Technical Report

Pattern Recognition and Image Processing Group
Institute of Computer Aided Automation
Vienna University of Technology
Favoritenstr. 9/1832
A-1040 Vienna AUSTRIA
Phone: +43 (1) 58801-18351
Fax: +43 (1) 58801-18392
E-mail: stefan.kuthan@gmx.at
URL: http://www.prip.tuwien.ac.at/

PRIP-TR-101 20th December 2005

# Extraction of Attributes, Nature and Context of Images

*Stefan Kuthan*

**Abstract**

In this thesis a framework for deriving high-level scene attributes from low-level image features is developed. Examples of attributes derived are photo-painting, indoor-outdoor, night-day and nature-city. The assignment of the attributes to images is done by a hierarchical classification of the low level features, which capture colour, texture and spatial information. A concise summary of current research and methods used in this field of research is given. Furthermore, a prototype for image classification is implemented, which aids in the evaluation of the different methods available. Training and test images are provided by the ImagEVAL project, a French computer vision evaluation project.

# Contents

**1 Introduction** **1**

**2 Literature Review** **4**

2.1 Classification by Attributes . . . . . 5

2.1.1 Colour Photo - Artistic Reproduction . . . . . 6

2.1.2 Indoor - Outdoor . . . . . 7

2.1.3 Urban - Natural . . . . . 8

2.2 Classification Systems . . . . . 9

2.2.1 Hierarchical Classification . . . . . 9

2.2.2 Content-Based Image Retrieval . . . . . 9

2.2.3 Linguistic Indexing . . . . . 11

2.2.4 Semantic Classification . . . . . 11

2.3 Features . . . . . 12

2.3.1 Colour . . . . . 13

2.3.2 Texture . . . . . 15

2.3.3 Region Matching . . . . . 16

2.3.4 Spatial Information . . . . . 17

2.4 Statistical Models . . . . . 18

2.4.1 Bayes Decision Rule . . . . . 18

2.4.2 $k$-Nearest Neighbour . . . . . 19

2.4.3 Decision Trees . . . . . 20

2.4.4 Multiresolution Hidden Markov Model . . . . . 20

2.4.5 Dimensionality Reduction . . . . . 21

**3 A System for Image Attribute Classification** **22**

3.1 Software . . . 22

3.2 Feature Extraction . . . 23

3.2.1 Colour . . . 23

3.2.2 Texture . . . 29

3.3 Classification . . . 32

3.3.1 Classifier . . . 32

3.3.2 Histogram Distance . . . 33

3.3.3 Spatial Information . . . 35

3.3.4 Combining Features . . . 36

3.3.5 Hierarchical Classification . . . 36

**4 Results** **41**

4.1 Black and White . . . 45

4.2 Manually Coloured Black and White . . . 47

4.3 Art . . . 49

4.4 Colour Photo . . . 51

4.5 Outdoor - Indoor . . . 52

4.6 Night - Day . . . 54

4.7 Urban - Nature . . . 56

4.8 Overall Result . . . 58

4.9 Sample Application: Query by Image . . . 60

**5 Conclusion** **62**

**Bibliography** **63**

# List of Figures

1.1 Examples of semantic annotation . . . . . 3

2.1 ImagEVAL image categories . . . . . 6

3.1 Histogram calculation . . . . . 24
3.2 RGB Histogram - green channel . . . . . 25
3.3 CIELUV / CIELAB luminance . . . . . 28
3.4 Coherent Edge Histogram . . . . . 30
3.5 Wavelet operation result . . . . . 31
3.6 Comparison Histogram distances A . . . . . 38
3.7 Comparison Histogram distances B . . . . . 39
3.8 Using image tessellation to capture spatial information . . . . . 40
3.9 Hierarchy of Classifiers . . . . . 40

4.1 Class Distribution . . . . . 41
4.2 Weka output . . . . . 55
4.3 Query by image . . . . . 61
4.4 Query by image using semantic Classes . . . . . 61

# Chapter 1

# Introduction

The automatic derivation of semantically-meaningful information from the content of an image is the focus of interest for much research on image databases. Image "semantics" can be categorized [WLW01] as

1. semantic types (e.g. landscape photograph, clip art)
2. object composition (e.g. a bike and a car parked on a beach, a sunset scene)
3. abstract semantics (e.g. people fighting, happy person, objectionable photograph)
4. detailed semantics (e.g. a detailed description of a given picture).

This thesis concerns the extraction of image semantic types, the first item in this list, from low-level image features. The extracted semantics could, for example, be used in conjunction with an automatic segmentation of images to guide the segmentation algorithm. Training and test images are provided by the ImagEVAL [Ima05] project, a French computer vision evaluation project. Image semantics to be extracted, as specified in the project description, are shown in the following list:

- Nature of the image
    1. Black and White - Colour - Manually Coloured
    2. Photograph - Painting
- Context of the image
    1. Outdoor - Indoor

2. Night - Day
3. City - Countryside

Figure 1.1 shows an example of the image annotation that is the goal in this thesis.

A variety of applications for image classification and feature extraction can be found in *Content Based Image Retrieval* (CBIR). Other fields in which image classification is widely used are: biomedicine, commerce, military, education, digital libraries, and web searching. An application especially suited to the classification under consideration here is the automatic colour correction of consumer photos during film development [LW03][SP98]. Another application could be the automatic classification of images in large electronic-form art collections, such as those maintained by museums or image archives of print media / television. Generally speaking, such a classification is useful everywhere where a manual classification or sorting process is infeasible because of the number of images under consideration.

From the classification of images, insights can be gained, as stated in [CHL03]: "The problem of separating photographs from paintings is interesting because it constitutes a first attempt at revealing the features of real-world images that are misrepresented in hand-crafted images." This leads to "digital forensics" because through classification, computer-generated images can be distinguished from hand-made art. Another point made in this paper, with regard to pornography filters, is distinguishing pornographic images from nude paintings.

The selection of applicable low-level image features with the aim of grouping images into semantically meaningful categories is a challenging task. Chapter 2 presents a summary of current research on image classification and methods used in this field. To aid the evaluation of the different methods available, a prototype for image classification is implemented. The classification of images in this work is achieved by a hierarchical classification of the semantic attributes listed above. The methods used in this prototype for feature extraction, building a statistical model and classification, are discussed in detail in Chapter 3. The results obtained are shown in Chapter 4, with the conclusion in Chapter 5.

- colour
- outdoor
- day
- nature

- colour
- outdoor
- night
- urban

Figure 1.1: Examples of semantic annotation

# Chapter 2

# Literature Review

The work in the field of automatic image classification is diverse – many different statistical methods and classification schemes are used. This chapter gives an overview of these methods. Technical details on the methods implemented for the prototype are given in the next chapter.

The first part of this chapter (section 2.1) concerns the classification by attributes for two-class problems. For evaluation of the results, a strictly defined sample of training and testing images is often used. Ambiguous images or images not belonging to either class are discarded. Therefore these results cannot be extended to a general domain straightforwardly.

For binary (i.e. two-class) classification, where the input is a test sample of images, the result obtained is either a labelling of the input images or the probabilities of either class. Evaluation and comparison of results is done on the accuracy of the methods applied to an independent test set or through the leave-one-out method.

In section 2.2, approaches towards systems for image classification, working in a wider domain, are introduced. These systems assign more than one attribute and several different usages are possible; determining input and output methods. For example, input can be an image to classify or a query for images by specifying a target class or parameters. Also possible is a search by example through the use of a query image or sketch. Again, the output of such a system may be a confidence interval of the class under consideration or a simple labelling. Other output options include a list of closest images found in a database, a semantic classification or a linguistic indexing (annotation) of the input image.

In either case, the underlying problem is finding clusters of images that are similar to

each other. This similarity is defined by human perception – equality and similarity between images in computer vision must therefore be defined so that a relation to human vision is obtained. In [SWS+00] the following rules for similarity are given:

- *Syntactic laws* govern the algorithms used e.g. object matches, spectrum similarity.
- *Human perception* is important because this defines equality on the same basis as a user experiences it, e.g. usage of the CIELAB colour space, which is closer to the human perception than the RGB space.
- *Physical laws*, e.g. physics of illumination are exploited to design features that are robust regarding lighting.
- *Geometric and topological rules*, e.g. objects appear smaller in the background.
- *Category-based rules* are special to a given narrow domain, e.g. knowledge of variation to expect in faces.
- *Culture-based rules* of equality and similarity, e.g. different appearances of faces or skin-colour.

Section 2.3 deals with feature extraction with regards to these considerations. Finally, section 2.4 gives insight into different classification methods used. More details on the features and methods selected for use in the prototype are given in Chapter 3

## 2.1 Classification by Attributes

The papers reviewed in this section propose methods for sub-problems of the specification of the ImagEVAL project, which are summarised in Figure 2.1. Each subsection deals with a different sub-problem. As no work on day-night classification appears to exist, this is omitted. A justification for the selection of semantic classes is found in [VFJZ01] where a small-scale experiment with 8 humans yielded the following high-level categories: forests and farmlands, natural scenery and mountains, beach and water scenes, pathways, sunset/sunrise, city, bridges and city scenes with water, monuments, scenes of Washington DC, a miscellaneous class of mixed city and natural scenes and a face image. However the images in question belonged to a small set of 171 vacation images, therefore this result cannot be generalised.

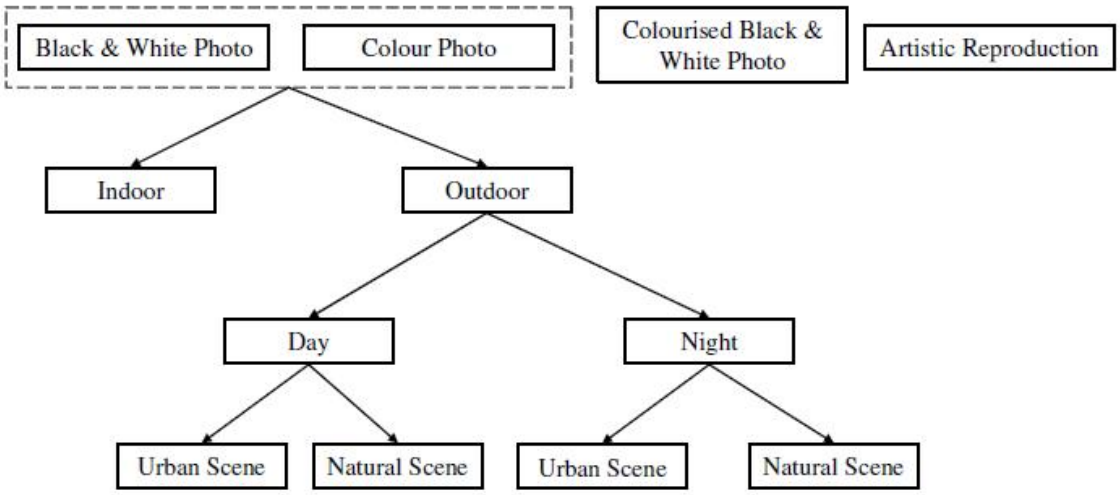



Figure 2.1: Image categories specified in the ImagEVAL project

### 2.1.1 Colour Photo - Artistic Reproduction

In [CHL03], the goal is the determination of image features distinguishing photographs of real-world scenes from (photographs of) paintings. Line (pencil or ink) drawings as well as computer-generated images are excluded but no constraint on the image content is made. Four scalar-valued visual features are defined:

1. Colour edges vs. intensity edges: removal of colour eliminates more visual information from a painting than from a photograph of a real scene.

2. Spatial variation of colour: local variation of colour - the mean of this quantity taken over all image pixels should be, on average, larger for paintings than for photographs.

3. Number of unique colours: paintings appear to have a larger colour palette than photographs.

4. Pixel saturation: paintings tend to contain a larger percentage of highly saturated pixels.

Each of the above had a hit rate of about 63%, a neural network trained on all 4 of them achieved about 71% correct classification (with a standard deviation of 4%).

In the RGBXY space proposed in this paper, an image is a represented by a point cloud in 5 dimensions. Due to the larger colour palette and larger spatial variation in paintings, photographs and paintings are expected to be well separated in this space.

The paper reports an improvement to a hit rate of 81% using the RGBXY space. A feature capturing texture, which is more repetitive in photos, is also helpful. The usage of Gabor filters yields 78% correctly classified. Through combining all these classifiers a hit-rate of 90% was achieved.

In [LF05] the aim is similar to the above: to distinguish photos from photorealistic, computer-generated pictures. An approach using the wavelet transform of the RGB channels is chosen. The features are calculated as higher order moments and error in prediction of wavelet coefficients. A linear discrimination analysis (LDA) for classification yields 99.2% correctly classified photos and 54.6% correct photorealistic images (total hit rate: 93.4%). A Support Vector Machine achieves similar results with 98.8% and 66.8% respectively (total hit rate: 94.6%). 32.000 photographic and 4.800 photorealistic images were used for training.

In [PCH+02] classification into natural picture (Photo) and business graphic (Graf) is achieved with a hit rate of 96.6%. The classifier is based on the observation that a typical natural picture has more detail, noise and smooth colour changes than a synthetic picture.

### 2.1.2 Indoor - Outdoor

In [SP98] a classification into indoor-outdoor photographs is performed. First a pre-processing step is done to achieve basic colour balancing: the images are converted to 24bit, the top and bottom 5% of intensity levels clipped and the histogram shifted to the centre and stretched.

As a baseline experiment a colour histogram for the whole image as well as for a $4 \times 4$ image tessellation (16 blocks) was calculated and the Euclidean distance used, which achieved a hit rate of 69.5%. This was enhanced by using the Ohta colour space and the histogram intersection norm. This raises the result to 74.2%. The texture features are computed using the multiresolution, simultaneous autoregressive model (MSAR) with the Mahalanobis norm. This gives 82.2% correctly classified.

As classifier a nearest neighbour algorithm gives good results although it doesn't exploit local properties. Experiments with a 3-layer neural network show that this is computationally expensive and that the results do not improve significantly.

The best result is achieved using a $4 \times 4$ image tessellation and combining the MSAR and the colour information, which yields 90.3% correctly classified. Misclassified were images of green plants, Christmas trees, green walls and close-ups. Relatively

few parameter settings are used in the classification process. Training and testing is done on 1300 consumer images provided by Kodak.

In [BOV03] the authors argue that a support vector machine with a kernel function tuned to the task (i.e. histogram intersection) is best suited to fulfil the task. As image database 600 random images from the Internet are used. Colour histograms and co-occurrence matrices are computed. The usage of a kernel without the need for explicit modelling of the association between the input images and the output labels means that this approach can be easily expanded to different problems. Using the HSV colour space, 93% accuracy is reported on the problem.

In [VFJZ01], image features calculated are the first- and second-order moments in the LUV colour space. To capture spatial colour distribution, a $10 \times 10$ sub-block tessellation was used. On a test set of 2540 images an accuracy of 88.2% is achieved. Other features tested are MSAR texture features and edge direction and coherence features. These features, as well as a combination with the colour moments did not yield a better accuracy. Misclassified images either are indoor images with sunshine through doors or windows or outdoor images with little contrast or uniform lighting (e.g. close-up shots).

### 2.1.3 Urban - Natural

In [VJZ98] the following features are evaluated for their discriminative power in urban-natural classification: colour histogram, colour coherence vector, DCT coefficient, edge direction histogram and edge direction coherence vector. The best result is achieved using a 5-Nearest Neighbour classifier on the edge direction coherence vector. The combination of features does not bring a significant improvement. The system achieves 93.9% accuracy on 2716 images using the leave-one-out method.

Misclassified images are: long distance city shots at night (difficulty in extracting edges), top view of city scenes (lack of vertical edges), highly textured buildings, buildings obstructed by trees, tree trunks and close-ups of plant stems and fences. Therefore this classifier seems to rely on strong vertical edges for discrimination between the two classes.

A different approach is followed in [IA99]. Images containing large man made objects are identified by object matching. Edges are grouped into shape representations including “L” junctions, “U” junctions and parallel groups. A channel energy model is utilized to extract lower-level feature vectors consisting of fractional energies in

various spatial channels. The image database consists of 150 monocular greyscale outdoor images taken from a ground-level camera. A hit rate of 80% is reported.

## 2.2 Classification Systems

The systems under consideration in this part are interesting with respect to the problem at hand in that they work in a wider domain, especially in that they assign more than one attribute to an image. The list of papers presented in this part is by no means exhaustive; the aim is to give an overview of possible solutions for multi-class problems. Also general findings and inherent limitations are referenced.

### 2.2.1 Hierarchical Classification

In [VFJZ01] a hierarchical classification of vacation images is achieved, using binary Bayesian classifiers. The images are classified into indoor or outdoor; outdoor images are further classified as city or landscape, finally, a subset of landscape images is classified into sunset, forest, and mountain classes.

The classifier is designed and evaluated on a database of 6931 vacation photographs. The achieved accuracy is 90.5% for indoor/outdoor, 95.3% for city/landscape, 96.6% for sunset/forest & mountain, and 96% for forest/mountain classification problems.

The paper further describes a learning method to incrementally train the classifiers as additional data become available. It also shows preliminary results for feature reduction using clustering techniques. The goal is to combine multiple two-class classifiers into a single hierarchical classifier.

### 2.2.2 Content-Based Image Retrieval

The advent of the Internet and a multitude of available digital vision sensors (digital cameras) has led to a steep increase of images available in digital archives. Indexing tools for these are required and form the justification for Content Based Image Retrieval (CBIR).

The expected result of a CBIR may vary and generally is wider then just the retrieval of images based on the presence or absence of objects. In [SWS+00] categories of application of CBIR are given as follows:

- *Search by association* is a search by iterative refinement and relevance feedback, started by a sketch or example image. This method facilitates easy browsing of large image archives, on the lookout for something interesting but not specific.
- To search for a specific image, e.g. browsing an art catalogue for an image in mind an *aimed search* can be implemented, started by an example image.
- *Category search* aims at finding an arbitrary image representative of a specific task. The query could be formulated by textual description or an input image of the same class.

The *semantic gap* is the difference between a data or language structure and the real world. This gap is introduced through the difference between a linguistic description of an image and the interpretation by a user. In speech, also ambiguous at times, the semantic gap is resolved through the context in which the information is received. For images however the context is often not known. For web based searches [DY02] shows the possibility to enhance image-content based classification with available Meta-data and surrounding text.

The *sensory gap* is the gap between an object in the real world and the information available though a description of a recording of that scene. This gap is greater in a *broad domain* e.g. all images on the Internet, than in a *narrow domain*, where variability is reduced through controlled conditions e.g. frontal views of faces under same illumination.

The common basis for CBIR systems is a signature and a comparison rule. The automatic derivation of optimal features is a challenging and important issue. Comparison between systems is difficult because human evaluation is difficult to keep consistent and therefore seriously biased [WLW01].

The low-level content of images is described by calculating features based on local colour, local texture and local shape. Features are often calculated for parts of the image, where pixel grouping can be done based on: *strong segmentation* into physical objects, currently not feasible because of the complexity in segmentation; *weak segmentation*, e.g. colour regions; *sign location* for objects with (nearly) fixed shape, e.g. an eye; or *image partitioning*, regardless of image content e.g. blocks of equal size. The image partitioning can be feasible as some normative rules, e.g. horizon in the upper half and the object of interest in centre are often followed.

Different image features are suitable for different semantic types [WLW01] (e.g.

colour indexing for outdoor but region-based for indoor) therefore it is sensible to categorise images so that semantically-adaptive searching methods can be applied in CBIR.

### 2.2.3 Linguistic Indexing

In [LW03] a statistical modelling approach to automatic *linguistic indexing* of pictures is introduced. 600 concepts with an average of 3.6 keywords per concept are trained. An example category is "male" with the keywords "man, male, people, cloth". According to the paper the advantage of linguistic indexing in CBIR is that a query image or sketch is not needed.

Training is done with a dictionary of concepts. Each concept is represented by a statistical model, the two-dimensional multiresolution hidden Markov model (see section 2.4 for details). Likelihood is used as a universal measure of similarity; no similarity distance for a particular set of features is needed. Image-features used are three colour and three texture features. Spatial information is extracted by using a Quad-tree split at 3 resolutions as well using a tessellation into $4 \times 4$ pixel blocks.

The output of the indexing scheme is an average of 6 words describing an image. To return specific and interesting information, those keywords that are most "surprising" or rare are selected. The topmost, or first, keyword returned is correct in 11.88% of all test cases. If comparing more than one returned keyword, and relaxing the "match" condition to having the true category included in first 5 returned keywords then the accuracy is 26.05%. A random scheme would yield 0.17% (1/600) accuracy. To train one concept, 30 minutes on an 800 MHz PC is needed.

### 2.2.4 Semantic Classification

In [WLW01] a CBIR system that uses *semantic classification* methods is discussed. A wavelet-based approach for feature extraction was chosen. A weak segmentation is used, enhanced by *Integrated Region Matching*. This metric is introduced to guide the clustering of the over-segmented image. Prior probabilities are consulted to determine probabilities of objects appearing together in an image (e.g. boat and water).

The overall image-to-image similarity provides a simple querying interface with a query image. A database of 200.000 general purpose images was tested.

The calculation of the signature (another term for feature vector) is done in two steps: first the image is classified based on the given categories graph-photograph and texture-nontextured. Then the signature is calculated based on features depending on the semantic type.

In the first step the image is partitioned into $4 \times 4$ pixel blocks before a statistical clustering algorithm is deployed. The segmented image is represented by a set of regions, roughly objects, characterised by colour, shape and texture. Segmentation is achieved by using a k-means algorithm. The features are 3 colour and spatial variation (wavelets in the LUV colour space). The clustering algorithm is halted when the distortion is below a threshold, the first derivative is near zero or when an upper bound is exceeded. This clustering is performed in about one second for an image of size $384 \times 256$. An image is then classified into one of $n$ manually-defined, mutually exclusive and collectively exhaustive semantic classes. In the case described $n$ is 4, the framework can be extended to more classes.

The information gained in the first step is then used to select the best features for the feature vector. Therefore the signature calculated depends on the semantic type of the image.

## 2.3 Features

> A feature has large discrimination power if the intra-class distances are small and the inter-class distances are large. [VJZ98]

The aim of feature extraction is to find representative values for each image, discriminating between the classes in question. Ideally, the obtained feature vector is of low dimension, speeding up the classification process. The features to extract can be determined by observing differences between the classes in question.

According to [WLW01], most CBIR systems use one ore more of the following categories to extract signatures from images: *"Histogram"*, *"Colour layout"* or *"Region-based"*.

Histograms are frequently used in the papers reviewed because they give a good measure of the colour-distribution found in an image and the dimensionality can be influenced through the number of bins used. A simple approach is to calculate histograms for each channel and concatenate the result to a vector.

Colour layout is a feature calculated for sub-blocks of an image. This is essentially a low resolution representation of the image (i.e. average colour of pixel blocks). Two CBIR systems which use this approach (mentioned in [WLW01]) are WBIIS, which uses Significant Daubechies wavelet coefficients and WALRUS, which exhaustively produces sub-images by sliding a window over the original image.

Region-based search represents images at object level. This requires an image segmentation where regions ideally correspond to objects. Comparison is then based on individual regions. However the automatic segmentation of an image is nearly as difficult as image understanding. The establishment of a clear mapping from a set of pictorial properties to semantics is also difficult.

Finding representative image features can also be achieved through *"salient features"* [SWS+00]. These are, for example, the points that survive longest when gradually blurring an image.

The computed features should be robust to variance between images found in the class. For example the algorithm should take different scales and rotation of the input images into account. In [WLW01] the following variations are named and tested: *"brighten, darken, blur, sharpen, saturation, pixelise, crop, shift, rotate, flip"*.

Robustness with respect to these variations is a direct result of the extraction process used. For example a histogram is robust to the translation of an object in the image, the drawback is that object location, shape and texture are discarded. Colour histograms are also sensitive to intensity variation, colour distortion and cropping. The colour layout search as well as the region-based search are sensitive to object shifting, cropping, scaling and rotation.

### 2.3.1 Colour

Colour is a product of the illuminant, surface spectral reflectance and sensor sensitivity (i.e. of digital sensors or of cones in the human eye). It can be characterized by the following parameters [Fai98, p.101]:

- Hue: visual sensation according to which the colour appears to be similar to one of the perceived colours: red, yellow, green and blue. There is no "zero" hue, therefore hue is often described as a circle. However a distinction can be made between an "achromatic" colour, without hue and a "chromatic" colour,

possessing hue.

- Brightness and lightness: brightness defines the amount of light an area appears to emit. Lightness is judged relative to a similarly illuminated area. As a side effect, as the overall luminance increases the perceived brightness follows suit while the lightness is constant.

- Colourfulness and chroma: colourfulness is the sensation according to which a colour is perceived to be more or less chromatic. Chroma is the colourfulness judged as a proportion of a similarly illuminated area that appears white. Therefore colourfulness is sensitive to changes in luminance, while chroma is approximately constant.

- Saturation: colourfulness of an area judged in proportion to its own brightness. This is similar to chroma but can be judged in isolation.

Many attempts have been made to capture perceptual colour similarity through different colour spaces. The underlying problem is that colour varies with the orientation of the surface, position and colour of the illumination as well as through different surface properties.

The most common primary colours in computing are red, green and blue (e.g. colours used in a monitor). Images are often described by three channels, in this case called the **RGB** colour space. When images are stored, or processed, in this colour space the advantage is that for the primary output device, the monitor, no conversion is needed. However for various applications different colour spaces have better properties.

The CIE[1] defined the **XYZ** primaries, which are virtual since they cannot be realized physically. Nevertheless they can be used for conversion between colour spaces. The Y primary corresponds to the luminance information, the X and Z to the chrominance. For the conversion different matrices are defined according to the illuminant used.

The CIE colour spaces have the advantage that the Euclidean distance between two colours models the human perception of difference. This has the effect that the **CIELUV** as well as the **CIELAB** colour space have good perceptional properties [LW03].

[1] CIE: International Commission on Illumination (Commission Internationale d'Eclairage)

To calculate a **greyscale** version of an image, the chroma and hue information is discarded. This is achieved by only considering the $L^*$ channel of the CIELAB or CIELUV colour space or, in the case of RGB encoded images, by an estimation calculated as the mean of the three channels.

The **Ohta** colour space [SP98] is composed by forming the 3 largest decorrelated eigenvectors of RGB. This was found by analysis of natural images and is therefore used for indoor-outdoor classification. The results in [SP98] show an increase in correctly classified images from 69.5% to 73.2% when Ohta is used instead of RGB for indoor-outdoor classification.

A similar colour space is referred to in [SWS+00], the **opponent colour** representation isolates brightness information on the third axis. This is of advantage because humans are more sensitive to brightness than to chroma.

Most methods in the literature propose the usage of colour histograms in one of these colour spaces. The number of bins used varies between 15 to 60 per channel and is found by empirical methods. In [SS94] it is stated that a 64-bin histogram has a maximum discriminating power of 25,000 images. For the prototype, described in Chapter 3, a bin size of 20 per channel was used as a compromise between dimensionality (memory usage) and discriminating power.

An alternative description of the histogram by the first three statistical moments is used in [VFJZ01], 6 features consisting of 3 means and 3 standard deviations are collected. Even simpler, the mean colour components of pixels is calculated in [LW03]. In [VJZ98] the colour histograms are calculated by a transform into the HSV colour space, followed by a clustering to 64 colours with k-means clustering. A Colour Coherence Vector proposed in [VJZ98] is an extension to histograms in that it takes into account the percentage of pixels per bin that are part of an 8-neighbour connected region of the same colour.

### 2.3.2 Texture

In [VJZ98] **edge direction histograms** are calculated by applying a Canny edge detector at 5° Intervals. The histogram is composed of 72 bins, normalised by the total number of edge points detected and an extra bin containing the count of pixels that are not part of an edge, normalized by the total number of pixels in the image. In the same paper an **edge direction coherence vector** is introduced. It discriminates structured edges from randomly distributed edges by a connected

component analysis. Edges that are not part of a larger connected component (0.1% of image size) are removed.

The detection of edges in multiple directions and at multiple scales can be performed through the use of **wavelets**, as used in [LW03]. In this paper the LUV colour space is used and the Daubechies-4 wavelet transform or the Haar transform applied. In [LF05] the first four order statistics (mean, variance, skewness, and kurtosis) of the sub band coefficient histograms at each orientation, scale, and colour channel (RGB) are collected. Also a linear predictor of coefficient magnitude is used to collect further statistics.

A texture feature that is often mentioned in literature is the multi resolution, simultaneous autoregressive model parameters (**MSAR**). Another way of capturing **frequency information** is performed in [SP98]: the 2D DFT magnitude is calculated, followed by the 2D DCT. In [VFJZ01] and [VJZ98] the DCT Coefficients are calculated using the JPEG image compression algorithm and its central moments of second and third order.

**Gabor wavelets** or Gabor filters can also be used to extract texture information at certain wavelength and orientation. Gabor filters are used in [MM96] and [Wag99].

In [Wag99] an overview of feature sets for texture analysis is given and benchmarks calculated. Table 2.1 shows the names of the feature, the size of the feature vectors, the theoretical principle and the result when applied on the Brodatz dataset, a widespread benchmark dataset with 13 classes. The Gabor and wavelet approach are seen to perform extremely well on this dataset.

### 2.3.3 Region Matching

This method aims at finding a match for an object in an image. The difficulty with region matching lies in the problem of finding representative digital descriptions of real-world objects (semantic gap). The drawback is that object rotation and different illumination have a strong effect on the result. Normally objects are easier to find if they do not have strong variance in appearance. For example in [RBK98], [GL00] and [RBK96] a neural network-based face detection performs region matching. The network examines small windows of an image and decides whether each window contains a face.

The aim in [IA99] is discriminating urban from nature images. Structures typical of man made objects are found by region matching. The templates used are: “L”

| Feature name | size | Principle | |
|---|---|---|---|
| Haralick | 14 | grey-level co-occurrence matrix | 82.9% |
| Unser | 32 | sum and difference histograms | 92.6% |
| Galloway | 20 | grey-level run lengths | 84.7% |
| Laine | 21 | wavelet packet signatures | 92.4% |
| Local | 14 | direct functions of grey values | 61.1% |
| Fractal (1) | 10 | fractal box dimension | 62.6% |
| Fractal (2) | 47 | fractal dimension from blankets | 66.5% |
| Laws | 14 | Laws' convolution matrices | 89.7% |
| Fourier coeff. | 33 | energy in ring-shaped regions of Fourier space | 92.7% |
| Chen | 16 | geometric properties from binary image planes | 93.1% |
| Sun et al. | 5 | modified Haralick approach | 63.9% |
| Pikaz et al. | 31 | pyramid decomposition | 79.4% |
| Gabor | 12 | Gabor wavelets | 92.2% |
| Markov | 7 | Markov random fields | 83.1% |
| Dapeng | 13 | grey-level difference co-occurrence | 85.8% |
| Amadasun | 5 | neighbouring difference histogram | 83.4% |
| Mao et al. | 6 | autoregressive models | 86.3% |
| Amelung | 18 | histogram and gradient features | 93.0% |

Table 2.1: Feature sets for texture classification (taken from [Wag99])

.

junctions, "U" junctions and parallel groups. These "ordered" edges occur more often in urban than in natural images.

### 2.3.4 Spatial Information

In [SP98], to capture spatial information a $4 \times 4$ image tessellation is used. Training is done on the 16 sub-blocks and a simple majority classifier used to combine the results. A neural net as well as a mixture of experts classifier proved to be slower than this method and the additional computation did not bring a significant improvement of results. However, additional information is gained by looking at the output of the neural net to see which sub-blocks are important for the classification.

Also in [SP98], multiple feature combination is introduced and different methods explained. Concatenation of feature vectors increases the dimensionality of the problem, also it is difficult to construct a metric permitting comparison between features. This problem is solved by concatenating the results of the independent sub-block classification. Again, a majority classifier is used for the final labelling of the images.

Various variations of this method are used, for example in [VJZ98] 5 local histograms are computed for the areas top-bottom, right-left and centre. In [VFJZ01] an image is divided into 100 ($10 \times 10$) sub-blocks and LUV histograms computed.

A combination of local and global features is extracted in [LW03]. This is achieved by using a quad-tree split at three resolutions.

In [QFW04] orientational colour correlograms are used. This is an extension of the grey-level co-occurrence matrix into the colour space. This method is then evaluated against colour histogram and colour correlogram and found to perform better.

Features that capture information at multiple scales, e.g. Wavelet and Gabor filters, also capture spatial information. All methods so far capture spatial information by calculation of a feature vector suited to this task. A different approach is to perform the extraction of spatial information in the classifier, an example for this, the Multiresolution Hidden Markov Model, is given in the next section.

## 2.4 Statistical Models

Only the most common classification methods are described in this section. Other approaches include fuzzy image classification [AK97] and neural networks [RBK98] [GL00] [RBK96].

### 2.4.1 Bayes Decision Rule

With only the *prior probabilities* $P$ of observing each class of a two class problem available, the following decision rule (for classes $\omega_1$ and $\omega_2$) can be formulated and is often referenced as "baseline" [DHS00]:

$$\text{Resultant Class} = \begin{cases} \omega_1 & \text{if } P(\omega_1) > P(\omega_2) \\ \omega_2 & \text{otherwise} \end{cases} \tag{2.1}$$

However normally measurements (features) that take on different values for each class are selected. For a given classification problem the observed values for each class are taken into account by calculating the *class-conditional probability density* function. This function shows the probability of measuring a particular feature value given the input class is $\omega_i$.

When the prior probabilities and conditional densities are known the probability of an observation $x$ belonging to class $\omega_i$ is formulated through the *Bayes formula*:

$$P(\omega_i|x) = \frac{p(x|\omega_i)P(\omega_i)}{p(x)} \tag{2.2}$$

or in words [IA99]: the a posteriori probability of class $\omega_i$ with given feature vector x is

$$\frac{\text{class conditional probability density function of x} \times \text{a priori probability of class } \omega_i}{\text{probability density function of observing x}}$$

The *Bayes decision rule* simply states that the classification is completed by choosing the class with the larger a posteriori probability.

In [VFJZ01] a small vector quantizer is used to model the class-conditional densities of the features, required by the Bayesian methodology. The authors see the advantage of the Bayesian approach in the small number of codebook vectors representing each class, reducing the number of comparisons necessary for each classification and that it allows for the integration of multiple features through the class-conditional densities. In addition the degrees of confidence may be used to incorporate a reject option [LSS05].

### 2.4.2 *k*-Nearest Neighbour

This algorithm makes the classification by examining the labels on the $k$ nearest neighbours of the feature vector to be classified in feature space and taking a vote [DHS00] – this selects the class of which more neighbours are counted. The selected $k$ should be odd to reduce the chance of ties. Underlying this method is an estimate of the a posteriori class probability. In particular: when $k$ approaches infinity, the estimated probabilities match the true probabilities and the error rate is the same as the Bayes rate. If $d_i$ is the [0,1] normalised Euclidean distance between the feature-vector under consideration and its $i^{th}$ nearest neighbour with true class $c_i$ $(i = 1, .., k)$ then the confidence $p_j$ that the test image belongs to class $j$ is calculated as [VJZ98]:

$$p_j = \frac{\sum_{i:c_i=j}(1 - d_i)}{\sum_{i \leq k}(1 - d_i)}, j = 1, 2 \tag{2.3}$$

If $p_j > 0.5$, the test image is assigned to class $j$.

### 2.4.3 Decision Trees

Decision trees [DHS00, p.395] are often used for nominal data, as opposed to real-valued data used in conjunction with the above classifiers. A series of questions and answers is modelled through a tree its nodes and the branches to other nodes. When "growing" the tree, each node corresponds to a sample of data. Beginning at the root, with all samples, a property is found to split the samples into two sub-samples (in the case of binary trees). The method for selection of this property is crucial for the quality of the tree. There are several measures available for measuring impurity in a sample. The split is done so that this value is as small as possible for the descending branches. The splitting is stopped at leaves that are pure, i.e. contain samples of a single category, or when a criterion for stopping is met (for example a sufficiently small remaining sample or impurity). In Weka [WF05] the tree "J48", a C4 tree, can be parametrised to stop on a given minimum number of instances per leaf and through the use of a confidence factor. This factor is used to test the hypothesis that a given split is different from a random split. Another option is pruning, a simplification step performed after the creation of the tree.

### 2.4.4 Multiresolution Hidden Markov Model

In [LW03] a two-dimensional multiresolution hidden Markov model is described. It summarizes clusters of feature vectors at multiple resolutions and the spatial relation between clusters at different scales. The parameters can be estimated by the maximum likelihood criterion using the expectation maximisation (EM) algorithm.

In [LGO00] a multiresolution hidden Markov model for classifying images is used. Each image is represented by feature vectors at several resolutions, which are statistically dependent as modelled by the underlying state process, a multiscale Markov mesh. Unknowns in the model are estimated by maximum likelihood, in particular by employing the expectation-maximization algorithm. An image is classified by finding the optimal set of states with maximum a posteriori probability. States are then mapped into classes. The multiresolution model enables multiscale information about context to be incorporated into classification. Sub-optimal algorithms based on the model provide progressive classification that is much faster than the algorithm based on single-resolution hidden Markov models.

### 2.4.5 Dimensionality Reduction

The high dimensionality of some feature vectors makes them impractical for building a classifier. Where the reduction of dimension cannot be carried out through design of the low-level features, reducing the dimension can be achieved through the algorithms mentioned here.

The most often mentioned method is the Principal Component Analysis (PCA) [DHS00, p.568]. This is a rotation of the data to an N-dimensional linear subspace, determined by calculating eigenvalues and eigenvectors and sorting these. Similar to this method is the Karhunen-Loeve mapping (KLM). In the implementation of PRTools [DJP+04] the difference is that the KLM is a PCA of the mean covariance matrix while PCA is computed on the overall covariance matrix.

In [QFW04] a boosting classification scheme is described that selects the most discriminative features automatically. This method is reported to work better than PCA.

Fisher's linear discrimination (FLD) criterion, described in [GL00] optimises the feature set by taking the class labels into account to maximise inter- and intra-class scatter.

# Chapter 3

# A System for Image Attribute Classification

Many algorithms used in the literature are presented in the previous chapter. This chapter gives details on the methods implemented in the prototype. The features selected are histograms in several colour spaces and features capturing texture information (Gabor and wavelet filters and coherent edge histograms). For classification the $k$-NN algorithm is used. Section 3.1 lists the software used. Section 3.2 describes how the features are extracted from the images. Finally, details on the classification scheme employed are given in section 3.3.

## 3.1 Software

For the prototype the following software was used:

- Matlab Version 6.5
- PRTools: Pattern Recognition Tools[1] Version 4.0.14 04-Mar-2005 [DJP+04]
- Weka: Waikato Environment for Knowledge Analysis[2] Version 3.4.4 [WF05]

An export function is used to convert the feature vectors and class labels to the format expected by Weka. This tool was used to experiment with different classifiers, dimensionality reduction algorithms and parameter settings. The findings were then

[1] available at http://www.prtools.org/
[2] available at http://www.cs.waikato.ac.nz/ml/weka/

incorporated into the design of the prototype, programmed in Matlab and making use of the PRTools functions. Detailed information on the tests carried out with Weka are not explored, rather the resulting parameter settings for classification with the prototype are explained.

## 3.2 Feature Extraction

Several features are implemented and evaluated for each sub-task. The framework of the prototype allows for a rapid testing of different combinations and parameter settings. However the calculation of the features is quite time consuming, roughly an hour per feature for the 5474 images. For the 7 features selected for the prototype, the calculation time per image is about 3.9 seconds. Once the parameter settings for the feature calculation are known this task has to be performed only once for all images in the database.

The only pre-processing step is the resizing of the images so that the smaller side has a length of 256 pixels, for the images used the other side then has a length of about 400 pixels. Both landscape and portrait images can be used. Few black and white images are encoded with a single channel, these are modified to have three identical channels.

### 3.2.1 Colour

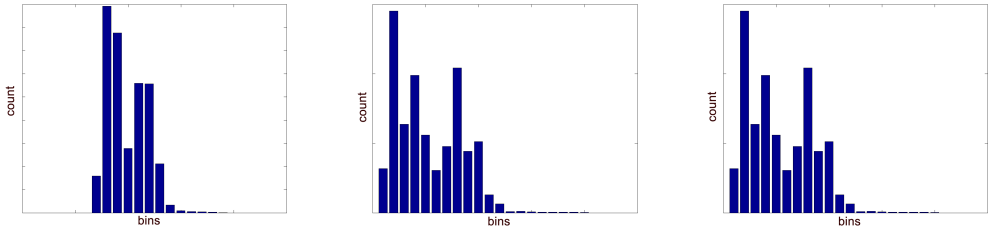
All input images are encoded in the RGB colour space. Therefore it would be of advantage to work with RGB since no conversion is needed. The drawback however is that this space is ill-suited for most classification based on colour. For example different illumination will change the perceived colour. While the human eye will make adjustments to accommodate for this, it is hard to construct a metric for which an image has the same (pixel) values regardless of lighting conditions. The human eye is also able to distinguish between more different levels of green then red or blue; therefore perceptually similar images and any feature constructed thereof should possibly consider the green channel more than the other two. The luminance information is more important to our perception than the chroma, a difficult fact to consider when using a colour-space where luminance is not directly available, rather being a combination of all three channels.

To capture colour information, histograms are calculated in several colour spaces.

This section shows why the particular conversions were considered and details on the parameters chosen.

For the calculation of a histogram two parameters have to be chosen: the number of bins to use and the interval to be represented by each bin. As mentioned in Section 2.3.1, the number of bins has a direct influence on the maximum discriminating power of the feature. However, a large number of bins has the disadvantage that the classifier has to deal with a large dimensionality. For an insufficient number of training images a large dimensionality can also lead to overfitting – if each image has a distinct distribution of colour values and therefore a unique feature vector, no generalisation is performed. We have therefore opted to use only 20 bins per channel.

The second parameter – the interval to be represented by each bin is calculated by a division by the bin size of the interval between minimum and maximum value to be expected in the colour space. For RGB for example the channels encode values between 0 and 255. A different approach is to calculate the intervals based on the minimum and maximum value for each image. The drawback of this approach is that a direct comparison between histograms is hampered since the bins represent different values. However this operation is analogous to a pre-processing step proposed in [SP98], an illumination compensation whereby the histogram of the image is shifted to the centre and stretched to accommodate all possible levels. Figure 3.1 shows that this basic colour balancing operation has a similar effect to choosing the intervals for the bins on a per image basis.




(a) fixed (b) per image (c) fixed preprocessed

Figure 3.1: Histogram calculation – determination of histogram intervals for greyscale image: (a) fixed intervals of bins; (b) calculation of intervals based on image; (c) fixed intervals applied to pre-processed image.

For the conversion to the CIELAB and CIELUV spaces, as indicated in the last chapter a conversion to the CIE XYZ primaries is needed beforehand. This is accomplished by a matrix multiplication of each RGB vector, for example RGB to

CIE XYZ with the C illuminant:

$$\begin{bmatrix} 0.607 & 0.174 & 0.200 \\ 0.299 & 0.587 & 0.114 \\ 0.000 & 0.066 & 1.116 \end{bmatrix}$$

The HSV colour space, representing hue, saturation and colour value (brightness) has the shape of a hexagonal cone. The angle is given by the hue, the distance from the centre of the cone by the saturation and the vertical position by the value. This colour space is used for the colour statistics.

**RGB Histogram** Although the RGB space was expected to perform worse than other colour spaces for the reasons mentioned above, there are good reasons for calculating a feature vector based on this space. An advantage is that no conversion errors are introduced. The classification of images into the nature and urban class was also expected to benefit from this space when considering the green channel which is expected to show higher values for the nature class. The means of the green channel for 1000 images are shown in Figure 3.2. The distributions follow a distinctly different pattern justifying the inclusion of this feature into the evaluation process.

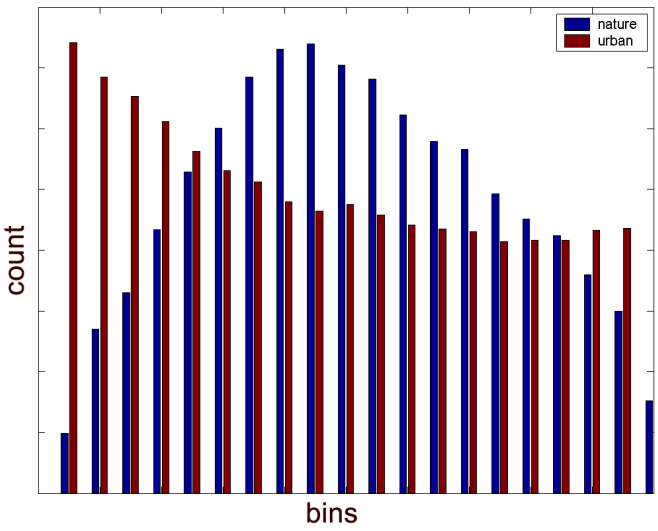



Figure 3.2: RGB Histogram: mean of green channel for classes nature and urban

**Ohta Histogram** The Ohta colour space is proposed for indoor-outdoor classification in [SP98], and is calculated as follows:

$$\begin{aligned} I1 &= R+G+B \\ I2 &= R-B \\ I3 &= R-2G+B \end{aligned} \tag{3.1}$$

The first channel of this space captures brightness information as it is the sum of the three channels of RGB. Therefore the interval for the histogram is fixed to a minimum of 0 and a maximum of $3 \times 255$. The interval of expected values for the second channel is $[-255, 255]$ and for the third channel $[-255 \times 2, 255 \times 2]$. Results with this colour space discouraged its further use. Although it is found to be suited for indoor-outdoor classification it is outperformed by the CIELAB space, also used for other sub-problems.

**CIELUV Histogram** As mentioned in the last chapter the CIELUV colour space has good perceptional properties and its use has been proposed in several papers on image classification. In the formula below $L^*$ is the luminance $u^*$ and $v^*$ encode chrominance. A reference white $X_nY_nZ_n$ is needed for the conversion.

$$L^* = \begin{cases} 116\sqrt[3]{\frac{Y}{Y_n}} & \text{if } \frac{Y}{Y_n} > 0.008856 \\ 903.3\frac{Y}{Y_n} & \text{otherwise} \end{cases} \tag{3.2}$$

$$\begin{aligned} u^* &= 13(L*)(u' - u'_n) \\ v^* &= 13(L*)(v' - v'_n) \end{aligned}$$

where

$$\begin{aligned} u' &= \frac{4X}{X+15Y+3Z} \\ v' &= \frac{9*Y}{X+15Y+3Z} \\ u'_n &= \frac{4X_n}{X_n+15Y_n+3Z_n} \\ v'_n &= \frac{9Y_n}{X_n+15Y_n+3Z_n} \end{aligned}$$

**CIELAB Histogram** For conversion to the CIELAB (also written as $L^*a^*b^*$) colour space a reference white $X_nY_nZ_n$ is needed. $L^*$ is the perceived whiteness, $a^*$

the red-green chrominance and $b^*$ the yellow-blue chrominance.

$$\begin{aligned} L^* &= 116 f\left(\frac{Y}{Y_n}\right) - 16 \\ a^* &= 500\left[f\left(\frac{X}{X_n}\right) - f\left(\frac{Y}{Y_n}\right)\right] \\ b^* &= 200\left[f\left(\frac{Y}{Y_n}\right) - f\left(\frac{Z}{Z_n}\right)\right] \end{aligned} \tag{3.3}$$

$$f(\omega) = \begin{cases} \omega^{1/3} & \text{if } \omega > 0.008856 \\ 7.787\omega + 16/116 & \text{oherwise} \end{cases}$$

An advantage of the CIELAB as well as the CIELUV colour space is that the Euclidean distance between two colours models the human perception of colour difference. The luminance information is directly available in the first channel. The calculation of the luminance is more complex than in the case of the Ohta colour space and is expected to lead to better results. Figure 3.3 shows the means of the first channel ($L$) for 200 images for the nature-urban classes. The CIELAB calculation of the luminance seems to be better suited for discriminating between these classes.

**Srgb Histogram** The calculation of the normalized RGB colour space[3] is performed as proposed in [CHL03]. The "intensity free" image is computed by division by the intensity at each pixel. The calculation of the intensities is based on the Matlab function *rgb2gray* which calculates $I$ as follows:

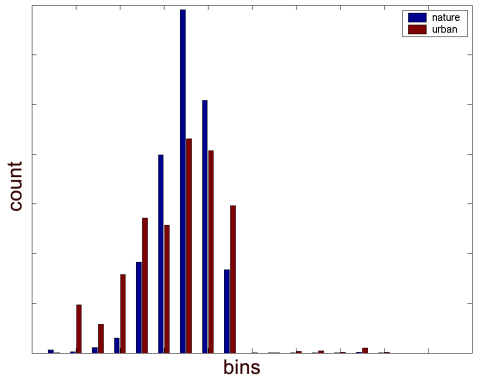
$$I = (299 * R + 587 * G + 114 * B)/1000 \tag{3.4}$$

The normalised RGB values are then:

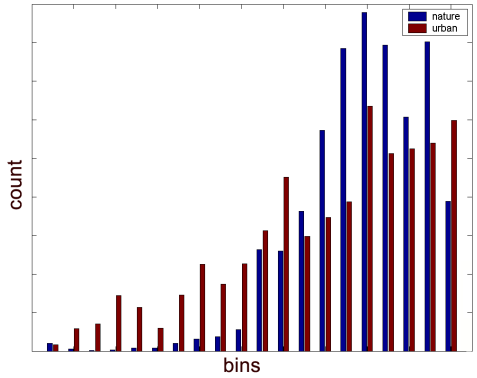
$$\begin{aligned} R^* &= R/I \\ G^* &= G/I \\ B^* &= B/I \end{aligned} \tag{3.5}$$

**Colour Statistics** Apart from histograms the following statistics are collected:

[3]This is not the sRGB as defined by IEC 61966-2-1 "Default RGB Colour Space".




(a) CIELUV luminance




(b) CIELAB luminance

Figure 3.3: means of luminance channel for classes nature and urban

- Illuminant: this value indicates the colour of the light source. Illuminance refers to the amount of incident light and can be estimated by the amount of light reflected from a surface (luminance). It is calculated in two versions, through the "Grey-world algorithm" and the "White patch algorithm". The former is calculated by the mean of the three colour channels, which is assumed to be "grey" (multiplied by 2 to get white), the latter is calculated by assuming that a white patch is always visible in an image, therefore taking the maximum value of each channel.

- Unique colours: this value is calculated by transformation into the HSV-space and counting the unique values in the Hue channel.

- Histogram sparseness: a histogram is calculated and bins containing counts higher than a fixed cut-off value counted.

- Pixel saturation: this is calculated as a ratio between the number of highly

saturated and unsaturated pixels in the HSV colour space [CHL03].

- Variance in each channel of the RGB space.
- Variance between the three channels of the RGB space.

### 3.2.2 Texture

The following texture features are implemented for usage in the prototype and evaluated for their discriminative power on the classification problems:

**Edge direction** This feature is used to compare the frequency of occurrence of edge directions. As with colour, a histogram is used to discretise the values.

There are a variety of edge detectors available in literature. To find the edge directions two convolution kernels are applied to the image to find horizontal and vertical edges. For example with the Prewitt operator:

$$h = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}, \quad v = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

For a greyscale image the gradient is calculated in the two directions. The next step is the calculation of the magnitude and direction at each pixel $x$:

$$m(x) = \sqrt{f_h(x)^2 + f_v(x)^2} \tag{3.6}$$

$$\theta(x) = arctan\left(\frac{f_v(x)}{f_h(x)}\right) \tag{3.7}$$

where $f_h$ and $f_v$ are the horizontal and vertical edges found by Matlab's *gradient* function.

The calculation of the **edge direction coherence vector** is accomplished by a morphological closing of the magnitude image with a line segment followed by a morphological opening with a small disk. Thereby the dominating structures are enforced while degenerate "edges" – isolated pixels – are removed. As above, a greyscale image is used for the input. In both cases the result is a histogram of the direction image multiplied (masked) by the thresholded magnitude image. The threshold is determined using the Matlab function *graythresh*. The 37 bins represent 5 degree intervals from $-90$ to 90 degrees. The number of edge pixels found is stored in an extra bin of the histogram. Normalization with the image size is also

performed. A result of taking the means of the histograms of 200 images can be seen in Figure 3.4. As can be observed, the histograms are distinct for the nature and urban classes.

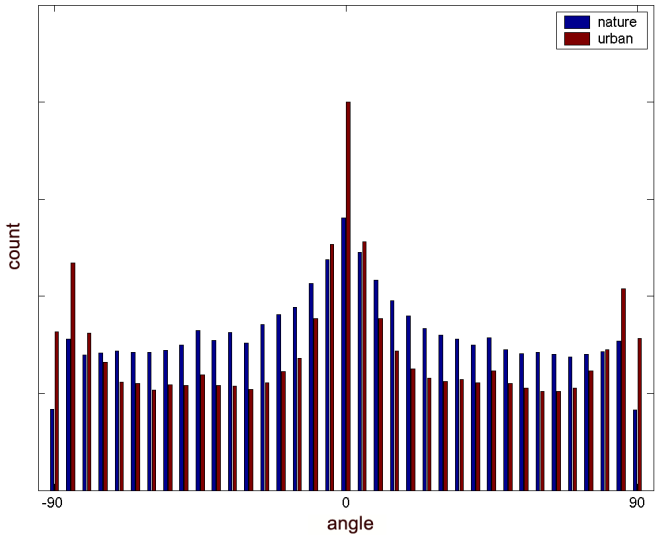



Figure 3.4: Coherent Edge Histogram for classes nature and urban

**Edge statistics** This feature is used to determine whether the edges in the image result from intensity changes, as is the case with natural images, or from changes in hue, a method employed in paintings [CHL03]. The intensity edges are found as above. The colour edges are found by first transforming the image into the sRGB space, resulting in normalised RGB components. The colour edges of the resulting "intensity-free" image are then determined by applying the edge detector to the three colour channels and fusing the results by taking the maximum. The feature extracted is the fraction of pure intensity-edge pixels.

**Wavelets** The Haar transform is used to decompose an image into frequency bands. This decomposition is generally used for compressing images. The Haar transform is defined as:

$$T = \frac{1}{\sqrt{2}} * \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \tag{3.8}$$

The decomposition is performed by using the formula $Y = T * X * T^T$. Applied to a $2 \times 2$ pixel block with values $[ab; cd]$ this yields:

$$T * \begin{pmatrix} a & b \\ c & d \end{pmatrix} * T^T = \frac{1}{2} * \begin{pmatrix} a+b+c+d & a-b+c-d \\ a+b-c-d & a-b-c+d \end{pmatrix} \tag{3.9}$$

To apply this to an image, the image is divided into $2 \times 2$ pixel blocks and the formula applied as above. The result is equivalent to filtering with the following functions:

- Top left: 4-point average or 2-D lowpass (Lo-Lo) filter.
- Top right: Average horizontal gradient or horizontal highpass and vertical lowpass (Hi-Lo) filter.
- Lower left: Average vertical gradient or horizontal lowpass and vertical highpass (Lo-Hi) filter.
- Lower right: Diagonal curvature or 2-D highpass (Hi-Hi) filter.

(a) input image (b) output image

Figure 3.5: Reordered result of the Wavelet filter: average, horizontal, vertical and diagonal information.

The result can be reordered (by putting all top-left, top-right etc. results together) to produce the output as shown in Figure 3.5. To extract an image feature this transform is applied to the $L$ component of a LUV image [LW03] (an option would be to apply it to the channels of RGB). The square root of the second order moment of wavelet coefficients in the three high-frequency bands is computed. This image feature captures variations in different directions.

When the transform is applied recursively to the top left quarter of the result, representing detail not captured by the gradient bands, effectively a multi scale feature is constructed. While the first step considers 4 pixels at a time, at the last level each of the 4 inputs is a quarter of the image. In the implementation of the prototype 4 levels are computed. This yields a feature vector of length 12.

The **Gabor filter** is a quadrature filter. It selects a certain wavelength range (bandwidth) around the centre wavelength using the Gaussian function. This is similar to using the windowed Fourier transform with a Gaussian window function. The feature vector is constructed by calculating the mean and standard deviation of the magnitude of the transform coefficients at several scales and orientations ([MM96] [Wag99]). This means that the fast Fourier transform (FFT) is applied to an image and then the Gabor filter, specific to this scale and orientation, is applied. Now the inverse of the FFT is taken and the mean and standard deviation calculated. For the prototype this filter is applied at 6 orientations and at 4 scales. Two values are collected at each point; therefore the feature vector has the length 48.

## 3.3 Classification

In this section various facets of the classification system of the prototype are explained.In section 3.3.1 the used classifier and its parameter as well as the employed dimensionality reduction is referenced. In section 3.3.2 algorithms for the calculation of feature vector distances are compared. Section 3.3.3 shows how spatial information is captured and section 3.3.4 how multiple features are combined. Finally, section 3.3.5 shows how the classification of the sub-problems is combined in one model.

### 3.3.1 Classifier

The framework provided by PRTools was used extensively for building a classifier. The feature vectors and target classes are stored in a “dataset”, a structure which also provides fields for prior probabilities. The data is split into a training and a test dataset. This aids the evaluation of the classifier at build time. A PRTools classifier can be initialised without data, only the parameters are set. Before the data is applied to the classifier a feature reduction is performed through the Prin-

cipal component analysis (PCA). Other algorithms for feature reduction that were tested are the Karhunen-Loeve Mapping (KLM) and Fisher's Least Square Linear Classifier.

The classifier used can be selected through a parameter, the following were tested:

- Back-propagation trained feed-forward neural net classifier
- Parzen classifier
- Linear Bayes Normal Classifier
- Quadratic Bayes Normal Classifier
- Mixture of Gaussian classifier
- Decision tree classifier
- Bootstrapping and aggregation of classifiers (Bagging)
- $k$-Nearest Neighbour Classifier ($k$-NN).

The results reported in Chapter 4 were obtained with the $k$-NN classifier, where the number of neighbours is set to 5. The other classifiers are deselected due to their complexity, sharply increasing computation time (neural net, Mixture of Gaussian), or because of their lower performance, probably because of the inability to model complex distributions (Linear and Quadratic Bayes and Parzen classifier). The Bagging classifier, based on $k$-NN and the Decision trees proved to be competitive but not as robust as the $k$-NN classifier.

The prototype-function for building a classifier returns a structure, representing the combination of feature reduction map and classifier, which is stored to disk for later use.

### 3.3.2 Histogram Distance

A histogram can be conveniently used as feature vector, where each bin accounts for a dimension of the feature vector. Similarity, or rather dissimilarity $s$, between two feature vectors $F$ is often calculated as a function of the **Euclidean distance** (also called L2 distance):

$$s(F^q, F^d) = g\left[d_{Euc}(F^q, F^d)\right] \tag{3.10}$$

$$d_{Euc}(F^q, F^d) = \sqrt{(F^q - F^d)^T(F^q - F^d)} \tag{3.11}$$

where $g$ is a positive, monotonically non-increasing function and $d$ is the distance function. Generally a problem with histograms is that classifiers expect decorrelated variables whilst in a histogram the values of the bins are highly correlated. In an attempt to overcome this and for "incorporating the metric of the feature space into the similarity measure" [SWS+00] the Mahalanobis distances take into account the similarity between related bins. The correlation of the data set is incorporated into the **Mahalanobis distance** metric by multiplication with the covariance matrix $\Sigma$.

$$d_{Maha}(F^q, F^d) = \sqrt{(F^q - F^d)^T\Sigma^{-1}(F^q - F^d)} \tag{3.12}$$

Another option to make the histogram distance robust to small variations is the usage of **cumulative histograms**. For the calculation of these the Matlab function *cumsum* is used. Although the information stored in a cumulative histogram is identical to that stored in a normal histogram, the comparison of a single bin has a different effect. For example: when comparing bin 10 (i.e. for a 20 bin histogram, in RGB, red channel, median of possible values) in a normal histogram the comparison shows which of the two images has a higher percentage of pixels with the colours specified by that bin. In a cumulative histogram, the comparison of the two bins shows how many pixels of the colour specified by this bin as well as all preceding bins are contained in each image. Therefore, when comparing only a subset of the available bins, the cumulative histogram might be of advantage. **Histogram smoothing** has a similar effect, single "spikes" are spread out and influence other bins.

The distance between histograms can also be measured with the **intersection distance**, which measures the amount of overlap between corresponding buckets of two histograms. It is less sensitive to outliers because the linear error instead of the squared error is penalised [SP98].

$$d_{\cap}(F^q, F^d) = \sum_{i=1}^{n} F_i^q - min(F_i^q, F_i^d) \tag{3.13}$$

$$\tag{3.14}$$

where $n$ is the number of bins.

In [SWS+00] the observation is made that a slightly modified version has the same ordinal properties as the **Manhattan distance** (also called L1 distance) when all images have the same number of pixels (same $\sum_i F_i$ for all $i$). The formula is given as:

$$d_{\cap}(F^q, F^d) = \sum_{i=1}^{n} min(F_i^q, F_i^d) \tag{3.15}$$

In [RTG98] the **Earth Mover's distance** is described. The underlying notion is an application of the transportation problem, well covered in the literature, to histograms. As a way of comparison in Figure 3.6 the intra- and interclass distances for the two classes "day" and "night" are shown in a histogram. Only the green channel (of the RGB histograms) and only the top - left image block was considered for simplicity. It can be observed that the distance metric does influence class distances considerably. However the somewhat more complex classes "nature" and "urban" show much less deviation for the three methods under consideration, see Figure 3.7.

These distance metrics are mostly considered when dealing with colour histograms, but can be extended to texture histograms where applicable. However, these variations of distance calculation did not bring a big improvement of classification results. Therefore, for the prototype the Euclidean distance was used after all.

### 3.3.3 Spatial Information

To capture spatial information, each image is divided into 16 sub-images. This $4 \times 4$ image tessellation is of benefit because image regions can be weighted according to their importance. For each sub-block a feature vector is calculated separately. A simple concatenation of these would increase the dimensionality by a factor of 16, to keep the classification simpler the following method is used: a classifier is built for each sub-block and a combining classifier, described in the next section, effectively weights the results of these.

A drawback of this approach is that only simple concepts can be captured through this method (e.g. blue sky at the top - for outdoor images). Complex concepts, such as XOR cannot be solved. As an example for successful weighting, Figure 3.8 shows the error rate for indoor-outdoor classification based on the RGB histogram,

averaged over the sub-blocks of 1000 test images when trained with 2000 images. In Figure 3.8, white represents the best error rate of 0.244% and black the worst with 0.365%. As can be observed the classification is better for the blocks in the upper part of the images, probably capturing the "sky" information. Also the combination of the results of the individual sub-blocks brings an improvement to an overall error rate of 0.183%.

### 3.3.4 Combining Features

The method used for incorporating spatial information is extended for several features straightforwardly. For each sub-block and for each feature a classifier is trained using a subset of 70% of the data available. Depending on the number of features used, between 16, for one feature, and 64, for 4 features, classifiers have to be trained.

These classifiers are applied to 100% of the training data independently. The output when applying a classifier is a value signifying the confidence with which each image belongs to the class under consideration. In the next step these values are concatenated to a feature vector and the combining classifier trained.

The training of the sub-blocks with 70% of the data is done to introduce "unseen" data for the combining classifier. This avoids overfitting the combining classifier. The number of classifiers for each sub-problem is the number of blocks times the number of features plus one.

Experiments were also carried out with the possibilities for combining classifiers provided by PRTools. These are: Product, Mean, Median, Maximum, Minimum and Voting combiner. However classification with these combiners generally shows an error rate higher than that achieved with the scheme above.

### 3.3.5 Hierarchical Classification

A hierarchical classification similar to that described in [VFJZ01] is implemented. The classifier for the whole problem is organised in the hierarchy shown in Figure 3.9.

At each node the training or application of a classifier takes place. Only the appropriate sub-sample of images, as determined by the node, is passed to the children nodes. At leaf nodes training or classification stops. This is a divide and-conquer strategy with several advantages. One advantage, compared to a classification of all attributes at once, is reduced complexity through reduction to two-class problems.

Also there is no need for a third class of images belonging to none of the classes under consideration.

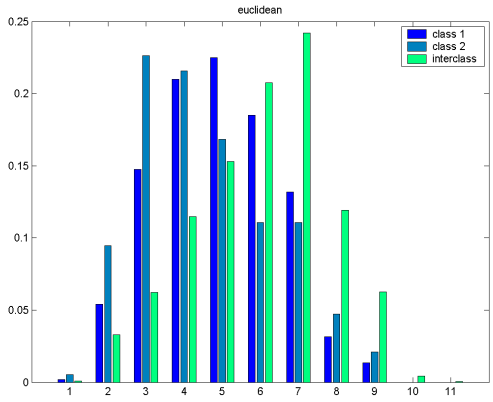
Each node can be configured individually. The prototype currently has settings for: enabling/ disabling classification, list of low-level features selected, prior probabilities, chosen combining scheme (classifier, voting scheme) and the list of children, if any. This structure could be extended for parameters specifying the type of classifier ($k$-NN, decision trees etc.) and parameters to use. During the training phase the obtained classifiers are also stored in this structure.

This scheme also helps to keep the feature-vector used for training and during classification as small as possible, for example for day-night classification only one feature is used.

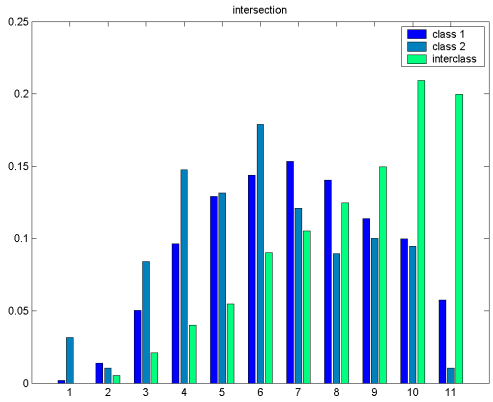
The logic of the problem-domain is easy to implement through the setting of the "children" list. This allows for a relatively easy extension to other attributes. Through this integration of the logic, inherent in the targets, a plausibility-check is not needed for the class labels (e.g. a setting of two contradicting labels does not lead to an error).

When applying the classifier, classification stops at the leaf nodes. This leads to an increase of speed and could be further exploited to only extract the needed features for each image.

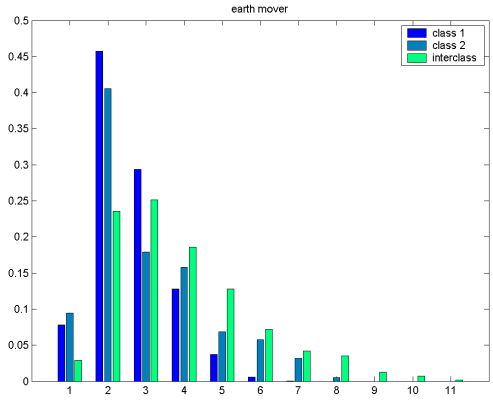
Each of the nodes can be analysed separately. Figures such as the one shown in Figure 3.8 are available for each attribute and feature pair and help to interpret performance at each node. During training an estimate of the error expected for each feature and node is output.




(a) Euclidean

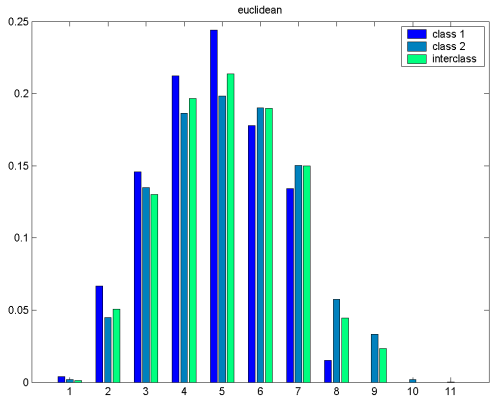



(b) Intersection

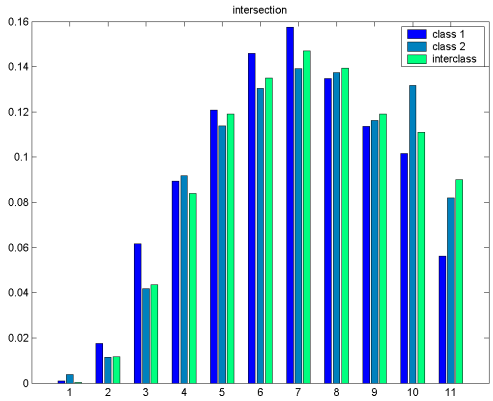



(c) Earth Mover's

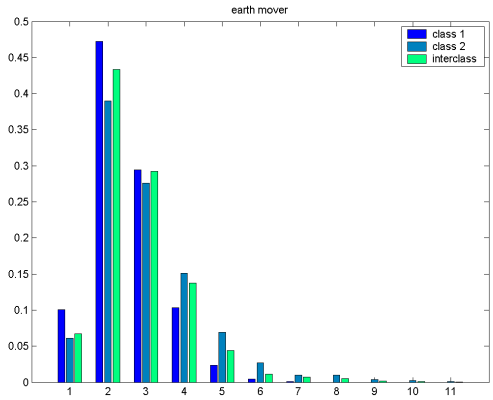
Figure 3.6: Comparison of Histogram distances for Day-Night Classes. Per diagram two intra class and the inter class distance is shown




(a) Euclidean




(b) Intersection




(c) Earth Mover's

Figure 3.7: Comparison of Histogram distances for Nature-Urban Classes. Per diagram two intra class and the inter class distance is shown

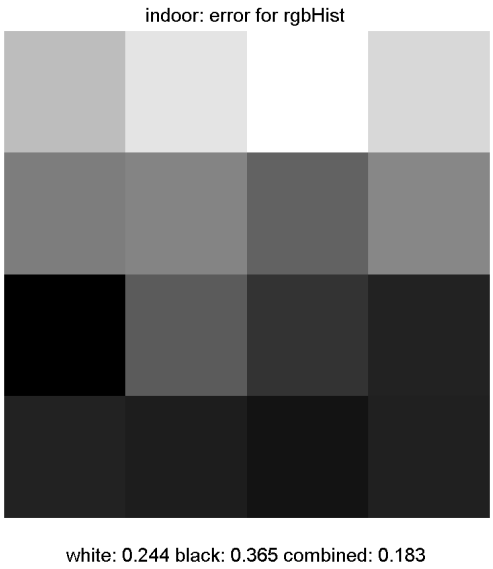



Figure 3.8: Using image tessellation to capture Spatial Information: Indoor-Outdoor

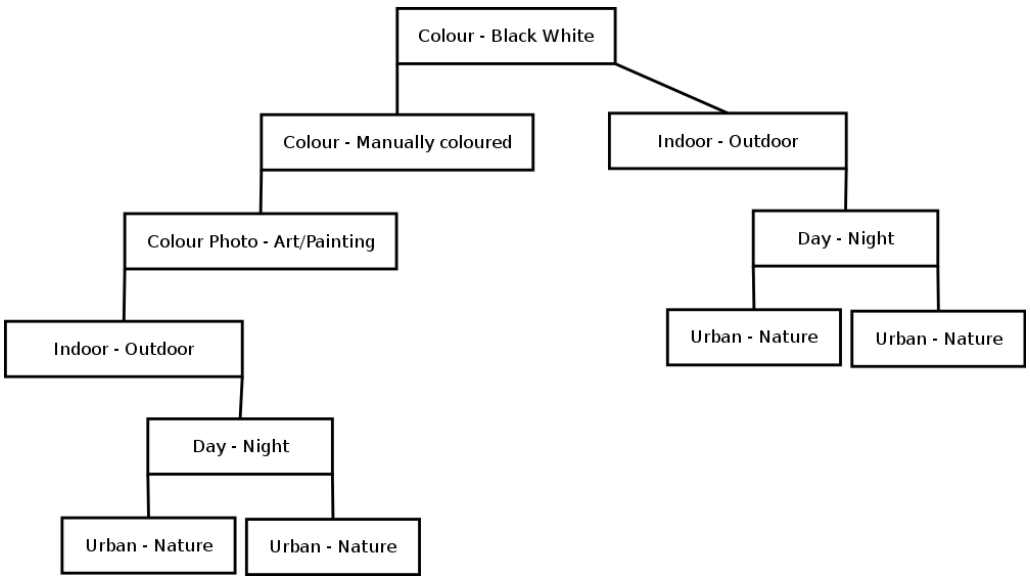



Figure 3.9: Hierarchy of Classifiers

# Chapter 4

# Results

This chapter reports on results achieved with the implemented prototype. For the evaluation a sample size of 2000 images is chosen for training and 1000 images are used for testing. The sample sizes were chosen for the purpose of faster testing, similar results are obtained when testing on the remaining 2474 images. Figure 4.1 shows the class distribution for the whole image database.

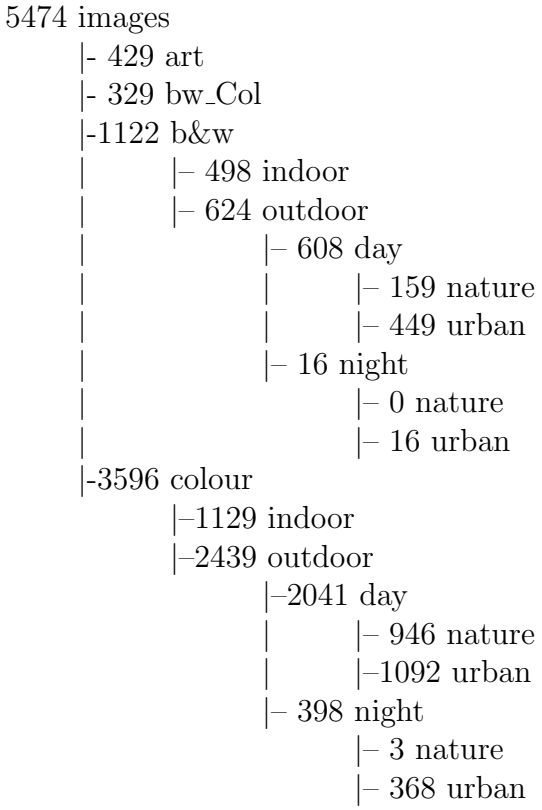



Figure 4.1: Class Distribution for the whole ImagEVAL database

As mentioned in the last chapter, a classifier is computed for each node of the tree shown in Figure 4.1. Each of these classifiers may have several sub-classifiers, one for each sub-block and feature. For the results shown in this chapter, rather then giving detail on each sub-classifier, the results are summarised for each attribute assigned to the images. The following figures and statistics are shown in each section.

For the comparison of features and also as a means to test their variance box plots were created with a (smaller) sample of 700 training and 200 test images. Each feature and attribute pair was tested with a randomized sample of images and 10 runs of training and testing. A box plot is a summary of a distribution, the box is limited by the lower quartile (25% of the data) and the upper quartile (75% of the data) values. The median is indicated by a horizontal line. To show the extent of the rest of the data "whiskers" are lines extending from each end of the box to the values lying above and below the quartiles but at most 1.5 times the box height. Other data points are considered as outliers, marked by crosses. Box plots offer a convenient way of observing variance and skewness of a distribution.

The vertical axis of the box plots is the classification error. The following abbreviations for the methods used are on the horizontal axis of the box plots:

**rgb** Histogram in the RGB space

**ohta** Histogram in the Ohta space

**luv** Histogram in the CIELUV space

**lab** Histogram in the CIELAB space

**srgb** Histogram in the normalised RGB space

**cStat** Colour statistics

**eStat** Edge statistics

**edge** Edge direction histogram

**edgeC** Edge direction coherence vector

**wav** Wavelet filter

**gabor** Gabor filter

**comb** Combined feature (all of above)

In this chapter, the task of classifying an image into the classes: “black and white”, “manually coloured”, “art” or “colour photo” is considered as a one-of-four classification problem. Therefore the total number of instances for the first 4 sections is 1000. Section 4.5-4.7 present the binary classification results.

The features selected for each problem are shown at the beginning of each subsection. The baseline is calculated as described in section 2.4.1, by division of the size of the bigger class by the total number of instances. This is the best result possible when guessing the class, without any feature available. To measure accuracy and retrieval effectiveness, the following statistics are collected for each classification task[1]:

**tn** true negatives: the instances correctly classified as negative;

**tp** true positives: the instances correctly classified as positive;

**fp** false positives: negative instances wrongly classified as positive;

**fn** false negatives: positive instances misclassified as negative.

**TP-Rate**: The true positive rate is the proportion of positive instances that were correctly reported as positive: TPr = tp / positives.

**FP-Rate**: The false positive rate is the proportion of negative instances that were erroneously reported as positive: FPr = fp / negatives.

**Precision**: The number of correctly classified instances as a proportion of the total number of instances classified as the class under consideration: tp/(tp+fp) and tn/(tn+fn).

**Recall**: The number of correctly classified instances as a proportion of the number of all instances of the class under consideration available: tp/(tp+fn) and tn/(tn+fp). As this prototype incorporates no reject option, this value is always the same as the TP-Rate.

**F-measure**: The harmonic mean of precision and recall $F = 2 * P * R/(P + R)$

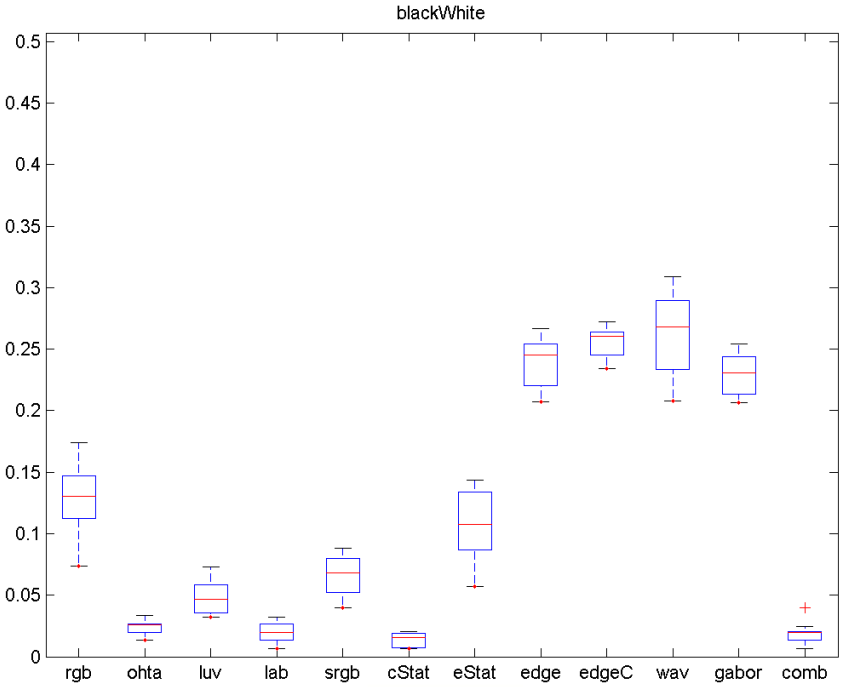
The first four of these values are summarised in form of a confusion matrix. This matrix has the following form:

| a | b | <– classified as |
|---|---|---|
| tn | fp | a |
| fn | tp | b |

[1]This is the format used by WEKA and was chosen to facilitate easy comparison.

As can be observed the sums of the rows of each class show how many instances belong to either class, whereas the sums of the columns show how many instances are classified to belong to each class.

## 4.1 Black and White




| | | |
|---|---|---|
| Features Chosen: | | Lab, cStat |
| Correctly Classified Instances: | 990 | 99.0 % |
| Incorrectly Classified Instances: | 10 | 1.0 % |
| Total Number of Instances: | 1000 | |
| Baseline: | 79.7% | |

Detailed Accuracy by Class

| Class | TP Rate | FP Rate | Precision | Recall | F-Measure |
|---|---|---|---|---|---|
| other types | 0.995 | 0.030 | 0.992 | 0.995 | 0.994 |
| black and white | 0.970 | 0.005 | 0.980 | 0.970 | 0.975 |

Confusion Matrix

| other types | black white | <– classified as |
|---|---|---|
| 793 | 4 | other types |
| 6 | 197 | black white |

The features used for the colour - black and white classifier are the colour statistics and the CIELAB histogram. The results achieved are very good, supporting the decision to choose these features. An interesting result shown in the box plot is that pure texture features perform significantly worse than colour features. While this re-

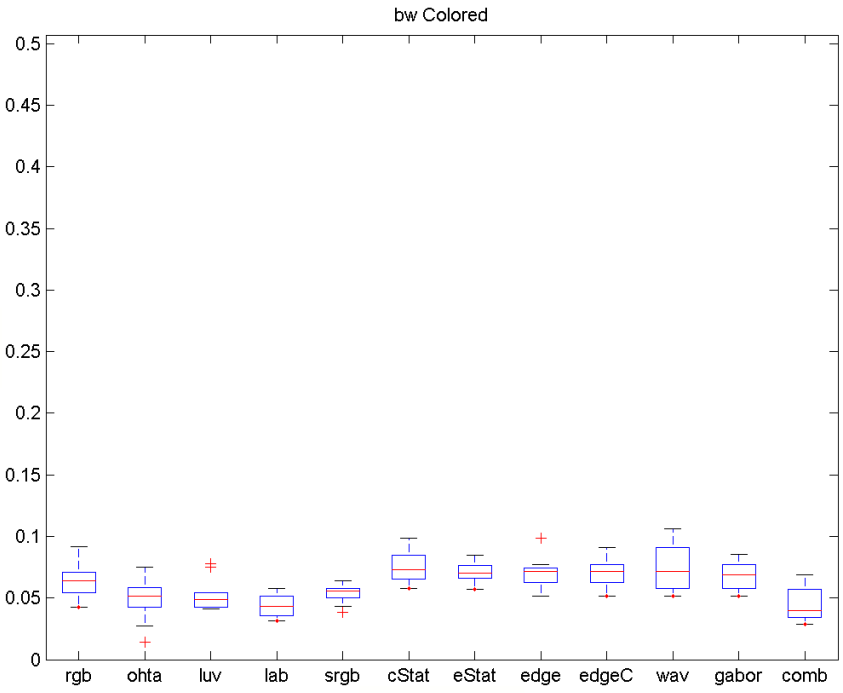
sult is not surprising it does show that the content (i.e. objects, image composition) of the images of the two classes is quite similar.

An analysis of the CIELAB histograms shows that this colour space is well suited for this problem because the chrominance is available separately. Colour images show a Gauss-like distribution in these two channels while black and white or greyscale images show a single spike around the value representing zero or achromacity and a very small percentage of other chrominance values. The separation of the classes in the CIELAB colour space is not perfect due to the inclusion of sepia images into the black and white class. The box plot shows that the RGB space, where chrominance as well as luminance is a product of the three channels, is not suited for this classification.

The colour statistics show good results because in black and white images there is nearly no variance between the three channels in the RGB colour space whereas colour images have a high variance. Again, sepia images are the reason for a small error. Both features show slightly better results for the four sub-blocks in the centre, probably because some images contain a border of a different colour/luminance. The combination of all features available did not yield a significant better result than each of the two features chosen.

Images of the colour class misclassified as black and white have a colour distribution similar to the sepia images or are very bright with little contrast. Nearly all misclassified black and white images are sepia images reported as manually coloured black and white.

## 4.2 Manually Coloured Black and White




| | | |
|---|---|---|
| Features Chosen: | | Lab |
| Correctly Classified Instances: | 961 | 96.1 % |
| Incorrectly Classified Instances: | 39 | 3.9 % |
| Total Number of Instances: | 1000 | |
| Baseline: | 93.4% | |

Detailed Accuracy by Class

| Class | TP Rate | FP Rate | Precision | Recall | F-Measure |
|---|---|---|---|---|---|
| other types | 0.983 | 0.348 | 0.976 | 0.983 | 0.979 |
| coloured BW | 0.652 | 0.017 | 0.729 | 0.652 | 0.688 |

Confusion Matrix

| other types | Coloured BW | <– classified as |
|---|---|---|
| 918 | 16 | other types |
| 23 | 43 | coloured BW |

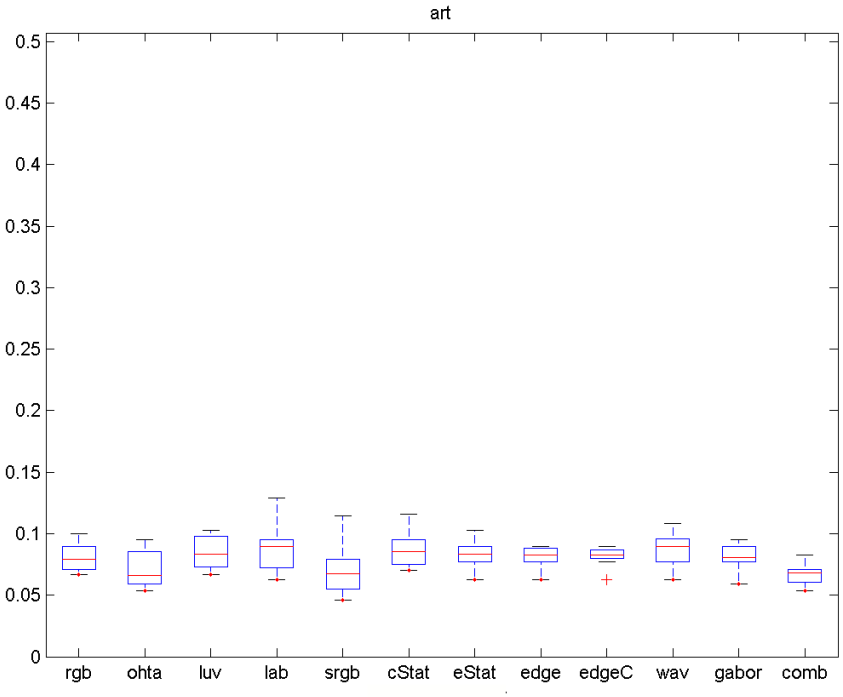
For the classification of images into the classes colour and manually coloured black and white images the CIELAB histogram is used. The results achieved are good, but compared to the classification in the previous section, the recall of the smaller class (manually coloured black and white) is not quite as satisfactory. This indicates

that the unequal distribution of instances is used to bias the classifier towards the larger class.

The box plot shows that all features available perform nearly equally well on this classification task and that a combination of all features does not yield a significantly better result. The CIELUV as well as the CIELAB colour space seems to be best suited. An analysis of the CIELAB histograms shows that the $L^*$ channel has a slightly different distribution for the two classes under consideration. While the colour images follow a near Gaussian distribution in this luminance channel, the distribution for the manually coloured images has two peaks. The second maximum represents noticeably higher luminance values than found in colour images. The third channel of the CIELAB histogram, representing yellow-blue chrominance also shows a significant variation between the two classes. The values for the manually coloured images show less variance around the zero value, representing achromaticity. The performance of the sub-block classifiers shows very little variance with respect to error-rate. As in the classification of black and white images, the texture features perform marginally worse than the colour features.

Misclassified images of the manually coloured class are assigned to the art or colour classes. The attribute “manually coloured image” is wrongly assigned to images of the colour class when an image contains uncommon colours, as with outdoor images with a lot of fog, but also to an aerial image. As mentioned above sepia images are difficult to classify into this class or the black and white class.

## 4.3 Art




| | | |
|---|---|---|
| Features Chosen: | | srgb, wav |
| Correctly Classified Instances: | 949 | 94.9 % |
| Incorrectly Classified Instances: | 51 | 5.1 % |
| Total Number of Instances: | 1000 | |
| Baseline: | 91.4% | |

Detailed Accuracy by Class

| Class | TP Rate | FP Rate | Precision | Recall | F-Measure |
|---|---|---|---|---|---|
| other types | 0.985 | 0.430 | 0.961 | 0.985 | 0.972 |
| art | 0.570 | 0.015 | 0.778 | 0.570 | 0.658 |

Confusion Matrix

| other types | art | <– classified as |
|---|---|---|
| 900 | 14 | other types |
| 37 | 49 | art |

The features selected for the classification of images into the classes photographic image - artistic reproduction/ paintings are the sRGB histogram and the wavelet filter. Similar to the observations made with the classification of images into the colour - manually coloured classes, the box plot does not show a single feature

outperforming the others and the combination of features does not seem promising. Also the results obtained show a lower recall rate on the smaller class (paintings).

The CIELAB histogram has slightly higher luminance values for the “art” class but a higher deviation is found in the green channel of the sRGB histogram. The distribution of the histograms in this colour space is generally less spread out for images belonging to the class of paintings. The wavelet filter shows higher values for colour images, this represents texture detail, indicating that paintings are less structured than photos of (natural) scenes. Both features show little variance with regard to the results of the sub-block classifiers; however slightly better values are observed for the off-centre blocks. This could be a result of the general layout of paintings, with the subject in the centre and less detail near the borders.

As mentioned above some manually coloured black and white images are wrongly assigned to this class. Furthermore colour images of richly decorated indoor scenes (palaces, gold plating) are considered as art, as are some outdoor scenes. Difficult to interpret is the reason why many art images are misclassified as colour photos. These images mostly have a realistic colour layout and a higher level of detail.

## 4.4 Colour Photo

| | | |
|---|---|---|
| Correctly Classified Instances: | 933 | 93.3 % |
| Incorrectly Classified Instances: | 67 | 6.7 % |
| Total Number of Instances: | 1000 | |
| Baseline: | 64.6% | |

Detailed Accuracy by Class

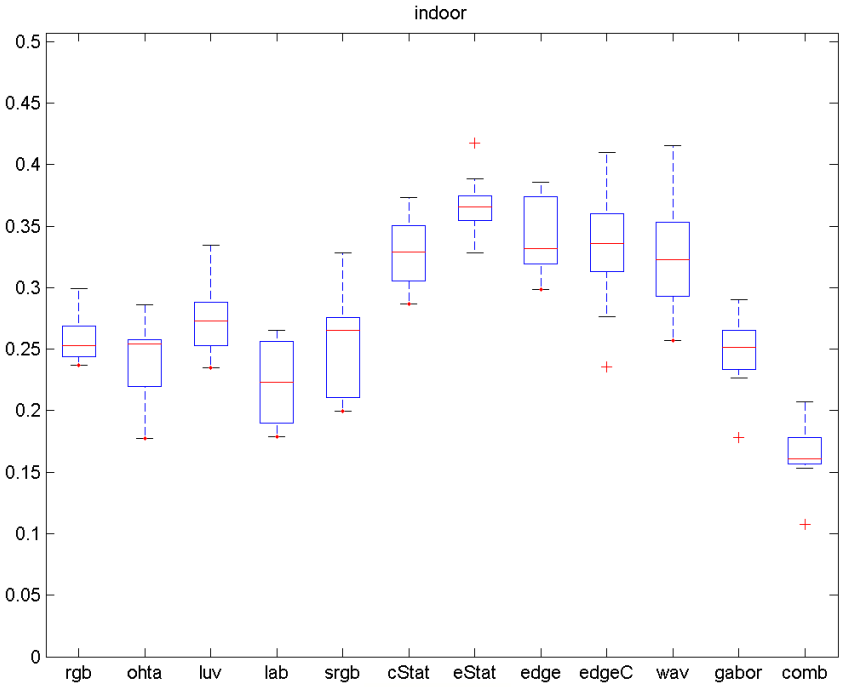
| Class | TP Rate | FP Rate | Precision | Recall | F-Measure |
|---|---|---|---|---|---|
| other types | 0.862 | 0.028 | 0.944 | 0.862 | 0.901 |
| colour photo | 0.972 | 0.138 | 0.928 | 0.972 | 0.949 |

Confusion Matrix

| other types | colour photo | <– classified as |
|---|---|---|
| 305 | 49 | other types |
| 18 | 628 | colour photo |

This table is a summary of results obtained so far in that it shows the error in classification for colour photos versus the other types in question. Through the hierarchic classification, the classifications performed until this point discriminate black and white images, manually coloured images and paintings from a general "colour" class, therefore what we are left with here are photographic colour images. Not considered, but also not part of the training sample, are black and white artistic reproductions. For the further classification only colour photos and black and white photos are considered.

## 4.5 Outdoor - Indoor




| | | |
|---|---|---|
| Features Chosen: | rgb, Lab, edgeC, wav | |
| Correctly Classified Instances: | 693 | 83.5 % |
| Incorrectly Classified Instances: | 137 | 16.5 % |
| Total Number of Instances: | 830 | |
| Baseline: | 63.5% | |

Detailed Accuracy by Class

| Class | TP Rate | FP Rate | Precision | Recall | F-Measure |
|---|---|---|---|---|---|
| outdoor | 0.869 | 0.223 | 0.870 | 0.869 | 0.869 |
| indoor | 0.777 | 0.131 | 0.775 | 0.777 | 0.776 |

Confusion Matrix

| outdoor | indoor | <– classified as |
|---|---|---|
| 456 | 69 | outdoor |
| 68 | 237 | indoor |

For the classification of images into the indoor or outdoor class the following features are selected: RGB and CIELAB histograms, coherent edge direction histogram and the wavelet filters. The result obtained in the classification process is not as good as those covered so far. However the results obtained by other authors (82% to

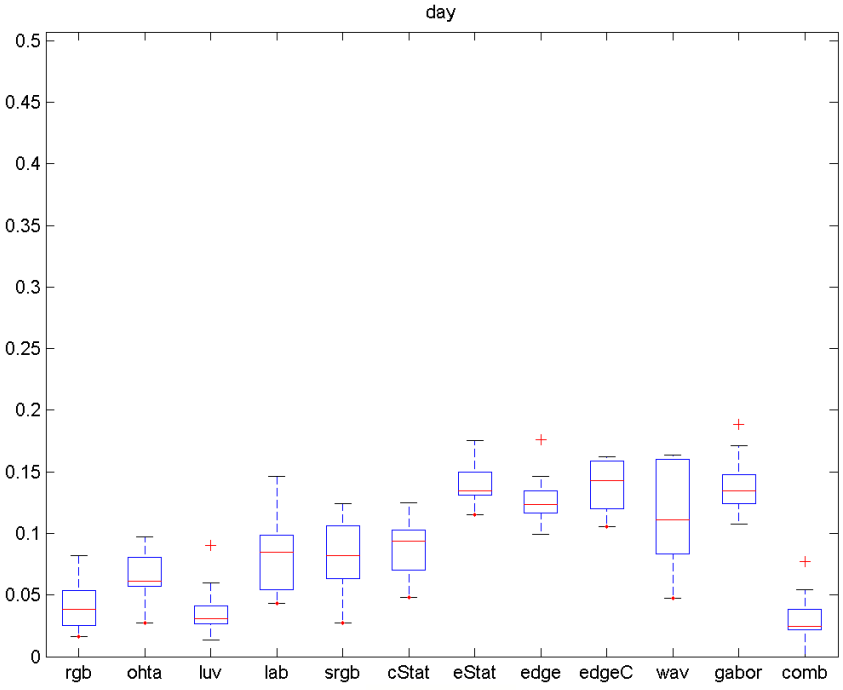
93% – see section 2.1.2) are comparable because their training and test sets are often smaller and ambiguous images are eliminated beforehand. An interpretation of the box plot is that generally colour features seem to perform better than texture features, what is striking is that the combination of all features yields a much better result than any single feature. Also the Gabor filter performs as well as the colour features.

An analysis of the RGB and CIELAB histograms shows that indoor images have slightly less luminance and (therefore) less highly saturated pixel values. Also the sub-block classifiers for these features perform better for the upper half of the images, this can be attributed to the presence or absence of a sky or alternatively that this area best reflects lighting conditions. The values obtained through the Gabor filters show higher values for the indoor class, indicating more structure or highly textured images. For the final implementation of the prototype the Gabor filter was deselected because of its high computational costs, however the wavelet filters, selected instead, show a similar response for this classification. As with the Gabor filter the result of the wavelet operation shows higher values for the indoor class. The coherent edge direction histograms show higher values for the outdoor class, seemingly contradicting this observation. Both classes show peaks at the values indicating horizontal, vertical and diagonal structures $-90$, $-45$, 0, 45 and 90 degrees. This effect is somewhat more pronounced for the indoor class.

The results obtained in combining the said features are similar to the combination of all features, as indicated by the feature “comb” in the box plot. It has been indicated in several papers that a combination of features has most effect when combining features of the “colour” group with those of the “texture” group.

The reason for indoor images to be classified as outdoor often seems to be lighting conditions caused by the presence of windows or doors. Also a strong presence of green or, in the case of black and white images, a bright background seems to bias the images into this class. The outdoor images classified as indoor either show very high detail or cluttering of the image or depict outdoor scenes with lighting common to indoor images, e.g. during dawn and dusk.

## 4.6 Night - Day




| | | |
|---|---|---|
| Features Chosen: | | Luv |
| Correctly Classified Instances: | 435 | 96.5 % |
| Incorrectly Classified Instances: | 16 | 3.5 % |
| Total Number of Instances: | 451 | |
| Baseline: | 86.8% | |

Detailed Accuracy by Class

| Class | TP Rate | FP Rate | Precision | Recall | F-Measure |
|---|---|---|---|---|---|
| night | 0.857 | 0.024 | 0.783 | 0.857 | 0.818 |
| day | 0.976 | 0.143 | 0.985 | 0.976 | 0.980 |

Confusion Matrix

| night | day | <– classified as |
|---|---|---|
| 36 | 6 | night |
| 10 | 399 | day |

The classification of images into the day - night classes is achieved using the CIELUV histograms. As can be observed in the box plot, colour features perform better than texture features and the combination does not bring an improvement over using the CIELUV colour space. The accuracy achieved is acceptable and the recall rates are

good for both classes. The means of the CIELUV histograms for this classification problem show a distinct deviation in the luminance channel. As can be expected photos during daylight are much brighter than night shots. Interestingly enough, the chrominance values of the night class are higher than those of the day class. This might be a conversion error due to the little luminance and therefore little hue information available or alternatively the presence of light emitting objects. The results for the sub-blocks are slightly better for the upper half of the images.

The reason for misclassification of day scenes is often a very dark sky and in one instance an underwater image with black background. Night scenes misclassified as day were taken during dusk, ambiguous even to a human observer. Three images show city scenes with man-made lighting.

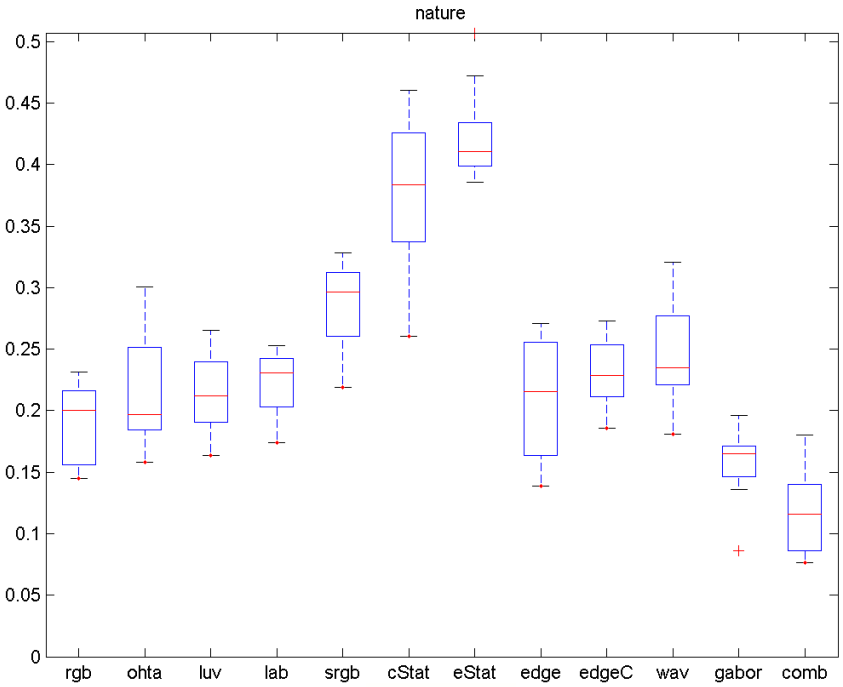
As an example of the output of Weka consider the decision tree shown in Figure 4.2. The input for this classifier is the feature vector for the top left sub-block, labelled luvHist1 through luvHist60 for the 3 channels comprised of 20 bins each. The 1657 feature vectors exported to Weka are split at 66% into training and test data. This classifier achieves a hit-rate of 92.2% on the test data. The decision tree is surprisingly small and performs quite well. The confidence factor is set to 0.01 and the minimum objects per leaf to 20, see section 2.4.3 for details on these parameters.

The analysis of a decision tree is helpful because it shows which features are important for classification. It also shows why a certain class is chosen. A comparable analysis is not possible with the $k$-NN classifier. In the tree the relation *isDay* is shown, therefore 0 represents night and 1 day. At each node an estimate of instances that reach the leaf is given (in brackets).

```
luvHist27 <= 0.002887
|   luvHist2 <= 0.137915
|   |   luvHist25 <= 0.027062: 1 (1466.0/97.0)
|   |   luvHist25 > 0.027062: 0 (27.0/11.0)
|   luvHist2 > 0.137915: 0 (52.0/13.0)
luvHist27 > 0.002887
|   luvHist8 <= 0.164309: 0 (89.0/18.0)
|   luvHist8 > 0.164309: 1 (23.0/3.0)
```

Figure 4.2: Weka output: decision tree for day (class 1) - night (class 0) problem

## 4.7 Urban - Nature




| | | |
|---|---|---|
| Features Chosen: | rgb, edgeC, wav | |
| Correctly Classified Instances: | 393 | 87.1 % |
| Incorrectly Classified Instances: | 58 | 12.9 % |
| Total Number of Instances: | 451 | |
| Baseline: | 63.2% | |

Detailed Accuracy by Class

| Class | TP Rate | FP Rate | Precision | Recall | F-Measure |
|---|---|---|---|---|---|
| urban | 0.917 | 0.194 | 0.871 | 0.917 | 0.893 |
| nature | 0.806 | 0.083 | 0.872 | 0.806 | 0.838 |

Confusion Matrix

| urban | nature | <– classified as |
|---|---|---|
| 243 | 22 | urban |
| 36 | 150 | nature |

For the classification problem nature - urban the following low-level features where selected: RGB histogram, coherent edge direction histogram and wavelet filters. As with the indoor - outdoor classification this seems to be a harder problem. The 87% hit-rate achieved lies close to the results reported on the problem by other papers

(see section 2.1.3). The box plot shows that colour features as well as texture features are suited for the classification, also the combination of features promises an improvement.

The analysis of the RGB histograms does not yield explicit evidence other than that all channels have higher values in the nature class. The distribution in the CIELAB colour space confirms a higher luminance for nature images. This can be attributed to a different illumination, rather counter-intuitive is that there is no abundance of green in nature images. The histograms of the coherent edge direction suggest more ordered structures in the urban class. The peaks and valleys are more pronounced for this class. While both classes have a maximum at 0°, representing horizontal detail, the urban class has maxima at -90 and 90 degrees, representing vertical detail, while the nature class has a near equally spaced distribution. This feature therefore draws on the assumption that nature images have smaller, chaotic structures than urban images. Both the Gabor and the wavelet filters show higher values for the urban class at all scales and orientations, confirming this observation.

The sub-blocks in the centre have higher accuracy for the RGB and the wavelet feature. For the coherent edge direction histogram the best results are achieved in the lower part of the images.

Natural images classified as urban show a very highly structured composition. This is caused by trees, rocks or landscape formations e.g. canyons. Some cases also show man made structures in the foreground, e.g. castles or walls. Abundance of sky or green plants as well as the presence of lakes and rivers in urban scenes seems to be the main reason for the wrong assignment of the attribute nature.

| Attribute | rgb | ohta | luv | lab | srgb | colour-stats |
|---|---|---|---|---|---|---|
| art | 91.9 | 92.9 | 91.5 | 91.2 | ***93.1** | 91.3 |
| blackWhite | 87.4 | 97.6 | 95.1 | 98.0 | 93.5 | ***98.6** |
| bw_Col | 93.6 | 95.1 | 94.7 | ***95.7** | 94.7 | 92.5 |
| day | 95.9 | 93.6 | ***96.2** | 91.6 | 91.8 | 91.1 |
| indoor | 74.2 | 75.7 | 72.4 | ***77.7** | 74.8 | 67.1 |
| nature | *80.9 | 78.4 | 78.6 | 77.8 | 71.3 | 61.9 |

| Attribute | edge-stats | edges | edges C. | Wavelet | gabor | combined |
|---|---|---|---|---|---|---|
| art | 91.6 | *91.8 | *91.8 | 91.2 | *91.8 | 93.3 |
| blackWhite | *89.3 | 76.1 | 74.4 | 73.7 | 76.9 | 98.1 |
| bw_Col | 93.0 | 93.0 | 92.9 | 92.5 | *93.2 | 95.6 |
| day | 85.9 | 87.3 | 86.1 | *88.6 | 85.9 | 96.8 |
| indoor | 63.4 | 65.8 | 66.9 | 67.6 | *75.4 | 83.5 |
| nature | 57.6 | 78.8 | 77.0 | 75.4 | ***84.2** | 88.1 |

Table 4.1: Comparison of Features - the percentage of correctly classified images is given; top: colour histograms, bottom: texture features. The "combined" feature is a combination of all features available. The best single feature is bold, an asterisk marks best result of each sub-table.

## 4.8 Overall Result

Table 4.1 is a summary of the feature comparisons, only the median hit-rate of each attribute and feature pair (taken from the box plots of the previous sections) is shown. As has been mentioned before, the combination of a texture and a colour feature is more promising then the combination of two features of the same type. Therefore not only the best overall result is marked bold, but the best texture/colour feature is marked by an asterisk.

| | | |
|---|---|---|
| Correctly Classified Instances: | 710 | 71.0 % |
| Incorrectly Classified Instances: | 290 | 29.0 % |
| Total Number of Instances: | 1000 | |
| Baseline: | 20% | |

An accuracy of 71% is achieved on the whole problem, i.e. assigning up to 4 attributes to an image. The images considered to be incorrectly classified have one or more wrongly assigned attributes. This value is not simply the sum of errors reported in the previous sections. The baseline error, the assignment of the class with the highest probability, would yield 20% hit-rate with this image database; however this would also imply a recall rate of 0% for all other classes in question because all images would be assigned to the same class. The overall hit-rate on the 3474 images

not used for training is 72.4%. This suggests that the classifier generalises well and can be expected to perform comparably on similar data.

To compare the results on a different dataset two test runs were made on a part of the Corel image database used in [LW03] and [WLW01]. A sample of 500 images was selected; Table 4.2 shows the chosen directories and the attributes manually assigned to them. Each directory contains 100 images.

| **directory** | **ground truth** |
|---|---|
| autumn | colour Photo, outdoor, day, nature |
| night | colour Photo, outdoor, night, urban |
| kitchen | colour Photo, indoor |
| paintings | art |
| ny_city | colour Photo, outdoor, day, urban |

Table 4.2: Selected groups from the Corel Image Database

In one test run the classifier, trained on the ImagEVAL data was applied to the 500 images. The percentage of correctly classified images is much lower than the one reported above with 47.2%. This can be attributed to the differences in prior probabilities as well as image composition. Notably, all 100 art images, mostly aquarelles, are misclassified.

In the second test run the classifier was trained on 400 images (using a random draw) and tested on the remaining 100. The percentage of correctly classified images rises to 80% (art: hit-rate of 95.9%). However these results do not give insight into how well the classifier generalises. The discrepancy between these two results can rather be explained by the different domain of the ImagEVAL images compared to the 500 images selected from the Corel image database.

## 4.9 Sample Application: Query by Image

A common application in CBIR is "query by image". To demonstrate how the classification achieved in this thesis can be of benefit a small application was implemented and tested on the 100 images from the Corel database previously used for testing the classifier. A query image is selected by the user and the feature vectors and semantic classes computed. Certainly simplified, as compared to existing CBIR systems, the RGB feature vector is used to find the images with the smallest Euclidean distance to the query image. The returned image list can then be filtered, excluding images that have different image semantics.

Figure 4.3 shows the result of the query without, and Figure 4.4 with filtering of the images through the use of semantic information gathered by the prototype. The image at the top left is the query image. As can be observed the query without filtering shows images of different semantic classes. Included are paintings and nature images although the query image clearly shows a town's skyline. Only one out of the eight returned images can be considered a match. For this image, and also others tested, the filtered results work much better. Only one mismatch, an image of stalagmites, is returned. Of course these are optimistic conditions, but the comparison shows that semantic information might be of benefit for comparable applications.

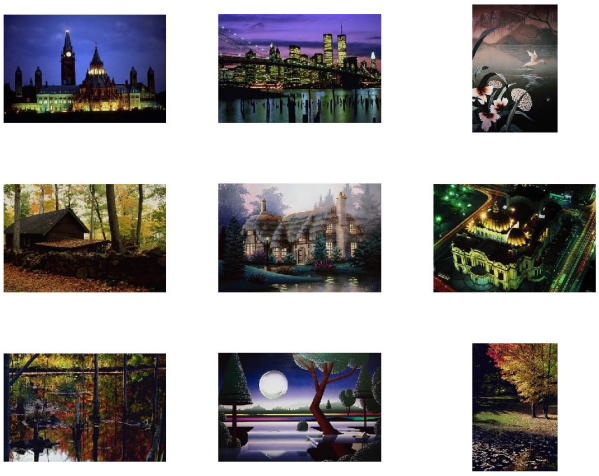


Figure 4.3: Query by image, RGB Euclidean distance, top-left is query image

Figure 4.4: Query by image using semantic Classes, RGB Euclidean distance, top-left is query image

# Chapter 5

# Conclusion

This thesis shows that reasonable results can be obtained in extracting image semantics with the aid of statistical methods. An accuracy of 71% is achieved on the problem posed by the ImagEVAL project. The image attributes extracted are: black and white photo, colour photo, manually coloured photo, artistic reproduction, outdoor, indoor, night, day, nature, urban.

The review of literature available in research of automatic image classification is helpful in selecting low-level image features. Some interesting approaches to the statistical modelling are also shown. Several suggestions in the literature are taken up in the implementation of the prototype. The hierarchical classification makes use of knowledge about the problem-domain. The attributes to be assigned to the images are mutually exclusive and cover a wide spectrum of input images. Features used are: colour histograms in several colour spaces, colour statistics, wavelet and Gabor filters and edge detectors.

The prototype developed is used for two aims. The result of image classification is used to evaluate and compare the discrimination power of several features on the given problems. Secondly, conclusions about the reasons why particular features are suited for a problem are drawn. This is done through an analysis of results and variables available at sub-stages during the training and testing phases. This seems to suggest an iterative development of the image classifiers, as more information about the problem domain is gained in the process.

The ambiguity of natural language, where, for example, “nature” and “urban” is not explicitly defined and leads to problems in classification of images that cannot be accurately described with either word, is an unsolved problem. An interesting

addition to this thesis would be a comparison of the results obtained by the prototype with the results obtained by humans on the same image database (manual classification).

It is quite impossible to generalise the obtained results to a "total" class of all images available world-wide. This hampers comparison with other research because a sub-set of possible images and therefore a biased classification is always chosen.

To make use of the biggest image collection in the world, the Internet, the implementation of more attributes would be of interest. For example business graphic - photograph. The integration of meta-information of the images, e.g. size of file or information stored in Exchangeable Image File Format (EXIF) tags, would also be of interest.

An improvement of feature extraction speed would be of advantage, not only in use of the prototype with large image databases, but also to rapidly test other parameter settings and low-level features. The results obtained with the prototype are comparable to those found in literature.

# Bibliography


[AK97] Yannis S. Avrithis and Stefanos D. Kollias. Fuzzy image classification using multiresolution neural networks with applications to remote sensing. In *Digital Signal Processing Proceedings, 1997. DSP 97., 1997 13th International Conference on*, volume 1, pages 261 – 264, July 1997.

[BOV03] Annalisa Barla, Francesca Odone, and Alessandro Verri. Old fashioned state-of-the-art image classification. In *Image Analysis and Processing, 2003.Proceedings. 12th International Conference on*, pages 566 – 571, Sept. 2003.

[CHL03] Florin Cutzu, Riad Hammoud, and Alex Leykin. Estimating the photorealism of images: Distinguishing paintings from photographs. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 2, pages II – 305–12, June 2003.

[DHS00] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley-Interscience Publication, 2000.

[DJP+04] R.P.W. Duin, P. Juszczak, P. Paclik, E. Pekalska, D. de Ridder, and D.M.J. Tax. Prtools4 a matlab toolbox for pattern recognition. URL: http://www.prtools.org/, 2004.

[DY02] Shou-Bin Dong and Yi-Ming Yang. Hierarchical web image classification by multi-level features. In *Machine Learning and Cybernetics, 2002. Proceedings. 2002 International Conference on*, volume 2, pages 663– 668, Nov. 2002.

[Fai98] Mark D. Fairchild. *Color Appearance Models*. Addison Wesley, 1998.

[GL00] Qian Gu and Stan Z. Li. Combining feature optimization into neural network based face detection. In *Pattern Recognition, 2000.*

*Proceedings. 15th International Conference on*, volume 2, pages 814 – 817, Sept. 2000.

[IA99] Qasim Iqbal and J.K. Aggarwal. Applying perceptual grouping to content-based image retrieval: building images. *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on*, 1, June 1999.

[Ima05] ImagEVAL. Imageval project description. URL: http://www.imageval.org, 2005.

[LF05] Siwei Lyu and Hany Farid. How realistic is photorealistic? *Signal Processing, IEEE Transactions on*, 53(2):845 – 850, Feb. 2005.

[LGO00] Jia Li, Robert M. Gray, and Richard A. Olshen. Multiresolution image classification by hierarchical modeling with two-dimensional hidden markov models. *Information Theory, IEEE Transactions on*, 46(5):1826 – 1841, Aug. 2000.

[LSS05] Jiebo Luo, Andreas E. Savakis, and Amit Singhal. A bayesian network-based framework for semantic image understanding. *Pattern Recognition*, 38(6):919–934, 2005.

[LW03] Jia Li and J.Z. Wang. Automatic linguistic indexing of pictures by a statistical modeling approach. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(9):1075 – 1088, Sept. 2003.

[MM96] B. S. Manjunath and W.Y. Ma. Texture features for browsing and retrieval of image data. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI - Special issue on Digital Libraries)*, 18(8):837–42, Aug 1996.

[PCH+02] Salil Prabhakar, Hui Cheng, John C. Handley, Zhigang Fan, and Ying wei Lin. Picture-graphics color image classification. In *Image Processing. 2002. Proceedings. 2002 International Conference on*, volume 2, pages II–785 – II–788, Sept. 2002.

[QFW04] Xipeng Qiu, Zhe Feng, and Lide Wu. Boosting image classification scheme. In *Multimedia and Expo, 2004. ICME '04. 2004 IEEE International Conference on*, volume 2, pages 1271 – 1274, June 2004.

[RBK96] Henry A. Rowley, Shumeet Baluja, and Takeo Kanade. Neural network-based face detection. In *Computer Vision and Pattern Recognition, 1996. Proceedings CVPR '96, 1996 IEEE Computer Society Conference on*, pages 203 – 208, June 1996.

[RBK98] Henry A. Rowley, Shumeet Baluja, and Takeo Kanade. Rotation invariant neural network-based face detection. In *Computer Vision and Pattern Recognition, 1998. Proceedings. 1998 IEEE Computer Society Conference on*, pages 38 – 44, June 1998.

[RTG98] Yossi Rubner, Carlo Tomasi, and Leonidas J. Guibas. The earth mover"s distance as a metric for image retrieval. Technical report, Stanford University, Stanford, CA, USA, 1998.

[SP98] Martin Szummer and Rosalind W. Picard. Indoor-outdoor image classification. In *Content-Based Access of Image and Video Database, 1998. Proceedings., 1998 IEEE International Workshop on*, pages 42 – 51, Jan. 1998.

[SS94] M. Stricker and M. Swain. The capacity of color histogram indexing. In *Computer Vision and Pattern Recognition 1994*, pages 704–708, 1994.

[SWS+00] Arnold W.M. Smeulders, Marcel Worring, Simone Santini, Amarnath Gupta, and Ramesh Jain. Content-based image retrieval at the end of the early years. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(12):1349 – 1380, Dec. 2000.

[VFJZ01] Aditya Vailaya, Mário A. T. Figueiredo, Anil K. Jain, and Hong-Jiang Zhang. Image classification for content-based indexing. *Image Processing, IEEE Transactions on*, 10(1):117 – 130, Jan. 2001.

[VJZ98] Aditya Vailaya, Anil K. Jain, and Hong-Jiang Zhang. On image classification: City vs. landscape. In *Content-Based Access of Image and Video Libraries, 1998. Proceedings. IEEE Workshop on*, pages 3 – 8, June 1998.

[Wag99] Thomas Wagner. Texture analysis. *Handbook of Computer Vision and Applications, Signal Processing and Pattern Recognition*, 2:275–308, 1999.

[WF05] Ian H. Witten and Eibe Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, San Francisco, 2005. www.cs.waikato.ac.nz/ml/weka.

[WLW01] J.Z. Wang, Jia Li, and G. Wiederhold. Simplicity: semantics-sensitive integrated matching for picture libraries. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23(9):947 – 963, Sept. 2001.