Text-aided Object Segmentation and Classification in Images

Agnes Tegen

Master's thesis
2014:E5

Lund University

Faculty of Engineering
Centre for Mathematical Sciences
Mathematics

Centrum Scientiarum Mathematicarum

Text-aided object segmentation and classification in images

Agnes Tegen

31 January 2014

Acknowledgements

I would like to thank everybody who have helped me in the work with my master thesis. I would especially like to thank my supervisor Kalle Åström for answering my stream of questions and making me go from confused to enthusiastic many times. I would also like to thank my co-supervisors Pierre Nugues and Magnus Oskarsson for all their ideas concerning the project and their help in technical matters. Additionally, I would like to thank my fellow students Rebecka Weegar and Linus Hammarlund, who wrote a project parallel and much in cooperation with my thesis. Finally, I would like to thank Peter Carbonetto for kindly lending me his code to use and edit.

Abstract

Object recognition in images is a popular research field with many applications including medicine, robotics and face recognition. The task of automatically finding and identifying objects in an image is challenging in the extreme. By looking at the problem from a new angle and including additional information beside the visual, the problem becomes less ill posed.

In this thesis we investigate how the addition of text annotations to images affects the classification process. Classifications of different sets of labels as well as clusters of labels were carried out. A comparison between the results from using only visual information and from also including information from an image description is given. In most cases the additional information improved the accuracy of the classification.

The obtained results were then used to design an algorithm that could, given an image with a description, find relevant words from the text and mark their presence in the image. A large set of overlapping segments is generated and each segment is classified into a set of categories. The image descriptions are parsed by an algorithm (a so called chunker) and visually relevant words (key-nouns) are extracted from the text. These key-nouns are then connected to the categories by metrics from WordNet. To create an optimal assignment of the visual segments to the key-nouns combinatorial optimization was used. The resulting system was compared to manually segmented and classified images.

The results are promising and have given rise to several new ideas for continued research.

.

Contents

.

1 Introduction

Recognizing objects in images and in real life is something many of us do everyday. Most of the time we do it at an instant and without thinking about it. Since humans perform this task with such ease one might easily overlook the complexity of it. Yet today it is not fully known how the brain recognizes objects, even though there are a number of theories that suggest models for this process [1]. Considering this, it is easy to see why teaching a computer, or more specifically designing an algorithm, to perform this task to the full extent is still a challenge.

Roughly, object recognition can be said to consist of two parts: segmentation and classification. Segmentation is the process of dividing an image into sections, e.g. finding where one object ends and another begins or separating foreground from background. Classification is the identification of what these sections represent.

Even though object recognition is a complex problem, huge strides have been made in the last decades. It is a popular research field with many applications. Because of the complexity of the task, it can be approached with different methodologies. Felzenszwalb et al [2] used a method where they divided the image into certain regions and utilized dynamic programming to optimize the segmentation and labeling. The method was used both on scenery and images depicting specific objects. A restriction was that the division into regions was done following a certain scheme of five labels (top/sky, bottom/ground, facing left, facing right and front facing) and the objects had to have a shape prior. Thus this method worked very well on a particular type of images but was less satisfying on others.

Another approach was made by Taylor et al [3]. They used video instead of single images and could thus utilize the information from all the sequential images. By comparing consecutive images, objects moving in front of a background could be detected. Areas occluded by an object in one image were visible in others, which helped with the otherwise often encountered problem that only a part of an object or area is perceptible.

Chum et al [4] used a method that is usually utilized in text retrieval contexts called query expansion. They focused on a particular object and wanted to retrieve all occurrences of this object from a large database of images. Starting with one query image, the visual information about the object increased as more instances were found which aided the detection of the specified object in even more images.

These examples illustrate the importance of limiting the research to a subproblem. Beside this, one must usually make assumptions about the images to be analyzed (e.g. Felzenszwalb et al [2]) and/or take in extra information from somewhere else (e.g. Taylor et al [3]). One method to acquire more information than just the visual information from an image is to explore images with associated text or annotations. This was the starting point for this master thesis. This approach is relatively new, but there has already been progress. Moscato et al [5] used images from the image hosting website *Flickr* as their database. This system contains, besides images, also tags, keywords, annotations etc. To classify the images they were particularly interested in visually similar images, but where the annotations made by humans differed.

Medved et al [6] also combined images with accompanying text to improve

classification. The focus was on human beings and horses. Images and associated articles were collected from Wikipedia. The goal was to classify the relation between the person(s) and horse(s) in the image (e.g. ride or lead).

This thesis explores how the addition of image descriptions alters object recognition, compared to just using information found in the image. The problem of object recognition becomes more well defined when an image description is added, not only for an algorithm, but also for humans manually segmenting the image. With a busy picture some might argue that each and every one of the objects should be segmented, while others might think differently. For example, if the picture depicts a city and is taken from an air plane, each individual house might be barely noticeable, but to everyone looking at the picture it is clear that it is a city, which contains a lot of buildings. With the added description "View of city from an airplane" it is clear that "city" is the important word and the segment of interest would be all the buildings combined. With another description, like "Aerial view of Paris, where Tour Montparnasse and the Eiffel Tower can be seen", one might rather want to specify the regions where the two buildings are located in the image respectively.

In this master thesis both text and image properties are used to find certain objects of interest in the image. The focus was slightly different from the examples mentioned earlier. The number of different types of objects/scenery to be classified started out quite small, at five, but was successively increased up to 100. The larger amount of labels was also divided into a set of clusters and these clusters were used for classification as well. The aim was to examine how the classification changed when adding information found in an accompanying description.

In addition to exploring the changes in classification, the results found were also utilized. With a model trained for classification, an algorithm was constructed. This started with the image description and from this it sought the relevant words in the image.

The task of identifying and classifying objects is important in a variety of fields including medicine, robotics, face recognition and video surveillance. With the constant advances in research, the possible fields of application expand as well. With the added information from descriptions and keywords, classification can be a useful tool when searching large databases of multimedia like the ever increasing internet.

This thesis is structured in the following way. The second chapter in this thesis goes through the actual process of the work that was done. It explains important background concepts and theory a bit more thoroughly. It also introduces the data and tools utilized. Both these parts can be seen as preparation for the final part of the chapter, which explains the methods in action. Chapter 3 presents the results obtained using the methods and data described in chapter 2. In chapter 4 a discussion about the results obtained is presented. The used methods, improvements of these and what could be done in the future are discussed as well. Finally, conclusions are drawn in chapter 5.

2 Process

The work of this master thesis can be divided into two parts. In the first section the question of how the extra information from descriptions affects object recognition compared to just information found in the image, is investigated. The aim was to see if the added information improved or impaired the results or maybe did not alter them at all. The focus was especially on how the changes in result differed among different types of objects. Different types of classifiers were also examined and compared.

The goal of the second part was to find objects corresponding to specific nouns in a picture. Given an image with a corresponding description, the task was to find which words in the description were relevant and locate the corresponding objects in the image.

This section presents the process of attempting to reach these goals. Firstly, the theory behind some important concepts used is explained. Secondly, the tools and the data used will be presented, along with explanations of their part in the method. Lastly, section 2.3 describes how the theory was implemented along with how the tools and data were used.

2.1 Theory

Techniques and tools used in this thesis are based on mathematical models and models found in natural language processing. The most important ones are introduced and explained in more detail in the following section.

2.1.1 Multinomial Logistic Regression

Logistic regression is a probabilistic statistical model used for classification. Statistical classification is the task of labeling new observations: given a group of different possible labels and a set of instances, each with a known label, the goal is to correctly classify new unknown examples. Usually the term logistic regression is employed in the case of binary classification, while multinomial logistic regression is a generalization of logistic regression which allows more than two possible outcomes. When using the model an assumption is made that the features of an instance and a set of parameters linearly combined can be used to model the probabilities that this particular instance belongs to each of the possible labels.

The sought after value is the probability of instance i belonging to category k, denoted $p_k = Pr(Y_i = k)$. Beginning with the binary case as an example, there are the two options, either $Y_i = 0$ or $Y_i = 1$, both cases have a certain probability.

To predict the probability, a linear predictor function, here denoted $f(k, i)$, is used. It is defined as

$$f(k,i) = \beta_{0,k} + \beta_{1,k}x_{1,i} + \beta_{2,k}x_{2,i} + \ldots + \beta_{M,k}x_{M,i} = \boldsymbol{\beta_k} \cdot \mathbf{x_i},$$

where $\beta_{m,k}$ is the regression coefficient connected to category k and feature m and $x_{m,i}$ is the observed value of feature m for instance i. $\boldsymbol{\beta_k} \cdot \mathbf{x_i}$ is a more compact way of writing this, using vector multiplication with $\boldsymbol{\beta_k} = [\beta_{0,k}\ \beta_{1,k}\ \ldots\ \beta_{M,k}]$ and $\boldsymbol{x_i} = [1\ x_{1,i}\ \ldots\ x_{M,i}]$. The features are numerical representations of information for instance i. They can be both continuous or discrete variables.

The linear probability function is also used in linear regression. The difference between these two is that while in linear regression the outcome is a continuous variable it is not so in logistic regression, where the result is a probability. Probabilities are between 0 and 1, but the linear predictor function $f(k, i)$ can be any real number however, which is why the natural logarithm is applied to the probabilities. The probabilities must sum to one, since they form a probability distribution, which is why they are multiplied with a normalizing constant C.

This gives

$$\ln(C \cdot Pr(Y_i = 0)) = \boldsymbol{\beta_0} \cdot \boldsymbol{x_i},$$

$$\ln(C \cdot Pr(Y_i = 1)) = \boldsymbol{\beta_1} \cdot \boldsymbol{x_i},$$

which, when solving for the probabilities gives

$$Pr(Y_i = 0) = \frac{e^{\boldsymbol{\beta_0} \cdot \mathbf{x_i}}}{C},$$

$$Pr(Y_i = 1) = \frac{e^{\boldsymbol{\beta_1} \cdot \mathbf{x_i}}}{C}.$$

The normalizing constant is the sum of all the un-normalized probabilities, $C = e^{\boldsymbol{\beta_0} \cdot \boldsymbol{x_i}} + e^{\boldsymbol{\beta_1} \cdot \boldsymbol{x_i}}$. With this, the sought after probabilities become

$$Pr(Y_i = 0) = \frac{e^{\boldsymbol{\beta_0} \cdot \boldsymbol{x_i}}}{e^{\boldsymbol{\beta_0} \cdot \boldsymbol{x_i}} + e^{\boldsymbol{\beta_1} \cdot \boldsymbol{x_i}}},$$

$$Pr(Y_i = 1) = \frac{e^{\boldsymbol{\beta_1} \cdot \boldsymbol{x_i}}}{e^{\boldsymbol{\beta_0} \cdot \boldsymbol{x_i}} + e^{\boldsymbol{\beta_1} \cdot \boldsymbol{x_i}}}.$$

In general the probability of instance i belonging to category k is

$$Pr(Y_i = k) = \frac{e^{\boldsymbol{\beta_k} \cdot \boldsymbol{x_i}}}{\sum_{j=1}^{K} e^{\boldsymbol{\beta_j} \cdot \boldsymbol{x_i}}},$$

when there are K categories.

The set of parameters, $\boldsymbol{\beta_k}$, are calculated from the so called training set of instances with known labels using maximum a posteriori (MAP) estimation. MAP utilizes the observed data to generate a probability density over the value that is to be estimated and uses the maximum of this density for the estimation. To read more about logistic regression, see [7].

2.1.2 Bag-of-words model

The bag-of-words model is a representation of text commonly used in natural language processing. The occurrence of each word in the text is used as a feature, which makes the entire collection of words in the text a feature vector for that particular document and can be used in classification. This model does not take grammar or word order into account, only which words are present and how often they appear in the text. To illustrate this model an example is shown below.

If we start with the three sentences "Smaug is a cat", "Coffee is a delicious beverage" and "I like coffee and I like cats", there is a total of eleven different words. Each of these words will be given a number, which will represent their position in the feature vector. One way is to arrange them in order of appearance: "Smaug": 1, "is": 2, "a": 3, "cat": 4, "Coffee": 5, "delicious": 6, "beverage": 7, "I": 8, "like": 9, "and": 10, "cats": 11. Using the bag-of-words model the second sentence will correspond to the vector $(0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 0)$ and the third to $(0, 0, 0, 0, 1, 0, 0, 2, 2, 1, 1)$. These are the vectors that can be used for classification. Note that "cat" and "cats" are not counted as the same feature in this case. It is possible to apply stemming on all the words prior to creating the feature vector. By stemming it is meant that the word is reduced to its stem or root form. This would for instance count "cats" to the label "cat" as well. Stemming was not practiced in this master thesis however, why the example feature vector has eleven features instead of ten. To learn more about the bag-of-words model see [8].

2.1.3 The assignment problem and the Hungarian method

The assignment problem is a combinatorial optimization problem. Assume there are two sets, one which represents a number of tasks to be done and the other represents a number of agents who can perform each task. The assignment problem consists of assigning agents to do the tasks so that each task gets done. However, the agents don't do all tasks at the same cost. This is what makes it an optimization problem. It can be mathematically formulated as follows.

Given two sets A, the agents, and T, the tasks, and a weight function $C : A \times T \to R$, find a bijection $f : A \to T$ such that the cost function

$$\sum_{a \in A} C(a, f(a))$$

is minimized. Thus, the problem can be formulated as

$$\sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij}$$

subject to the constraints

$$\sum_{j=1}^{n} x_{ij} = 1, i \in A, x_{ij} \in \{0, 1\},$$

$$\sum_{i=1}^{n} x_{ij} = 1, j \in T, x_{ij} \in \{0, 1\},$$

where c_{ij} is the cost for agent i to do task j and x_{ij} is the assignment of task j to agent i. This is 1 if it is assigned, 0 otherwise.

A common solution to the assignment problem is the Hungarian method. This method starts by finding the agents with the lowest cost for each task. If some task gets more than one agent assigned to it the "next cheapest" alternative is tried out for one of the agents. This step is repeated until an optimal assignment where all the constraints hold is reached. To read more in depth about the Hungarian method see [9].

In the linear assignment problem the number of agents and number of tasks are equal, but this was not the case for the problem in this thesis as will be shown in Section 2.3.2. The modifications done to this standard problem will be explained there.

2.2 Tools and data

2.2.1 The segmented and annotated IAPR TC-12 dataset

The SAIAPR TC-12 dataset is a large dataset containing approximately 20 000 images and is available for download, see [10]. Besides the actual images, the dataset also contains a lot of other useful information. Probably most important is that every image in the dataset has been manually segmented into regions and each region has been given an appropriate label, taken from a predefined vocabulary. Aside from these collection of regions, called segmentation masks, each image also has a corresponding image description, manually written as well. These descriptions are part of the IAPR TC-12 benchmark, which contains the same set of images as the SAIAPR TC-12 dataset but without the segmentations and with the added descriptions. This set is also available for download, see [11]. A chosen set of visual properties have been calculated for every region. These features include area, the ratio boundary/area, the width and height of the region, the average and standard deviation in x- and y-coordinates, convexity, average value, standard deviation and skewness in both the RGB and CIE-lab colour space. This results in a feature vector containing 27 visual features for each segment.

This collection of images was the data used for classification. The provided segmentation, annotation and feature vectors was used for the purpose of both training and testing models. A new, automatic segmentation (see Section 2.2.3) were also applied to the images and along with new calculated feature vectors used to test the classification model.

2.2.2 LIBLINEAR

LIBLINEAR, see [12], is an open source package containing different methods used for machine learning (statistical methods that are trained on instances and used for classification or regression). It is written in C++, but has interfaces in other languages as well (MATLAB among others, which was the one used in this master thesis). Linear classification is a fast classification method when dealing with a large number of features and data where each instance is sparse.

LIBLINEAR supports different types of support vector classification and logistic regression for multi-class classification. Which of the classifiers that performs the best depends on the problem and setup at hand. While all of them were tried out in this thesis, only the one found generating the best results in this case will be explained in more detail.

Different classifiers result in different versions of the problem formulation, but regardless of which classifier is used, it is the unconstrained optimization problem

$$\min_{\omega} \frac{1}{2}\omega^T\omega + C\sum_{i=1}^{l} z(\omega; x_i; y_i)$$

Figure 1: An example of an image from the dataset with the segmentation masks and the description *"A yellow parrot with light blue wings is sitting on a branch"* .

that is solved. Here $(\boldsymbol{x_i}, y_i), i = 1, ..., l$ are instance label pairs. $\boldsymbol{x_i} \in \mathbb{R}^n$ is the feature vector of instance i and $y_i \in K$ states its category belonging, were K is the space of possible categories. $\boldsymbol{\omega} \in \mathbb{R}^n$ are the weights to be optimized, given the set of instance pairs. $C > 0$ is a penalty parameter. It is by default set to 1, but can be altered manually. $z(\omega, x_i, y_i)$ is called the loss function and this is what differentiate the classifiers. The first part of the problem formulation is also switched to the 1-norm when the the classifier is L1-regularized instead of L2-regularized.

Except for the initial testing part, mainly one classifier was used: the L1-regularized logistic regression. When the L1-regularized logistic regression is used, the problem to be solved is

$$\min_{\omega} \parallel \omega \parallel_1 + C \sum_{i=1}^{l} \log(1 + e^{-x_i \omega^T y_i}),$$

where $\| \cdot \|_1$ denotes the 1-norm. This was the classifier that was found to be best suited for this situation.

2.2.3 Constrained Parametic Min-Cuts for Automatic Object Segmentation

The Constrained Parametic Min-Cuts (CPMC) package is used for segmentation and is free for academic use. It can be found at [13]. The algorithms contained in the CPMC package produce a list of possible segments, given an image. The segmentation is done without prior knowledge about what the image might contain. All segments are given a score of how plausible they are. By plausible is meant how reasonable it is that the specified segment represents the boundary for e.g. an object or part of the scenery, like the sky, in the image. The score is based on a continuous model trained to rank object hypotheses.(6)

The CPMC package was used to segment the original images from the SAIAPR TC-12 dataset automatically (compared to manually). The segmentation masks were then used for classification of the targeted objects in the image.

2.2.4 Chunker

Chunking is a technique commonly used in natural language processing (NLP). It is a lighter form of parsing, also called Shallow parsing, where a string from either a natural or computer language is analyzed. A chunker is an algorithm that identifies the different parts of a sentence (e.g. nouns and verbs), but does not specify their internal relations. For instance, the output from a chunker given the image description in Figure 1 *"A yellow parrot with light blue wings is sitting on a branch"*, would be *"parrot"*, *"wings"* and *"branch"*.

In this thesis the texts accompanying each image were analyzed by a chunker. From the chunking, nouns were extracted for all the images. These nouns were later used as a base of what objects was desirable to find in the images. This part of the thesis was done in cooperation with a project in NLP, Computer Sciences, LTH. To read about the chunking process in detail see Hammarlund and Weegar [14].

2.3 Method

As mentioned earlier, the thesis can be divided into two parts. In this section the method for exploring text-aided classification of images, where the classification was examined with the addition of information of descriptions, is described first. After this the method for detection of objects through text, where given an image and an accompanying description relevant words from the text were found in the image, is described.

2.3.1 Method: Exploring text-aided classification of images

There are 255 different labels in the SAIAPR TC-12 dataset connected to the segmented images. The frequency of their occurrence differs a lot, which is illustrated in Figure 2. While the most common label in the dataset, "sky-blue", has 5717 entries, some labels like "dragonfly" and "viola" occur only once. Slightly more than half of the labels had less than 100 occurrences. This put restrictions on which labels that could be used or not, since a certain amount of examples are needed for training and testing a classifier (the exact amount is difficult to specify, since it differs).

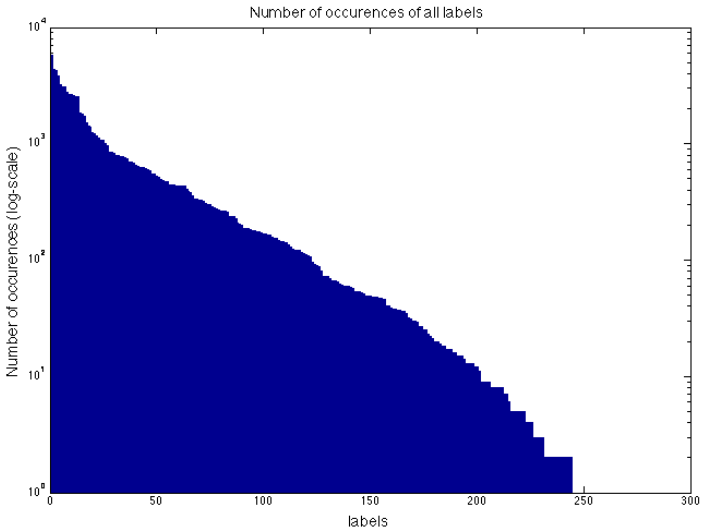

Figure 2: An image illustrating the number of occurrences of all the labels in the SAIAPR TC-12 dataset. Note the log-scale on the y-axis.

First, five labels were chosen as a starting point for the model. The labels "grass", "man", "rock", "sky-blue" and "trees" are among the most common in the data set and were also considered different enough to avoid mix-ups or overlapping meaning in an image (which, for example, the words "man" and "group-of-persons" might induce). All eight classifiers included in LIBLINEAR were tested ten times. For each label, in each classification, 1000 segments representing this label with their accompanying 27 image features were randomly chosen from the dataset. If more than one segment with the same label were

chosen from the same image one of them was dismissed and replaced by a new one. Only one instance of a specific label per image was allowed since some pictures had a lot of the same objects.

The chosen instances were divided so that 90 % from each label were used to train the model and the remaining 10 % were saved for testing it. The accuracy of each classification was calculated simply as the percentage correctly classified in the testing set. The result of the trials and how the classifiers compared respectively can be seen in Appendix B, Figure 10. L1-regularized logistic regression gave the highest accuracy in most of the trails, with L1-regularized L2-loss support vector classification as a close second.

The next step was to involve the image descriptions in the classification process. All words from all descriptions in the dataset were extracted to form a bag-of-words model. This gave each segment in the database 6865 new features, resulting in a total of 6892 features. The same method was applied again to compare the different classifiers. The result can also be viewed in Appendix B, Figure 10. Even in this test L1-regularized logistic regression gave the best results, with L2-regularized L2-loss support vector classification (primal) and L2-regularized logistic regression (primal) not far behind.

When the best-performing classifiers for each case had been found the result of the classification could be examined in more detail. The classification was done again, but this time only with the most high performing classifier. The results were, beside the total accuracy, displayed in a confusion matrix. A confusion matrix is a matrix were each column represents the instances classified as a certain category, while the rows represents the instances actually belonging to the category. Element a_{ij} in the matrix represents how many instances from category i were classified as j. The matrix is usually normalized along the rows, making it easier to quickly see how many of a certain category were correctly classified. By using confusion matrices the classification results for individual labels could be studied.

The same procedure was then applied to a set of ten labels. Added to the five ones mentioned earlier were "cloud", "ground", "group-of-persons", "vegetation" and "wall". In this case there were labels which could be overlapping, e.g. "ground" and "grass", but this was partly why they were chosen: to see if the classifier could differ between them. The result from the testing of classifiers can be seen in Appendix B, Figure 11. In both cases L1-regularized logistic regression gave the highest accuracy overall, even though other classifiers were close.

In the third round the number of labels was increased to 100. These were the 100 most common labels in the dataset and can be viewed in Appendix A. Since most of these labels didn't have 1000 occurrences among the segments, the number of instances representing each label had to be lowered to 60. Aside from this, the procedure carried out was the same as before and the results are displayed in Appendix B, Figure 12. Again, both with and without bag-of-words, L1-regularized logistic regression produced the best accuracy.

In the group of 100 labels there was a lot of similarity and overlapping meaning among the words. For example, among the most common labels were "sky", "sky-blue" and "sky-light". When looking at the pictures it became clear that what was labeled "sky-blue" in one image, was labeled "sky" in another. The fact that these unclear lines obstruct the classification process gave birth to the idea that the labels could be divided into clusters before classification took

place.

A few different set of clusters were tried out, with varying results. The clustering found most successful were based on the accompanying hierarchy of all labels in the SAIAPR TC-12 dataset. This resulted in clusters with similar labels and would by a human be considered as "natural" groups. The size of the groups was also taken into account, with the aim to make them as equally sized as possible. In some cases there was a trade off between equal size and natural groups. Four out of the 100 labels did not fit into any of the clusters naturally and were therefore discarded. This gave in total 96 labels divided into 13 clusters. The resulting division can be seen in Appendix A.

The classifiers were tried out on the 13 clusters as well and the outcome of this can be found in Appendix B, Figure 13. As these figures show, even here L1-regularized logistic regression performed the best.

When the best performing classifier had been chosen it was used to create a confusion matrix for each case. The classification was done using 10-fold cross validation, where the confusion matrix shows the mean value from these classifications. With these matrices showing how accurate each label was classified the results could be studied in more detail. The matrices can be found in Section 3.

2.3.2 Method: Detection of objects through text

The aim in the second part of the project was to construct an algorithm that given text and an automatically segmented image could find the relevant words in the image with a classifier pretrained on the 13 clusters. As mentioned earlier, given an image, the number of relevant objects to find is a matter of subjectivity. However, with a fixed text describing the image the number of relevant objects can be decided objectively, by using the text as a starting point.

The relevant words, or key-nouns, were extracted from the text with a chunker. Figure 3 shows an image from the database with its accompanying text. Even though there are backpacks on the shelf in the background, they are not present in the text and therefore impossible to choose as key-nouns. The process of classification was made with the 13 clusters to choose from. The relevant words from the text are often not one of these. This required a connection between the key-nouns and the clusters. WordNet is an online lexical database for the English language where for instance different metrics between words can be calculated. This was done between each of the extracted key-nouns and each of the cluster names, resulting in a similarity matrix containing all the metrics. To read more about WordNet and the metrics see [15].

Figure 3: An image from the SAIAPR TC-12 dataset with the accompanying text *"a woman with a red cloth on her head on a train"*. The words *"woman"* and *"cloth"* can be found both in the text and in the image, but the backpacks on the shelves are not mentioned in the text and are therefore not chosen as key-nouns.

Since the goal was to make the whole process as automatic as possible algorithms from the CPMC package were used to find segments in the images. At this stage, there was no information about what was to be found in the picture. The result was 500-1000 overlapping segments that the algorithm found more or less plausible as segments. All of the segments were classified using logistic regression. For each segment, the probabilities that it represented each of the different clutsers could hence be retrieved. Again, the information was stored in a matrix, called the segment matrix, which had the size the number of extracted segments for the given image times number of clusters.

The next step was to combine the similarity matrix with the segment matrix by multiplying them. The result, called the probability matrix, gives the probability that a certain segment represents a certain key-noun, for all segments and all key-nouns. The logarithm of the values in the probability matrix was used, since we can use addition instead of multiplication for the probabilities. This simplifies calculation, but the optimized result is the same. From all these probabilities, the segments best suited for each of the key-nouns in the current image was to be found. To do this, an optimization algorithm that solves an altered version of the assignment problem was used. The number of key-nouns to be found was between 2 and 17, thus much lower than the number of segments. This was the reason for an altered version of the assignment problem. All of the key-nouns need to be assigned to one segment, but all segments do not correspond to a word. An extra category was introduced which could hold all segments without an assigned word. Also, each segment was only allowed to

be assigned to one word.

Some of the relevant words were often not possible to find in the image either, e.g. 'view'. Therefore an extra category was introduced among the segments as well, where words that were by the chunker considered relevant, but to abstract to be visually relevant. Assume there are X segments and Y key-nouns. The number of agents is $X+1$, with all agents having the supply 1 except one which has supply Y. The number of tasks is $Y+1$, with all tasks having the demand 1, except one which has demand $X-Y$.

For validation of the classification a number of images were manually annotated with respect to the key-nouns. These were used as ground truth to compare with the system's annotated segments. In total there were 466 relevant words, where both the system and the human annotator found a segment. The evaluation was then done using a Jaccard index. The Jaccard index is defined by

$$J(A,B) = \frac{\mid A \cap B \mid}{\mid A \cup B \mid},$$

where A is the set of pixels for the ground truth mask and B is the set of pixels in the mask produced by the system.

Another approach was also carried out where WordNet was not used to connect the relevant words with the clusters, but instead each word was manually assigned a cluster. Again the extra category for words not possible to find was present. With this approach it was possible to get a better understanding of how the classification process worked, without being compromised by possible errors from the generated connections. This however came at the cost of making the process less automated.

While the set of automatically generated segmentations was used for the classification task, the process of generating the possible segments automatically was not altered in this project. With this in mind, it was also of interest to try the classification without interference from potential faults in the segments. If, for instance, there is no exact match between the manual and the automated segmentation the result can never be 100% correct, even though the best segment of the ones available is chosen. Thus, in this approach the segment from the CPMC pool most resembling the manually annotated ground truth segment was chosen as the new ground truth. Since the classification was the work of this project, this approach gave a better insight to how well it worked, given "optimal" segments.

3 Results

3.1 Results: Exploring text-aided classification of images

Below are the confusion matrices for 5, 10 and 100 labels as well as 13 clusters. Both when only image features were used and when they were used together with bag-of-words are displayed. In each case the classifier that performed the best in the test of all classifiers was used to produce the matrix. In the case of 100 labels the results are presented in a colored surface plot instead, to give a better overview. Appendix C contains the standard deviations for the 10-fold cross validation in each case.

	grass	man	rock	sky-blue	trees
grass	69	6	12	2	11
man	3	80	10	3	5
rock	7	10	72	2	8
sky-blue	2	3	3	90	2
trees	12	7	10	3	68

Table 1: Confusion matrix for the recognition of 5 labels with image features, estimated with 10-fold cross validation. Accuracy: 75.8.

	grass	man	rock	sky-blue	trees
grass	73	6	9	3	10
man	4	85	6	3	3
rock	8	8	77	2	5
sky-blue	3	3	2	90	2
trees	12	4	4	3	78

Table 2: Confusion matrix for the recognition of 5 labels with image features and bag-of-words, estimated with 10-fold cross validation. Accuracy: 80.5.

	cloud	grass	ground	g.o.p.*	man	rock	sky-blue	trees	vegetation	wall
cloud	77	1	2	2	2	2	8	2	1	3
grass	2	54	13	4	4	4	1	6	6	6
ground	3	6	63	4	2	13	1	1	1	6
g.o.p.*	1	2	4	67	14	5	1	2	0	4
man	2	2	2	13	66	5	1	2	1	5
rock	3	3	25	6	7	38	1	5	3	10
sky-blue	11	1	1	1	2	2	79	2	0	2
trees	2	9	4	5	6	4	1	45	20	5
vegetation	1	15	4	2	4	4	1	22	42	5
wall	7	4	6	5	10	8	3	5	3	49

Table 3: Confusion matrix for the recognition of 10 labels with image features, estimated with 10-fold cross validation. Accuracy: 58.5 . *group-of-persons

	cloud	grass	ground	g.o.p.*	man	rock	sky-blue	trees	vegetation	wall
cloud	75	3	2	2	2	1	11	2	1	1
grass	3	56	12	3	4	4	2	6	10	3
ground	3	9	62	3	3	12	1	2	3	3
g.o.p.*	1	2	2	70	11	3	1	2	1	7
man	2	2	2	12	65	5	1	2	1	8
rock	3	4	18	3	6	53	1	2	5	4
sky-blue	11	2	1	1	1	1	79	2	0	1
trees	2	7	2	4	2	2	1	64	16	1
vegetation	2	14	4	2	3	5	1	15	53	2
wall	2	2	3	7	9	3	1	1	2	68

Table 4: Confusion matrix for the recognition of 10 labels with image features and bag-of-words, estimated with 10-fold cross validation. Accuracy: 64.3. *group-of-persons

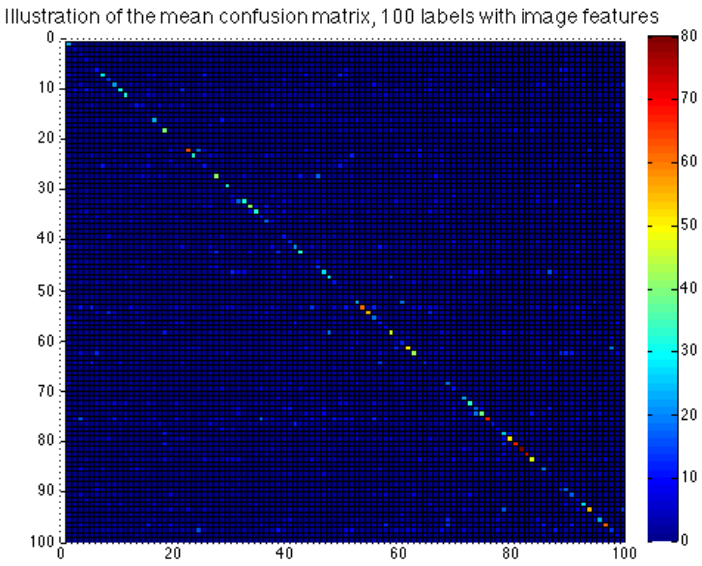

Figure 4: The results from 10-fold cross validation for 100 labels with image features. The mean confusion matrix is illustrated with a plot. Note that over half of the values in the confusion matrix are strictly below 15 %. Accuracy: 19.5%.

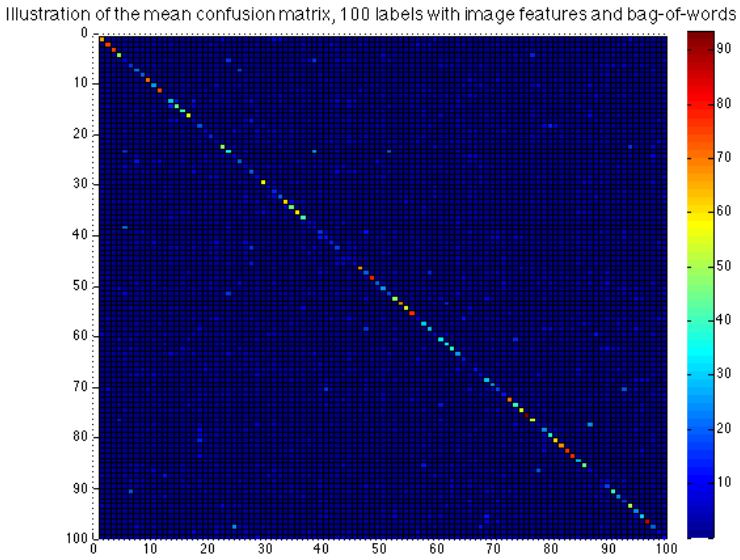

Figure 5: The results from 10-fold cross validation for 100 labels with image features and bag-of-words. The mean confusion matrix is illustrated with a plot. Note that half of the values in the confusion matrix are strictly below 25 %. Accuracy: 31.5 %.

	water	sky	veget.	constr.	human	h.o.*	ground	animal	vehicle	mount.	road	floor	fabrics
water	38	6	8	17	2	22	4	0	0	0	1	0	0
sky	4	56	4	12	1	17	2	0	0	2	1	0	0
veget.	3	3	46	12	5	29	2	0	0	0	0	0	0
constr.	3	6	14	37	6	31	2	0	0	0	1	0	0
human	2	3	9	10	24	50	1	0	0	1	0	1	1
h.o.*	3	3	10	12	6	64	1	0	0	0	0	1	0
ground	15	5	11	19	1	22	23	0	0	0	3	1	0
animal	3	5	13	25	5	43	3	0	0	0	1	0	0
vehicle	5	5	11	14	12	48	3	0	1	1	0	0	0
mount.	16	7	10	18	3	22	1	0	1	23	0	0	0
road	20	5	9	12	6	30	11	0	1	2	3	1	0
floor	16	1	6	12	3	35	9	0	0	0	2	18	0
fabrics	5	3	10	14	15	50	1	0	1	0	0	1	1

Table 5: Confusion matrix for the recognition of the 13 visual categories with image features, estimated with 10-fold cross validation. Accuracy: 36.3. *house object

	water	sky	veget.	constr.	human	h.o.*	ground	animal	vehicle	mount.	road	floor	fabrics
water	59	6	7	3	3	4	5	1	5	5	2	0	1
sky	7	53	7	12	2	6	3	2	1	3	1	1	1
veget.	4	3	54	13	6	11	2	1	2	1	2	0	0
constr.	2	3	12	50	5	19	1	1	2	0	2	1	2
human	3	2	6	8	47	24	1	2	3	1	1	1	3
h.o.*	1	2	5	11	9	65	0	1	1	0	2	2	2
ground	16	5	13	7	1	3	40	1	2	4	7	1	0
animal	5	5	10	13	3	9	1	48	2	0	3	0	2
vehicle	10	3	9	9	9	11	1	1	37	1	8	0	2
mount.	23	7	10	5	3	0	9	3	2	33	5	0	0
road	5	2	9	22	5	10	10	1	5	2	26	1	1
floor	2	1	5	16	6	46	1	0	2	0	2	17	2
fabrics	1	2	5	12	19	46	0	1	1	0	0	1	12

Table 6: Confusion matrix for the recognition of the 13 visual categories with image features and bag-of-words, estimated with 10-fold cross validation. Accuracy: 49.2. *house object

3.2 Results: Detection of objects through text

The Jaccard indices were 0.09 (automatically connected words and clusters), 0.15 (manually connected words and clusters) and 0.53 (optimal CPMC segment choice).

Figure 6 shows the Jaccard indices for the classification, both when the connections between words and clusters were done manually and when they were done automatically. It also displays the approach were segments from the CPMC pool were chosen as ground truth instead of the manually annotated ones.

Figure 7 shows an image where segmentation was done and which regions that were chosen by the system as the representation of the words. Figure 8 also displays some of the classified segments. In some cases the algorithm produced segments that were more correct than the manually annotated ground truth. This can for example be seen with the second segment ("sky") in Figure 8.

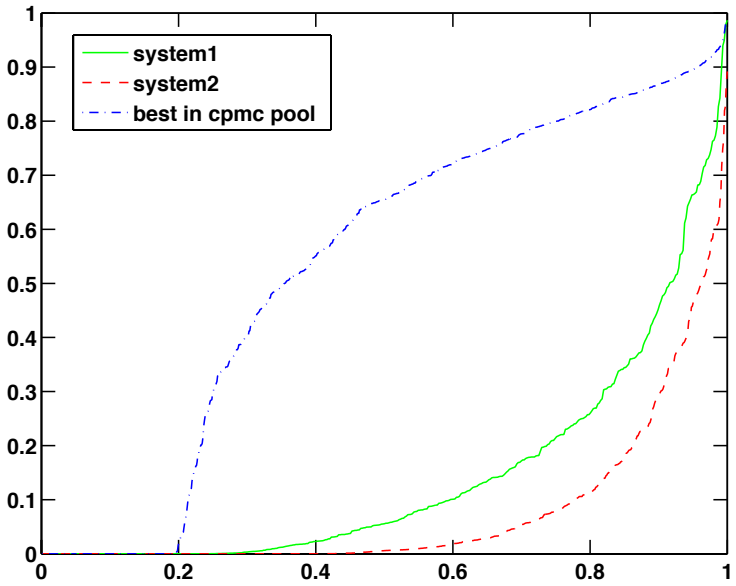

Figure 6: Jaccard index for each of 466 segmented key-noun regions and corresponding ground truth segmentation. System 1 is the result for manual correspondence between words and visual categories. System 2 is the result using calculated distances between words and visual categories using WordNet. The blue dash-dotted line is the result if one could, for each key-noun, select the optimal segment among the pool of segments from the CPMC segmentation.

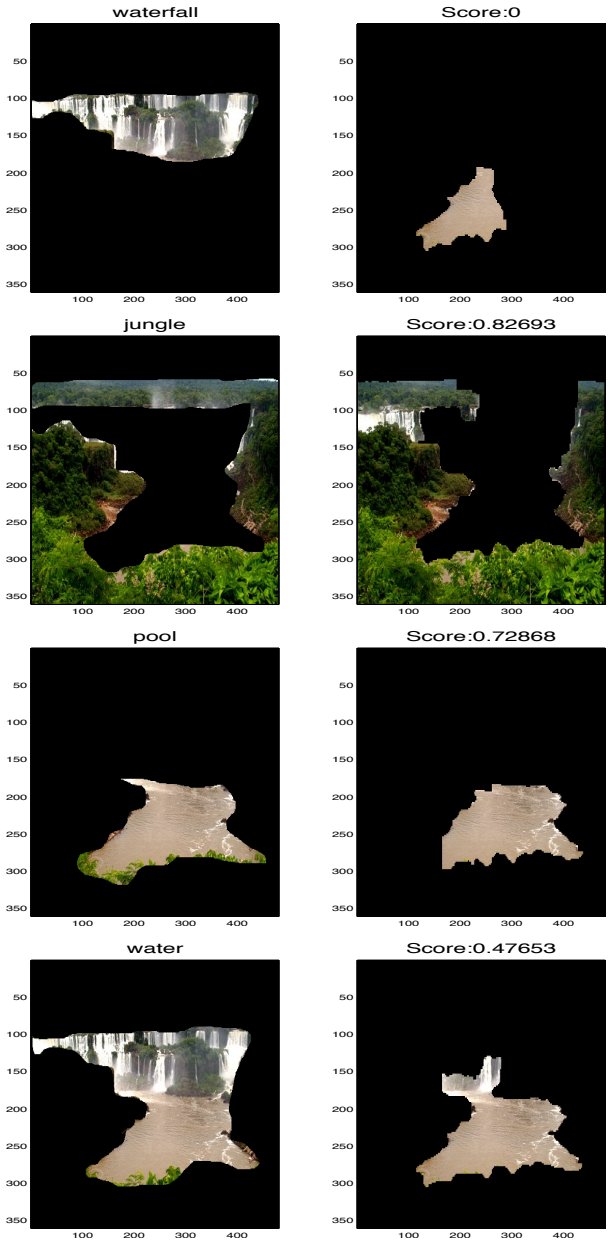

Figure 7: Results of segmentation and recognition of the words *waterfall*, *jungle*, *pool* and *water* from the annotation *A cascading waterfall in the middle of the jungle; front view with pool of dirty water in the foreground.* To the left is shown ground truth segmentation with corresponding key-noun. To the right is shown the system output with the matching score according to the Jaccard index.

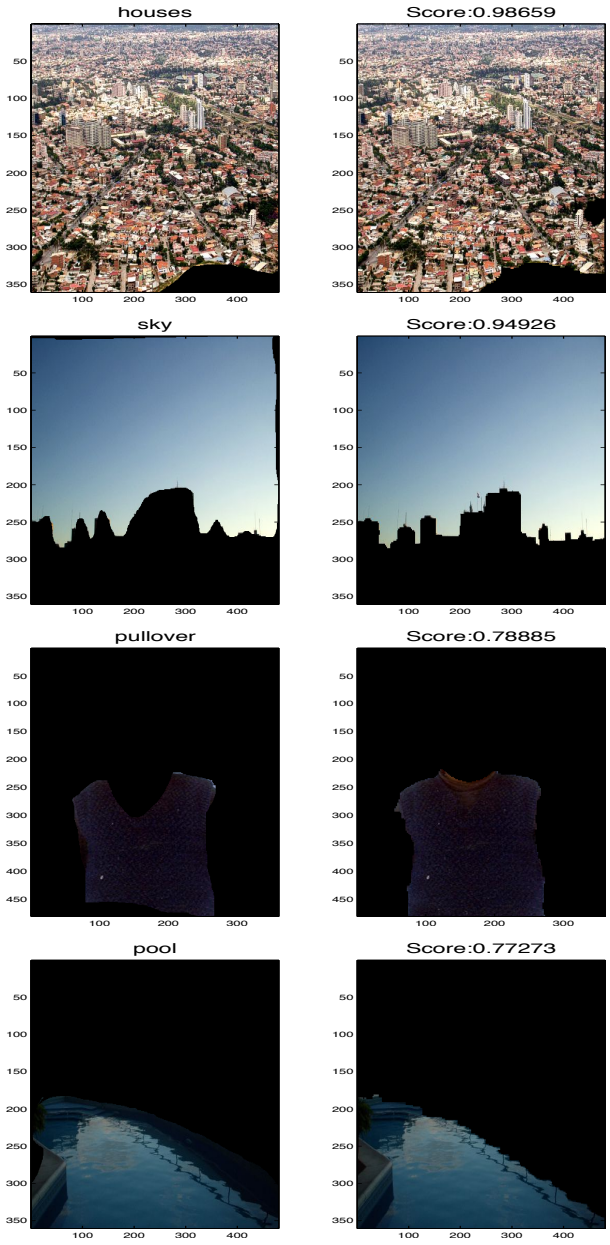

Figure 8: Results of segmentation and recognition of the words 'Houses', 'Sky', 'Jungle', 'Pullover', 'Pool'. Left column shows the manually segmented ground truth segmentations together with the corresponding key-noun. Right column shows the response from the system described in the paper, together with the matching score of the Jaccard index.

4 Discussion

In all of the trials with different numbers of labels or clusters, the total accuracy improved with the addition of a bag-of-words model. If the confusion matrices are studied in detail it becomes clear that some labels improved more than others. Certain labels' accuracy actually decreased with the addition.

In the case of five labels the classification process gave decent results even when only image features were used. The low number of categories and each category's distinctive visual features compared to one another, decreased potential overlapping of concepts. The large number of instances, which resulted in a larger training set, of each category probably also helped the classification. The only category that did not improve by adding the information from text was "sky-blue". When looking at manually written descriptions of images it becomes clear that concepts like "sky" tend to not be mentioned, unless something out of the ordinary is happening with it, like a beautiful sunset. The word "sky" is something that is part of the background in most images and a human annotator focuses more on mentioning e.g. a "man". If, for instance, it becomes clear from the description that the setting is outdoors, the mentioning of "sky" might seem redundant to a human. The categories that had a tendency to be wrongly classified were "grass" as "trees" and vice versa. Both being a type of vegetation this is understandable.

When adding five labels, the distinction between the labels became a little less clear. Most of the labels did not change significantly when the extra features were added. The features that did however were "rock", "trees", "vegetation" and "wall", which all improved with the additional features. Meaning that the ones that did not change significantly were "cloud", "grass", "ground", "group-of-persons", "man" and "sky-blue". The labels "cloud", "grass", "ground" and "sky-blue" could all be considered as often belonging to the background and with the idea presented when discussing the five labels case, their unaltered accuracy seems reasonable. More surprising is the fact that "group-of-persons" and "man" did not improve much. These two categories had a tendency to be classified as each other though, which might have complicated the classification. A man can obviously be a part of a group of persons, which raises the question if it even should be considered completely wrong to classify a group of persons as "man". Along the same line it is worth mentioning the high number of instances labeled "rock" that were classified as "ground", both with and without the bag-of-words model. If a small rock lies on the ground, should it be classified as "rock" or "ground"? This is a matter of subjectivity. It became clear already at ten labels that distinguishing categories visually enters philosophical territory and can be difficult even for a human.

With 100 labels the number of labels with a similar meaning increased a lot, along with difficulties in distinguishing them. In most cases the probability of a label being correctly classified was low. There were a few labels that got high scores though. Interestingly the labels with high scores when only image features were used were not the same as the ones with high scores when they were used in combination with a bag-of-words model. In the first case, when image features were used, "seal", "sky-light", "sky-night" and "sky-red-sunset-dusk" were the four labels with over 65% of their instances correctly classified. In the second case with the bigger feature vector the labels with highest accuracy were "horse", "llama", "seal", "snow" and "waterfall", all of which had an accuracy

higher than 75%. The only one of the labels in the first case with an accuracy higher than 75% was "sky-night".

For some labels adding the bag-of-words model really changed their accuracy. The biggest improvement could be seen with the labels "bicycle", "bird", "horse" and "llama". All of them had an increase of more than 50 percentage points. As a contrast to this, "cloud", "curtain", "face-of-person", "mountain" and "painting" decreased with more than 15 percentage points. The labels with a high increase are words that are specific: if the word "llama" is mentioned in the text the probability of the word meaning something else than the animal llama is low. Also, if an image contains a llama (or another animal for that matter) it will probably be mentioned in the description, since it is the type of thing humans tend to focus on.

The overall lower accuracy obtained with 100 labels is probably not only due to the overlapping concepts, but also to the fact that only 60 instances of each label were used, compared to 1000 instances for five and ten labels.

When classifying the 13 clusters, the variation among the instances in each category was bigger than before, since each category contained a set of different labels. This resulted in very low accuracy for some categories when only image features were used. All of them had a high tendency to be classified as "house objects". This category contains a high number of labels compared to some of the others, which probably trained the classifier to assume a lot of different segments were a part of this cluster. With the added bag-of-words model almost everyone of the categories improved significantly. Most improvement could be seen in the clusters "vehicle" (which went from 1% to 37%) and animal (which went from 0% to 48%).

What can be said about the confusion matrices for 13 clusters is that the labels in some clusters probably not were that similar visually, but more so conceptually. The, in most cases, much improved result when adding a bag-of-words model would support this.

The aim of the second part of the thesis, to design an algorithm that could, given an image with a description, find key-nouns from the text and mark their presence in the image, was fulfilled. An automatic version was constructed, but to test the actual classification process some manual adjustments were made as well. Considering all the steps taken in the algorithm the final result of the Jaccard indices are not bad. When constructing the algorithm many questions were raised and decisions concerning design had to be made. One of the paramount decisions being the clustering of the labels. Even though different types of clustering were tested, there are obviously a lot more to try out! It is important to mention that when constructing the clusters the accuracy from classification trials was not what they were based on. It would be possible to do "optimal clusters" and only look at the accuracy rates. However, that would adapt the algorithm a little too well to this particular data set. If other images were to be included the results might not be valid at all.

Another problem with clustering labels together arises when we for instance have a picture with a bear and a dog, both mentioned in the description. Both belong to the category "animals" and if the classification process works alright that is what they will be classified as. There is no distinction made in the algorithm between the two words "bear" and "dog", meaning that the two animal segments and the two animal words will be randomly assigned. Still, the human annotator, and thus the ground truth, do distinguish between the

Figure 9: Two segmented images from the SAIAPR TC-12 data set. The rightmost one has been manually segmented with more detail.

two. Thus, the result might be wrong, even though the classifier did everything right. This problem would be something to look into in the future if the work was continued.

A drawback when it comes to the training process of the classifier is the quality of the manually segmented images in the data set. The problem is that the details of the segmentations are not consistent. In Figure 9 two different segmented images from the data set are shown. The leftmost one only have three regions: two "trunk" and one "vegetation" describing the bottom half of the image. The sky and the remaining trees do not have a segment. In the right image all parts have been assigned a segment, even the clouds each get an individual segment.

Deciding what is ground truth in an image is also a matter of subjectivity. In the introduction the example of the view of a city was made. Regardless of how one decides to do the segmentation (e.g. seen as a "city" or separating the regions between the individual houses), it is important to be consistent to obtain optimal results from the classification.

In Hammarlund and Weegar [14] connections between words were extracted. If the description was *"A hole in the ground"* the preposition "in" links the two key-nouns "hole" and "ground". This could give a lot of extra useful information in the classification process. Instead of just stating that there is a hole somewhere in the picture, it is established that this object is also connected to the other key-noun "ground". With this information extra restrictions are put on the segments that should represent "hole" as well as "ground".

Another opportunity for future work would be to look into the visual features used. These were predefined in the SAIAPR TC-12 dataset and had already been calculated for all the images. If given time they could be tested individually to see if it was possible to discard some and/or find new visual features to add to the feature vector.

5 Conclusions

Classification of objects with only access to visual information was compared to classification when information found in an accompanying text made by a human annotator also was available. In general the information found in image descriptions improved object classification. Labels that did not improve significantly were typically the type of objects traditionally included in the background.

An algorithm that classifies objects in an image with starting point in an image description was developed. When the process was less automated the results improved. However, with further research it is possible the results will improve, while still keeping the system automatic.

The results presented in this thesis have given rise to several new ideas for continued research.

A Appendix: Labels and clusters

A.1 List of 100 labels

A list of the 100 labels:
sky-blue, man, group-of-persons, ground, grass, cloud, rock, vegetation, sky, trees, wall, woman, mountain, ocean, sky-light, window, building, tree, person, couple-of-persons, floor, house, car, face-of-person, street, fabric, plant, hat, hill, sand-beach, lamp, sidewalk, floor-other, city, river, door, chair, bush, bed, public-sign, bottle, child-boy, table, palm, water, wooden-furniture, lake, child-girl, painting, snow, cloth, trunk, highway, glass, fence, ruin-archeological, church, bicycle, sand-dessert, wood, curtain, head-of-person, roof, branch, dish, flag, boat, road, stairs, column, floor-wood, non-wooden-furniture, edifice, hut, sky-night, kitchen-pot, cactus, horse, water-reflection, hand-of-person, paper, generic-objects, sky-red-sunset-dusk, statue, waterfall, plant-pot, leaf, bird, seal, handcraft, llama, construction-other, flower, fruit, castle, flowerbed, fountain, ship, umbrella, monument

A.2 List of labels in the 13 clusters

The division of 96 labels into 13 clusters and the four discarded labels:

water
water
water-reflection
lake
ocean
river
waterfall
snow

sky
cloud
sky
sky-blue
sky-light
sky-night
sky-red-sunset-dusk

vegetation
fruit
vegetation
cactus
flowerbed
flower
plant
leaf
trees
branch
trunk
bush

palm
tree

construction
construction-other
edifice
column
roof
stairs
wall
ruin-archeological
building
castle
church
house
hut
monument
fountain
statue

human
couple-of-persons
group-of-persons
person
face-of-person
hand-of-person
head-of-person
child-boy
child-girl
man
woman

house objects
bed
chair
door
fence
non-wooden-furniture
wooden-furniture
table
window
handcraft
painting
generic-objects
lamp
paper
plant-pot
public-sign
bottle
glass
hat

kitchen-pot
umbrella

ground
ground
sand-beach
sand-dessert
grass

animal
bird
horse
llama
seal

vehicle
bicycle
car
boat
ship

mountain
mountain
hill

road
road
highway
sidewalk
street

floor
floor
floor-other
floor-wood

fabrics
fabric
cloth
curtain
flag

discarded:
rock
dish
wood
city

B Appendix: Classifiers

The classifiers used in the testing were:
classifier 1: L2-regularized L2-loss support vector classification (dual)
classifier 2: L2-regularized L2-loss support vector classification (primal)
classifier 3: L2-regularized L1-loss support vector classification (dual)
classifier 4: support vector classification by Crammer and Singer
classifier 5: L1-regularized L2-loss support vector classification
classifier 6: L1-regularized logistic regression
classifier 7: L2-regularized logistic regression (dual)
classifier 8: L2-regularized logistic regression (primal)

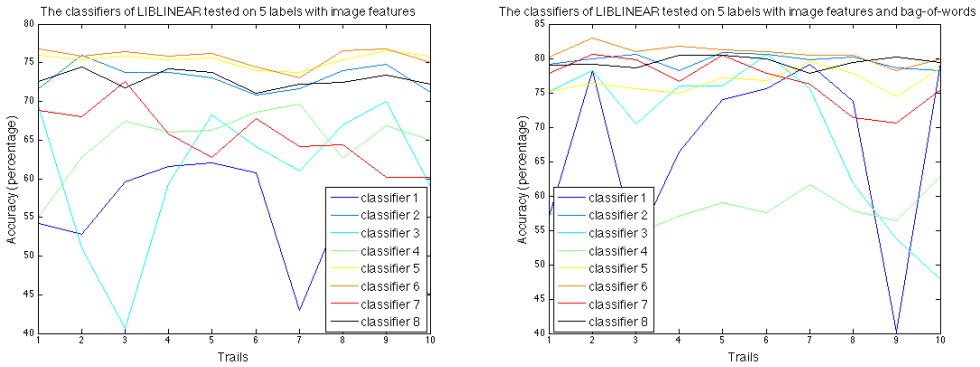

Figure 10: The eight classifiers included in LIBLINEAR were tested ten times each on five labels from the data set. The picture to the left depicts when only image features were used, while the image to the right shows when both image features and bag-of-words were used. Classifier 6 was found to give the highest accuracy.

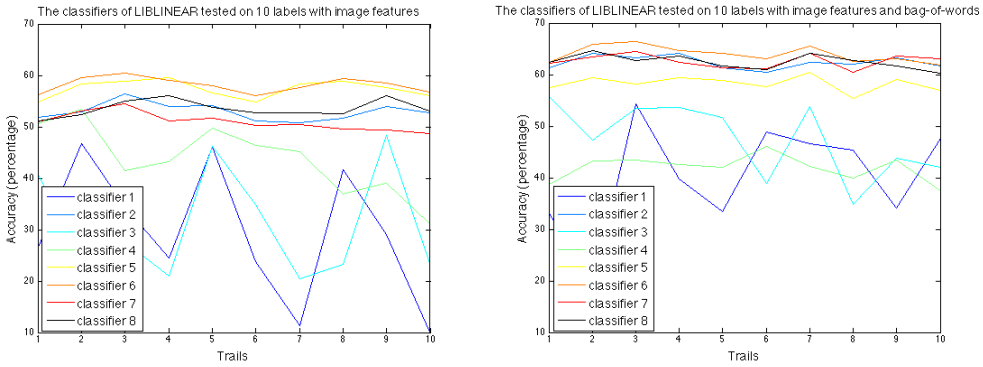

Figure 11: The eight classifiers included in LIBLINEAR were tested ten times each on ten labels from the data set. The picture to the left depicts when only image features were used, while the image to the right shows when both image features and bag-of-words were used. Classifier 6 was found to give the highest accuracy.

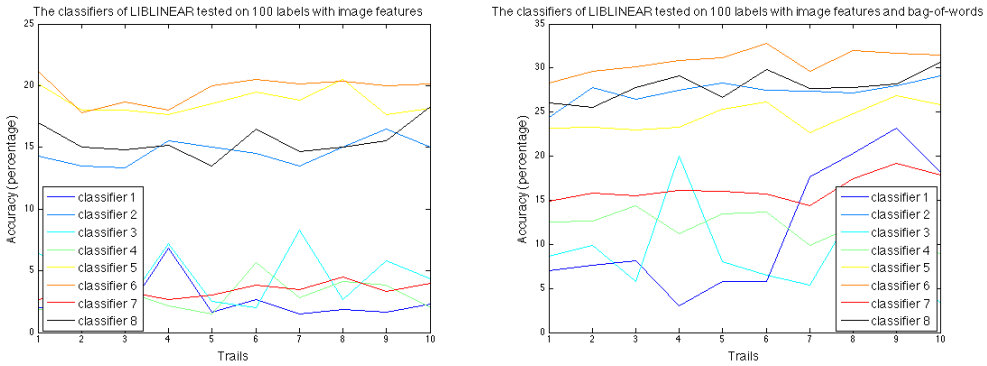

Figure 12: The eight classifiers included in LIBLINEAR were tested ten times each on 100 labels from the data set. The picture to the left depicts when only image features were used, while the image to the right shows when both image features and bag-of-words were used. Classifier 6 was found to give the highest accuracy.

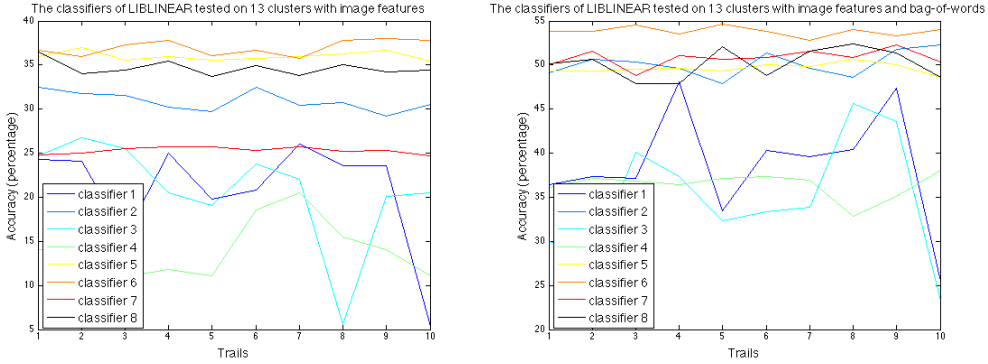

Figure 13: The eight classifiers included in LIBLINEAR were tested ten times each on 13 clusters constructed from labels from the data set. The picture to the left depicts when only image features were used, while the image to the right shows when both image features and bag-of-words were used. Classifier 6 was found to give the highest accuracy.

C Appendix: Standard deviations for Section 3.1

	grass	man	rock	sky-blue	trees
grass	3	2	3	1	3
man	2	2	3	1	2
rock	3	3	4	2	3
sky-blue	1	2	1	3	2
trees	2	2	4	1	4

Table 7: Standard deviation for the 10-fold cross validation with image features of the 5 labels.

	grass	man	rock	sky-blue	trees
grass	4	2	3	1	4
man	2	3	3	1	2
rock	2	3	4	1	2
sky-blue	2	1	1	4	2
trees	3	1	3	2	3

Table 8: Standard deviation for the 10-fold cross validation with image features and bag-of-words of the 5 labels.

	cloud	grass	ground	g.o.p.*	man	rock	sky-blue	trees	vegetation	wall
cloud	4	1	1	2	1	1	2	1	1	1
grass	1	5	3	2	2	2	1	2	2	3
ground	2	2	4	2	2	4	1	1	1	3
g.o.p.*	1	2	2	5	3	3	1	1	1	2
man	1	1	2	2	4	1	1	1	1	2
rock	1	1	4	2	3	5	1	2	2	3
sky-blue	3	1	1	1	1	2	5	2	0	2
trees	1	3	2	2	2	3	1	5	4	2
vegetation	1	2	2	2	1	2	1	2	2	2
wall	3	2	2	2	2	3	2	1	1	5

Table 9: Standard deviation for the 10-fold cross validation with image features of the 10 labels. *group-of-persons

	cloud	grass	ground	g.o.p.*	man	rock	sky-blue	trees	vegetation	wall
cloud	6	2	1	1	1	1	4	1	1	1
grass	2	5	4	1	2	2	1	3	3	2
ground	2	2	5	1	2	4	1	1	2	2
g.o.p.*	1	1	1	7	3	1	1	2	1	3
man	1	1	1	4	5	2	1	1	1	2
rock	2	2	4	2	4	7	0	1	3	3
sky-blue	2	1	1	1	1	1	4	2	1	1
trees	1	3	1	1	1	1	1	4	4	1
vegetation	1	3	2	1	2	2	1	4	3	1
wall	1	2	2	3	3	2	1	1	1	6

Table 10: Standard deviation for the 10-fold cross validation with image features and bag-of-words of the 10 labels. *group-of-persons

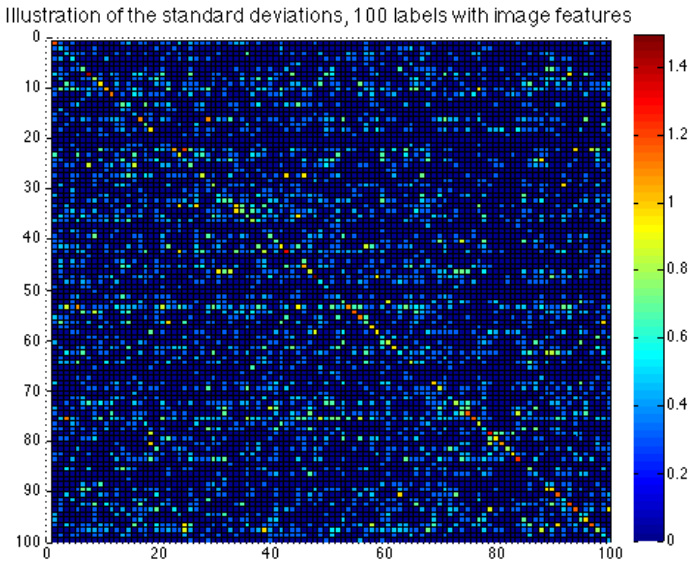

Figure 14: The standard deviation from 10-fold cross validation for 100 labels with image features, illustrated with a plot.

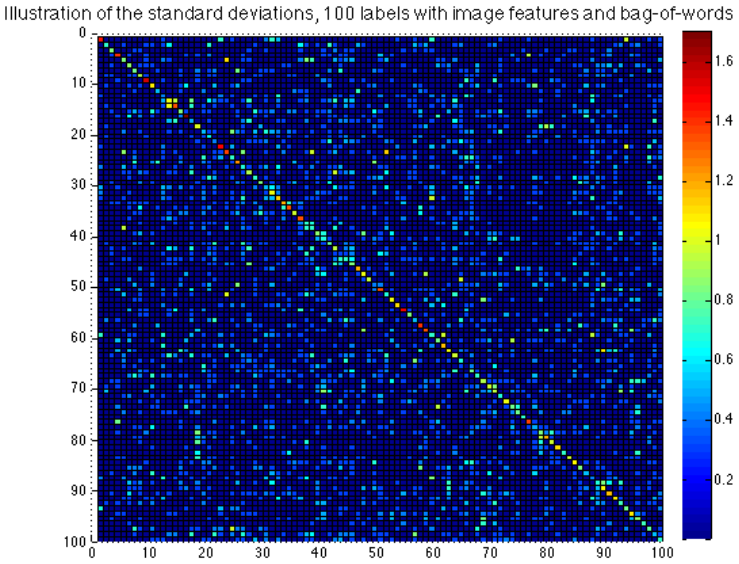

Figure 15: The standard deviation from 10-fold cross validation for 100 labels with image features and bag-of-words, illustrated with a plot.

	water	sky	veget.	constr.	human	h.o.*	ground	animal	vehicle	mount.	road	floor	fabrics
water	2	1	2	2	1	4	1	0	0	0	1	0	0
sky	1	2	1	2	1	2	1	0	0	1	0	0	0
veget.	2	1	4	3	3	2	1	0	0	0	0	0	0
constr.	1	1	5	3	3	3	1	0	0	1	1	1	0
human	1	1	2	2	3	4	0	0	0	1	0	0	0
h.o.*	2	2	3	5	2	6	1	0	0	0	1	1	0
ground	1	1	2	3	0	2	2	0	0	0	1	1	0
animal	1	1	1	2	1	2	1	0	0	0	0	0	0
vehicle	1	1	1	2	2	2	1	0	1	0	0	0	0
mount.	1	1	1	1	0	1	0	0	0	1	0	0	0
road	3	1	1	1	1	2	2	0	0	1	1	0	0
floor	2	0	1	1	1	2	2	0	0	0	1	2	0
fabrics	1	1	1	1	1	2	0	0	0	0	0	0	0

Table 11: Standard deviation for the 10-fold cross validation with image features of the 13 visual categories. *house object

	water	sky	veg.	constr.	human	h.o.*	ground	animal	vehicle	mount.	road	floor	fabrics
water	4	1	2	1	2	1	1	1	2	1	1	0	0
sky	1	2	1	2	1	1	1	1	1	1	1	0	1
veg.	2	1	5	3	3	3	1	1	1	1	1	0	0
constr.	1	2	4	3	2	3	1	1	1	0	2	1	1
human	1	1	2	2	4	3	0	1	2	1	1	0	1
h.o.*	1	1	2	4	3	5	1	1	1	0	1	1	2
ground	2	1	1	1	0	1	2	0	1	1	1	0	0
animal	1	1	2	2	1	1	0	3	1	0	1	0	1
vehicle	2	1	1	1	1	1	1	1	2	1	1	0	1
mount.	2	1	0	1	0	0	1	1	1	2	1	0	0
road	1	1	1	3	1	1	2	0	1	1	2	0	1
floor	1	1	1	1	1	3	0	0	0	0	0	2	0
fabrics	0	1	1	1	2	1	0	0	0	0	0	0	1

Table 12: Standard deviation for the 10-fold cross validation with image features and bag-of-words of the 13 visual categories. *house object

References

[1] M. W. Passer and R. E. Smith, *Psychology: The science of mind and behavior*. McGraw-Hill, 2004.

[2] P. F. Felzenszwalb and O. Veksler, "Tiered scene labeling with dynamic programming," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pp. 3097–3104, IEEE, 2010.

[3] B. Taylor, A. Ayvaci, A. Ravichandran, and S. Soatto, "Semantic video segmentation from occlusion relations within a convex optimization framework," in *Energy Minimization Methods in Computer Vision and Pattern Recognition*, pp. 195–208, Springer, 2013.

[4] O. Chum, J. Philbin, J. Sivic, M. Isard, and A. Zisserman, "Total recall: Automatic query expansion with a generative feature model for object retrieval," in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pp. 1–8, IEEE, 2007.

[5] V. Moscato, A. Picariello, F. Persia, and A. Penta, "A system for automatic image categorization," in *Semantic Computing, 2009. ICSC'09. IEEE International Conference on*, pp. 624–629, IEEE, 2009.

[6] D. Medved, P. Nugues, F. Jiang, K. Åström, and M. Oskarsson, "Combining text semantics and image geometry to improve scene interpretation," in *ICPRAM*, 2014.

[7] Collaborative, "Logistic regression." http://en.wikipedia.org/wiki/Logistic_regression. Accessed: 2013-12-11.

[8] Collaborative, "Bag-of-words model." http://en.wikipedia.org/wiki/Bag-of-words_model. Accessed: 2013-12-11.

[9] K. Holmberg, *Optimering*. Liber, 2013.

[10] H. J. Escalante, C. Hernández, J. Gonzalez, A. López, M. Montes, E. Morales, E. L. Sucar, and M. Grubinger, ""the segmented and annotated iapr tc-12 benchmark"," *Computer Vision and Image Understanding*, 2009.

[11] M. Grubinger, C. P. D., H. Müller, and T. Deselaers, "Constraint enforcement in structure and motion applied to closing and open sequence," in *Proc. Asian Conf. on Computer Vision, Jeju Island, Republic of Korea*, 2004.

[12] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "LIBLINEAR: A library for large linear classification," *Journal of Machine Learning Research*, vol. 9, pp. 1871–1874, 2008.

[13] J. Carreira and C. Sminchisescu, "Constrained parametric min-cuts for automatic object segmentation, release 1." http://sminchisescu.ins.uni-bonn.de/code/cpmc/.

[14] L. Hammarlund and R. Weegar, "Visual entity linking," 2014. Computer Sciences, Lund University, Sweden.

[15] "About wordnet." http://wordnet.princeton.edu, 2010. WordNet. Princeton University.

Master's Theses in Mathematical Sciences 2014:E5
ISSN 1404-6342

LUTFMA-3258-2014

Mathematics
Centre for Mathematical Sciences
Lund University
Box 118, SE-221 00 Lund, Sweden

http://www.maths.lth.se/