation of additional images out of the existing database, could be a useful option to increase the overall performance.

4.4.1 Results

The results of four runs on the Hamming cluster computer are used to evaluate the CNN approach for the map classification task. Given the two classes, positives and negatives, the runs differ by the factor of augmentation of the input dataset. By activating the mirror function, every patch is additionally mirrored, and this mirrored patch is also included into training. Both ways result in a recall of about 99% (99.17% with no mirror, 98.89% with

mirror); the more noticeable effect of activating the mirror option is an increase in the precision from 90.60% to 93.44%.

The results of training and testing the CNN on four map sub-classes and the negatives are shown in Table 4.8. No obvious improvement can be observed by using the higher augmentation factor. The number of correctly classified negatives is slightly better with mirror; among the map sub-classes, basic maps and web maps perform best. As the training data is imbalanced, which means that the number of input images for training differs between the classes/sub-classes, the trained CNN will be biased in a way that prefers to classify in favor of overrepresented classes/sub-classes. This problem is examined, for example, by Masko & Hensman (2015). The CNN is additionally trained and tested on a balanced set of classes/sub-classes. As the last experiment of this work, this balanced approach is running under time constraints. Therefore, no oversampling method is used, the number of training images is 105 and the number of evaluation images is 45 for each class/sub-class. The results can be seen in Table 4.9. For sub-class sketches up to 100.00% (no mirror) of the contained images are classified correctly. Additionally 73.33% of the sub-class web maps are misclassified as sketches using no mirror.

4.4.2 Discussion

For the first approach, using two classes, the impact of augmentation can be observed for the negatives class. Having more images, because of mirroring patches, allows an improvement of 2.84 percentage points for correct classifications. The results of the second approach are for both ways, balanced and imbalanced, examples for limitations of CNNs. A balanced dataset with only a few images for training will result in unreliable classifiers. In this case a CNN classifies 100% of all sketches correctly (without mirror), but at the same time it also classifies 73.33% of all web maps as sketches. The imbalanced dataset causes a bias that prefers to classify data more often as members of the classes/sub-classes with the highest number of training images. This is observable in Table 4.8, where the classification process prefers a basic maps membership, as basic maps is the sub-class with the largest number of training images. But a solution is at hand: balancing the dataset by either oversampling the underrepresented sub-classes (e.g., rotating, mirroring or adjusting contrast of the training images), or raising the number of sub-class images by intensifying the online-search for images.

	Mirror:NO	Basic Maps	Pilotage Charts	Predicted Web Maps	Sketches	Negatives
	Basic Maps	**98.67**%	0.67%	0.67%	0.00%	0.00%
	Pilotage Charts	28.89%	**68.89**%	2.22%	0.00%	0.00%
Actual	Web Maps	5.00%	0.00%	**94.17**%	0.83%	0.00%
	Sketches	22.22%	0.00%	35.56%	**42.22**%	0.00%
	Negatives	0.56%	3.89%	2.22%	0.83%	**92.50**%
	Mirror:YES	Basic Maps	Pilotage Charts	Predicted Web Maps	Sketches	Negatives
	Basic Maps	**98.00**%	0.67%	1.33%	0.00%	0.00%
	Pilotage Charts	28.89%	**71.11**%	0.00%	0.00%	0.00%
Actual	Web Maps	5.00%	0.00%	**94.17**%	0.00%	0.83%
	Sketches	11.11%	0.00%	35.56%	**48.89**%	4.44%
	Negatives	0.56%	1.94%	1.11%	1.67%	**94.72**%

Table 4.8: Confusion matrices of four sub-classes and negatives for runs with deactivated/activated mirror on a imbalanced training set. Pilotage charts and sketches are the weakest links in the chain.

	Mirror:NO	Basic Maps	Pilotage Charts	Predicted Web Maps	Sketches	Negatives
	Basic Maps	**71.11**%	6.67%	0.67%	22.22%	0.00%
	Pilotage Charts	33.33%	**42.22**%	8.89%	8.89%	6.67%
Actual	Web Maps	0.00%	0.00%	**11.11**%	73.33%	15.56%
	Sketches	0.00%	0.00%	0.00%	**100.00**%	0.00%
	Negatives	0.00%	0.00%	6.67%	15.56%	**77.78**%
	Mirror:NO	Basic Maps	Pilotage Charts	Predicted Web Maps	Sketches	Negatives
	Basic Maps	**91.1%1**	4.44%	0.00%	4.44%	0.00%
	Pilotage Charts	26.67%	**57.78**%	6.67%	8.89%	0.00%
Actual	Web Maps	0.00%	0.00%	**33.33**%	66.67%	0.00%
	Sketches	8.89%	0.00%	2.22%	**88.89**%	0.00%
	Negatives	0.00%	0.00%	6.67%	17.78%	**75.56**%

Table 4.9: Confusion matrices of four sub-classes and negatives for runs with deactivated/activated mirror on a balanced training set. Pilotage charts and sketches are no longer the weakest links in the chain.

4.5 Insights for the Research Questions

As the questions for the geographical space of interest and for the relevant sub-classes for this work are answered in chapter 3, this section merges the discussion points of the former paragraphs to answer the remaining research questions. It addresses the impact of sub-classes on the performance of the classifier, other factors that impact the classification results, and thoughts about the potential of BOW and CNN for a map classification application.

One task of this thesis is to determine whether there are any special types of sub-classes that significantly weaken the classifier in its performance. With web maps and sketches for BOW such sub-classes show up. But as for these sub-classes, the misclassifications address only other map sub-classes; this impact is almost eliminated with a switch to a two-class classification task on the negatives class and the positives class. The results of the BOW approach can be explained for web maps with a discrimination weakness between large scale images of this sub-class and sketches. Sketches by themselves provide very little information to calculate reliable descriptors. In comparison with the sub-class pilotage charts, which shares with sketches the same low number of training and evaluation images, the BOW method is able to perform outstandingly as the training images of this sub-class allow a more distinct description by visual words. This remarkable insight recommends BOW as a solution for classification tasks on small datasets with useful content for calculating descriptor properties.

Part of this research is to answer the question, which of the chosen methods performs best with respect to recall and precision? For both BOW and CNN, a maximal recall of 99.17% can be observed, which means that three positive images are misclassified as negatives. BOW performs better with respect to precision: Its best result, 97.01%, is 3.57 percentage points better than the best result with a CNN. When the size of the training data is increased, the precision of the CNN improves from 90.60% to 93.44%. This research does not address whether the precision can be increased through additional inputs until BOW is finally outperformed. The best results of the different approaches are summarized in Table 4.10.

This work shows that several factors influence the classification results. As the research

is not focused on an exhaustive examination of this influence, it remains for future work to evaluate the importance of these factors in relationship to one another. For the CNN, the results recommend a further investigation with significantly bigger training data and balance between the training images of the classes/sub-classes to learn the correct response of its neural network to new data. As BOW is limited in the description of the solution space by the defined dictionary size, it relies on the usefulness of the training data for the descriptor calculation much more than on the number of training images. This method is fragile to similarities between and high variability within sub-classes. This problem is quite reasonable as even human beings can have similar difficulties in discriminating dog breeds (inter-class similarity) or recognizing, for example, strawberries as a genus of the rose family (intra-class variability). In these cases, the limited amount of visual words of BOW is comparable to a limitation of the memory capacity, which allows the remembering of rough distinctive characteristics more than fine details. This work also shows that just an increase of the dictionary size does not guarantee an improvement of the method. Overfitting can become an influential factor; for the chosen dataset, this can be observed for the negatives class, where the number of true negatives decreases by increasing the dictionary size above 100.

Finally, it is necessary to discuss whether one or both of the chosen methods become handy applications in an analyst's environment. Considering the results, having a tool with a recall of 99% and a precision of roughly 95% can certainly be helpful. Assume a situation with an input dataset of one million images; among them, unknown to the analyst, are 1,000 positives. The tool will work on this non-stop, until it finally comes up with 990 positives and an additional 49,950 misclassified negatives (false positives). At first glance, 50,940 classified positives (true and false) seems to be a lot, but it means that 949,060 images (among them 10 positives) do not have to be scanned by an analyst. Exhausting this example a little bit, assume an analyst can visually scan and classify one image per second (which was the computational speed for BOW with step size [16 16] and 100 visual words), and he or she is able to do this for six hours a day (which seems difficult). This means that he or she can classify 21,600 images per day. So classifying 50,940 images would be done in two days, two hours and nine minutes. Classifying one million images would be done in 46 days, one hour and 46 minutes. Therefore, the time for analyzing the data is reduced to 5.1% of the time needed to analyze the original dataset by using one of the

examined computational approaches and accepting the loss of 10 positive images. As both the computational method and the analyst can work simultaneously, computational speed is less relevant. Even assuming BOW or CNN needs three seconds to classify an image (typical speed for BOW might be 1 second, CNNs can classify 30+ images per second), 24/7 computations allow a total of 28,800 classifications a day, which exceeds the analyst's daily capacity.

So in this case, both BOW and CNN can be helpful classification tools for the specific task of map classification. Switch the prior example to 999,000 positives and 1,000 negatives. Things change, as now most of the images classified as negatives are actually positives (~10,000). The usage of a computational method in this situation is obviously not helpful. Whether BOW and CNN fit to data forensic tasks depends on specific needs for such an application in the working environment and what kind of data should be processed. A CNN provides a huge learning capacity and therefore the possibility to cover an extremely wide class and sub-class spectrum. BOW is a useful alternative whenever training data is rare but useful with respect to the descriptor generation. To eventually come up with a useful application, there must be additional research carried out as well as close cooperation between a potential user and the developer. Both BOW and CNN have the potential to become the computational backbone of such a product. Chapter 5 addresses this point with ideas for future work.

		Recall	Precision	Accuracy
BOW–MATLAB	two classes	0.9917	0.9701	0.9791
BOW–EasyCV	two classes	0.9914	0.9382	0.9594
CNN	two classes	0.9917	0.9346	0.9597
BOW–MATLAB	with sub-classes			0.9181
CNN	with sub-classes			0.9097

Table 4.10: Overview of the best results for the different classification approaches. Recall and precision are not necessarily results of the same parameter setting. Accuracy, which provides as portion of correct classifications among all images a measurement of how close the result is to the actual data (sum of all correct classifications divided by the overall number of images in the evaluation set), is calculated for the runs with sub-classes and, for a better comparison, also for the two-classes classification runs.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 5:
Conclusions and Future Work

5.1 Future Work

In this research, neither CNN or BOW are examined in its full capacity. For CNN, the dataset has to be increased, especially for the sub-classes of sketches and pilotage charts to guarantee balance. Knowing that sketches are rare, adding new images will be a time-consuming challenge and could possibly be avoided by generating additional training images (e.g.,by rising the augmentation factor). The dataset could be spread out to new map sub-classes to evaluate the learning capacity of the CNN.

Expanding the number of sub-classes in the dataset is also an interesting research task for BOW. Given the options of training one classifier for all sub-classes and classes or generating individual models for each of them, the performance of BOW should be examined on a dataset with a larger number of sub-classes and classes. Will BOW thereby become too expensive to be implemented in an application? This task is asking for the impact of a dataset spread on the necessary dictionary size to guarantee high recall and precision, and the computational speed for evaluating data, using classifiers that are parallel or successive.

For the observed BOW problem with sub-classes, which contain only a few, low detection-quality images, it would be interesting to see whether the classification can be improved by combining a interest point detection algorithm with the grid step method. In combination with augmentation (which has thus far only been an option for CNN), this variation could possibly increase the number of interest points. The interest point detection algorithm should guarantee that the meaningful interest points will not be missed. Using MATLAB, this approach should additionally increase the number of usable interest points for all classes (because the number of usable interest points is bounded by the number of available interest points for the weakest class). This could therefore affect the classification outcome for each class.

For BOW and the sub-class web maps, the intra-class variability has an effect on the classification outcome, as shown in subsection 4.2.2. Can recall and precision be improved

by tightening sub-classes? Most detectors are scale invariant, but changing the scale of a map usually means changing the provided information (contained objects). For example, zooming into a Google map reveals additional information like street names or symbols in a first step. For even larger scales, it may be that the geographic zone of interest is just a monotone-colored area with a single road on it. Just defining one sub-class for Google maps would mean training a classifier for the variability of possible objects, not only in relation to the geographic region with its individual properties, but also for scale-dependent content. Perhaps it is better to divide the web maps sub-class into several additional sub-classes to cover different scales.

Future research should also spread the map classification space by adapting differing real world data. A photo of a map on a table is a simple example that illuminates the discrepancies between the image data used in this work and something that might be more likely to show up on an analyst's screen. So the performance of BOW and CNN should be evaluated on what may be more realistic input: images containing maps as objects among other objects.

Aside from examination of the evaluation speed of CNN and BOW on standard computer systems, a last recommendation for future work is the construction of a challenge-dataset that mirrors, as far as possible, a data forensic domain. This dataset should become the standard corpus for measuring the accuracy of a method, either on specific classes or on the whole dataset, just as the Large Scale Visual Recognition Challenge dataset is such a standard corpus for researchers. This would be helpful for developing methods and applications oriented towards the user's working environment.

5.2 Conclusions

Given the task of finding geographic map images in a dataset, this thesis evaluates two computer-based classification and object detection methods–Bag of Words and a Convolutional Neural Network. The outcome of these approaches is examined on a self-constructed dataset with 1,200 positive and 1,200 negative images. Although both methods produce recalls up to 99.17% on tasks with two classes, BOW's best precision, at 97.01%, tops CNN by over 3.5 percentage points. But the results with different amounts of training images indicate that the precision with CNNs can be subject to further improvement by additional

data augmentation.

With up to 100% correct classifications, BOW can generate remarkable results on sub-classes with low numbers of training images. Rather than image quantity, the performance of BOW depends much more on the descriptor-specific image information and the intra-class variability. For example, this variability may occur when the geographic scale of the map data changes within a sub-class. Facing the challenge of having a sub-class with few images, determining the best method to create a classifier depends on the answer to the following question: Is this data good enough for BOW or, if not, could it be numerically increased (e.g., by augmentation) to get a good classifier by training a CNN? For the specific task of classifying the map image space as spanned by the sub-classes described in this work, both approaches can become the backbone of helpful applications, especially if the search for maps is comparable to the search for a needle in a haystack. Although the evaluated methods show remarkably good results on the chosen dataset, there are still new, unanswered questions.

THIS PAGE INTENTIONALLY LEFT BLANK

List of References

About.com (2015). Types of maps. Retrieved July 23, 2015, from http://geography.about.com/od/understandmaps/a/map-types.htm

Arel, I., Rose, D. C., & Karnowski, T. P. (2010). Deep machine learning - A new frontier in artificial intelligence research. *IEEE Computational Intelligence Magazine*, 5(4), 13–18.

Baltsavias, E. (2004). Object extraction and revision by image analysis using existing geodata and knowledge: Current status and steps towards operational systems. *ISPRS Journal of Photogrammetry and Remote Sensing*, 58(3), 129–151.

Bay, H., Tuytelaars, T., & Van Gool, L. (2006). Surf: Speeded up robust features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, (pp. 404–417). Luxembourg, Luxembourg: Springer.

Bengio, Y., LeCun, Y., & Henderson, D. (1994). Globally trained handwritten word recognizer using spatial representation, convolutional neural networks, and hidden Markov models. In *Proceedings of the Neural Information Processing Systems Conference*, (pp. 937–937). La Jolla, CA: Neural Information Processing Systems Foundation.

Bilen, H., Pedersoli, M., Namboodiri, V. P., Tuytelaars, T., & Van Gool, L. (2014). Object classification with adaptable regions. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, (pp. 3662–3669). Piscataway, NJ: IEEE.

Bradski, G. (2000). OpenCV [Programming libary]. *Dr. Dobb's Journal of Software Tools*.

Calonder, M., Lepetit, V., Strecha, C., & Fua, P. (2010). Brief: Binary robust independent elementary features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, (pp. 778–792). Luxembourg, Luxembourg: Springer.

Chang, S.-K., & Hsu, A. (1992). Image information systems: Where do we go from here? *IEEE Transactions on Knowledge and Data Engineering*, 4(5), 431–442.

Csurka, G., Dance, C., Fan, L., Willamowski, J., & Bray, C. (2004). Visual categorization with bags of keypoints. In *Proceedings of the ECCV Workshop on Statistical Learning in Computer Vision*, Vol. 1, (pp. 1–2). Aachen, Germany: UMIC Research Centre.

Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR*, Vol. 1, (pp. 886–893). Piscataway, NJ: IEEE.

Deb, S., & Zhang, Y. (2004). An overview of content-based image retrieval techniques. In *Proceedings of the 18th International Conference on Advanced Information Networking and Applications*, Vol. 1, (pp. 59–64). Piscataway, NJ: IEEE.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. Retrieved from http://www.image-net.org/papers/imagenet_cvpr09.pdf

Divvala, S. K., Farhadi, A., & Guestrin, C. (2014). Learning everything about anything: Webly-supervised visual concept learning. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, (pp. 3270–3277). Piscataway, NJ: IEEE.

Dorai, C., & Venkatesh, S. (2003). Bridging the semantic gap with computational media aesthetics. *IEEE multimedia*, 10(2), 15–17.

Fei-Fei, L. (2010). Imagenet: crowdsourcing, benchmarking & other cool things. Retrieved from http://www.image-net.org/papers/ImageNet_2010.pdf

Felzenszwalb, P. F., Girshick, R. B., McAllester, D., & Ramanan, D. (2010). Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9), 1627–1645.

Ganpatrao, N. G., & Ghosh, J. K. (2014). Information extraction from topographic map using colour and shape analysis. *Sadhana*, 39(5), 1095–1117.

Garcia, C., & Delakis, M. (2004). Convolutional face finder: A neural architecture for fast and robust face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(11), 1408–1423.

Geoscience Australia (2015). What is a topographic map? Retrieved July 23, 2015, from http://www.ga.gov.au/scientific-topics/geographic-information/topographic-maps-data/basics/what-is-a-topographic-map

Google (2015a). Google image search. Retrieved August 6, 2015, from https://google.com/

Google (2015b). Google maps. Retrieved August 6, 2015, from https://maps.google.com/

Guillaumin, M., Verbeek, J., & Schmid, C. (2010). Multimodal semi-supervised learning for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (pp. 902–909). Piscataway, NJ: IEEE.

Haar, A. (1910). Zur theorie der orthogonalen funktionensysteme. *Mathematische Annalen*, 69(3), 331–371.

Harris, Z. S. (1954). Distributional structure. In *Word*. New York City, NY, Linguistic Circle of New York.

Heisele, B., Ho, P., Wu, J., & Poggio, T. (2003). Face recognition: Component-based versus global approaches. *Computer Vision and Image Understanding*, 91(1), 6–21.

Horridge, G. (1987). The evolution of visual processing and the construction of seeing systems. *Proceedings of the Royal Society of London B: Biological Sciences*, 230(1260), 279–292.

Infographic, D. (2015). What happens in an Internet minute [Infographic]. Retrieved July 26, 2015, from http://dig.dailyinfographic.com/what-happens-in-an-internet-minute-infographic

Jain, A. K., & Vailaya, A. (1996). Image retrieval using color and shape. *Pattern recognition*, 29(8), 1233–1244.

Jang, K. S., Yi, J., Jung, J. Y., Kim, J., & Chang, K. H. (1997). A recognition of map using the geometric relations between lines and the structural information of objects. In *Proceedings of the International Conference on Image Processing*, Vol. 3, (pp. 150–153). Piscataway, NJ: IEEE.

Janssen, R. D., Duin, R. P., & Vossepoel, A. M. (1993). Evaluation method for an automatic map interpretation system for cadastral maps. In *Proceedings of the Second International Conference on Document Analysis and Recognition*, (pp. 125–128). Piscataway, NJ: IEEE.

Jégou, H., Douze, M., & Schmid, C. (2010). Improving bag-of-features for large scale image search. *International Journal of Computer Vision*, 87(3), 316–336.

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., & Darrell, T. (2014). Caffe: Convolutional architecture for fast feature embedding. Retrieved from http://ucb-icsi-vision-group.github.io/caffe-paper/caffe.pdf

Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. In *Proceedings of the 10th European Conference on Machine Learning (ECML-98)*, (pp. 137–142). Luxembourg, Luxembourg: Springer.

Karpathy, A., & Fei-Fei, L. (2014). Deep visual-semantic alignments for generating image descriptions. Retrieved from http://cs.stanford.edu/people/karpathy/cvpr2015.pdf

Kato, T. (1992). Database architecture for content-based image retrieval. In *Proceedings of the Symposium on Electronic Imaging: Science and Technology (SPIE/IS&T)*, (pp. 112–123). Bellingham, WA: International Society for Optics and Photonics.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. F. Pereira, C.J.C. Burges, L. Bottou, & K.Q. Weinberger, (Eds.), *Advances in neural information processing systems*, (pp. 1097–1105). Rostrevor, N. Ireland: Curran Associates.

Lawrence, S., Giles, C. L., Tsoi, A. C., & Back, A. D. (1997). Face recognition: A convolutional neural-network approach. *IEEE Transactions on Neural Networks*, 8(1), 98–113.

LeCun, Y., & Bengio, Y. (1994). Word-level training of a handwritten word recognizer based on convolutional neural networks. In *Proceedings of the International Conference on Pattern Recognition*, (pp. 88–92). Piscataway, NJ: IEEE.

LeCun, Y., & Bengio, Y. (1995). Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10).

Leutenegger, S., Chli, M., & Siegwart, R. Y. (2011). Brisk: Binary robust invariant scalable keypoints. In *Proceedings of the International Conference on Computer Vision (ICCV)*, (pp. 2548–2555). Piscataway, NJ: IEEE.

Li, L.-J., Su, H., Fei-Fei, L., & Xing, E. P. (2010). Object bank: A high-level image representation for scene classification & semantic feature sparsification. In *Advances in neural information processing systems*, (pp. 1378–1386). Rostrevor, N. Ireland: Curran Associates.

Liu, Y., Zhang, D., Lu, G., & Ma, W.-Y. (2007). A survey of content-based image retrieval with high-level semantics. *Pattern Recognition*, 40(1), 262–282.

Lloyd, S. P. (1982). Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2), 129–137.

Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, Vol. 2, (pp. 1150–1157). Piscataway, NJ: IEEE.

Machajdik, J., & Hanbury, A. (2010). Affective image classification using features inspired by psychology and art theory. In *Proceedings of the International Conference on Multimedia*, (pp. 83–92). Beijing, China: ACM.

Maron, M. E. (1961). Automatic indexing: An experimental inquiry. *Journal of the ACM*, 8(3), 404–417.

Masko, D., & Hensman, P. (2015). The impact of imbalanced training data for convolutional neural networks. Retrieved September 5, 2015, from http://www.diva-portal.org/smash/get/diva2:811111/FULLTEXT01.pdf

MATLAB (2014). Mathworks matlab version 8.4.0.150421 (r2014b) [programming language]. Natick, MA: MathWorks.

Microsoft (2015). Bing maps. Retrieved August 6, 2015, from https://www.bing.com/maps/

Nowak, E., Jurie, F., & Triggs, B. (2006). Sampling strategies for bag-of-features image classification. In *Proceedings of the International Conference on Computer Vision (ECCV)*, (pp. 490–503). Luxembourg, Luxembourg: Springer.

Nowlan, S. J., & Platt, J. C. (1995). A convolutional neural network hand tracker. F. Pereira, C.J.C. Burges, L. Bottou, & K.Q. Weinberger, (Eds.), *Advances in neural information processing systems*, (pp. 901–908). Rostrevor, N. Ireland: Curran Associates, Citeseer.

NPSVisionLab (2015). Easy! Computer vision. Retrieved August 12, 2015, from http://movesinstitute.org/~kolsch/CVAC/tutorial.html

PennyStocks.la (2015). The Internet in real time. Retrieved July 15, 2015, from http://pennystocks.la/internet-in-real-time/

Python Software Foundation (2014). Python programming language. Retrieved from http://www.python.org

Rosin, P. L. (1999). Measuring corner properties. *Computer Vision and Image Understanding*, 73(2), 291–307.

Rosten, E., & Drummond, T. (2005). Fusing points and lines for high performance tracking. In *Proceedings of the Tenth International Conference on Computer Vision (ICCV)*, Vol. 2, (pp. 1508–1515). Piscataway, NJ: IEEE.

Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. (2011). Orb: An efficient alternative to sift or surf. In *Proceedings of the International Conference on Computer Vision (ICCV)*, (pp. 2564–2571). Piscataway, NJ: IEEE.

Rui, Y., Huang, T. S., & Chang, S.-F. (1999). Image retrieval: Current techniques, promising directions, and open issues. *Journal of Visual Communication and Image Representation*, 10(1), 39–62.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., ..., & Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, (pp. 1–42).

San Roque, L., Kendrick, K. H., Norcliffe, E., Brown, P., Defina, R., Dingemanse, M., . . . , & Majid, A. (2015). Vision verbs dominate in conversation across cultures, but the ranking of non-visual verbs varies. *Cognitive Linguistics*, 26(1), 31–60.

Scale (map). (n.d.). (2015). In *Wikipedia*. Retrieved August 24, 2015, from https://en.wikipedia.org/wiki/Scale_(map)

Serrano, N., Savakis, A. E., & Luo, J. (2004). Improved scene classification using efficient low-level features and semantic cues. *Pattern Recognition*, 37(9), 1773–1784.

Sweeney, L. (2001). Information explosion. Doyle, P., Lane, J., Theeuwes, J., & Zayatz, L., (Eds.), *Confidentiality, disclosure, and data access: Theory and practical applications for statistical agencies*, (pp. 43–74). New York, NY: Elsevier Science.

Szeliski, R. (2010). *Computer vision: Algorithms and applications*. London, UK: Springer.

Torralba, A., Russell, B. C., & Yuen, J. (2010). Labelme: Online image annotation and applications. *Proceedings of the IEEE*, 98(8), 1467–1484.

University of Texas (2015). Perry-Castañeda Libary map collection. Retrieved August 07, 2015, from http://www.lib.utexas.edu/maps/

Vailaya, A., Figueiredo, M. A., Jain, A. K., & Zhang, H.-J. (2001). Image classification for content-based indexing. *IEEE Transactions on Image Processing*, 10(1), 117–130.

Vailaya, A., Jain, A., & Zhang, H. J. (1998). On image classification: City images vs. landscapes. *Pattern Recognition*, 31(12), 1921–1935.

Vapnik, V., & Chervonenkis, A. (1964). A note on one class of perceptrons. *Automation and Remote Control*, 25(1).

Vlasenko, P. (2015). Russian and Ukraine maps. Retrieved August 07, 2015, from http://maps.vlasenko.net/

Vondrick, C., Patterson, D., & Ramanan, D. (2013). Efficiently scaling up crowdsourced video annotation. *International Journal of Computer Vision*, 101(1), 184–204.

Wolkenhauer, W. (1895). *Leitfaden zur Geschichte der Kartographie in tabellarischer Darstellung*. [Guideline of the history of maps in tabular representation]. Wroclaw, Poland: F.Hirt.

Yammen, S., & Muneesawang, P. (2014). An advanced vision system for the automatic inspection of corrosions on pole tips in hard disk drives. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 4(9), 1523–1533.

Yang, J., Jiang, Y.-G., Hauptmann, A. G., & Ngo, C.-W. (2007). Evaluating bag-of-visual-words representations in scene classification. In *Proceedings of the International Workshop on Multimedia Information Retrieval*, (pp. 197–206). Beijing, China: ACM.

Zhang, P., Wang, J., Farhadi, A., Hebert, M., & Parikh, D. (2014). Predicting failures of vision systems. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, (pp. 3566–3573). Piscataway, NJ: IEEE.

Zhang, T., Ramakrishnan, R., & Livny, M. (1997). Birch: A new data clustering algorithm and its applications. *Data Mining and Knowledge Discovery*, 1(2), 141–182.

THIS PAGE INTENTIONALLY LEFT BLANK

Initial Distribution List

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California