Escola Politècnica Superior de Castelldefels

UNIVERSITAT POLITÈCNICA DE CATALUNYA

MASTER THESIS

TITLE: Color Based Image Classification and Description

MASTER DEGREE: Master in Science in Telecommunication Engineering & Management

AUTHOR: Sergi Laencina Verdaguer

DIRECTOR: Francesc Tarrés Ruiz

DATE: October 20th 2009

Títol: Color Based Image Classification and Description

Autor: Sergi Laencina Verdaguer

Director: Francesc Tarrés Ruiz

Data: 20 d'octubre de 2009

Resum

L'objectiu d'aquest projecte consisteix en poder descriure i ordenar fotografies de manera automàtica en funció del seu color. Per a aconseguir-ho, s'han estudiat dos descriptors de color definits per l'estàndard MPEG-7: el Color Layout Descriptor i el Dominant Color Descriptor.

En primer lloc s'ha analitzat l'estàndard MPEG-7. Aquest ofereix mecanismes per tal de descriure qualsevol tipus de contingut multimèdia. En concret, en aquest projecte s'han estudiat les descripcions visuals de l'estàndard i en particular les del color.

A continuació s'han escollit com a descriptors de color el Color Layout Descriptor (CLD) i el Dominant Color Descriptor (DCD). El primer és un descriptor molt compacte i dissenyat per representar de manera eficient la distribució espacial dels colors en una imatge. El segon proporciona una descripció compacta dels colors representatius d'una imatge.

Per a cadascun d'ells, se n'ha realitzat una implementació i se n'han millorat els seus resultats. Pel cas del CLD s'ha millorat el seu rendiment fent el descriptor més robust als canvis de brillantor i contrast de les imatges. Al DCD se li ha afegit una etapa prèvia a l'obtenció del descriptor per tal de reduir el seu temps d'extracció.

Un cop estudiats i millorats els dos descriptors de color, s'han integrat els seus resultats en una mesura comuna per tal d'obtenir els avantatges que cadascun d'ells aporta.

Posteriorment, s'han definit i realitzat una sèrie de tests a diferents persones per tal d'avaluar els resultats proporcionats per cadascun dels tres mètodes implementats: el CLD, el DCD i la integració dels dos descriptors. D'aquesta manera s'ha pogut comprovar si els resultats dels mètodes distaven molt dels resultats de les persones.

Finalment, s'han emprat els resultats obtinguts en aquest projecte per tal de desenvolupar una pàgina web on l'usuari pot pujar les seves fotografies i aquestes li son retornades ordenades en funció del color.

Title: Color Based Image Classification and Description

Author: Sergi Laencina Verdaguer

Director: Francesc Tarrés Ruiz

Date: October 20th 2009

Overview

The objective of this project is to describe and arrange automatically photos only according to its color. Two color descriptors defined by the MPEG-7 standard have been studied in order to achieve this objective: the Color Layout Descriptor and the Dominant Color Descriptor.

In the first place it has been analyzed the MPEG-7 standard. This standard provides the guidelines to describe all kind of multimedia content. In this project the visual descriptors of the MPEG-7 and especially its color descriptors have been studied.

Then, the Color Layout Descriptor (CLD) and the Dominant Color Descriptor (DCD) have been chosen as color descriptors. On the one hand, the CLD is a very compact and resolution-invariant representation of color and is designed to efficiently represent the spatial distribution of colors. On the other hand, the DCD provides a compact description of the representative colors in an image or image region.

An implementation and different improvements have been performed for each color descriptor. The robustness of the CLD against brightness and contrast changes of the images has been improved as well as a pre-process step has been added before the DCD extraction in order to reduce its computing time.

Once both descriptors have been studied and improved successfully, the CLD and the DCD have been integrated in a common measure in order to take the advantages that each one provides.

Later, an evaluation method in order to know if the results obtained by the descriptors are similar to the results obtained by human beings has been defined.

Finally, it has been decided to apply the results obtained during the execution of this project to a real application. It has been developed a web application which allows classifying uploaded images depending on its color.

Voldria agrair a totes aquelles persones que m'han donat suport durant la realització d'aquest projecte: al meu tutor Francesc Tarrés per la seva paciència, dedicació i per haver-me guiat en aquest projecte. A en Toni Rama i a en Pere Quim per totes les reunions i correccions que m'han fet. A l'Oriol Solé per ajudar-me amb la Web. A la Núria Berga amb l'anglès. A tota la gent que ha realitzat els tests. A tota la família en general i en particular als meus pares i a la meva germana Eulàlia, que sempre els he tingut al costat. I sobretot a la Jana, per tindre una paciència enorme amb mi, per tot el suport que m'ha donat i per aguantar-me en els moments més difícils.

Moltes gràcies a tots!

Finalment voldria dedicar aquest projecte al meu avi, per tot el suport i comprensió que sempre va mostrar cap a mi.

INDEX

INTRODUCTION

The photography's world has changed very fast in the last decades. Some years ago, almost all the cameras used film and the developed photos were used to design albums as a sequence of photos, selected by the user manually and then assembled. The albums ease the process of showing these photos to friends and family and facilitate the storage of picture events.

Over the years, digital photography has gained ground to the classical photography. The quality and variety of cameras have been increasing day after day. Also, nowadays, nearly all mobile phones include digital cameras.

However, digital cameras use memory cards instead of films and has prompted that only in very special cases photos are developed. And where do these photos remain? On computers or uploaded on the Web. The following table shows the total number of images warehoused on the main Webs:

Table 0.1 Image warehouses on the Web on April 7, 2009 [1]

Web Site	Number of total stored images (in billions)
ImageShack	20
Facebook	15
PhotoBucket	7.2
Flickr	3.4
Multiply	3

In this sense, the ease to showing photos through an album to friends and family has been lost. A necessity has been created: the digital albums.

Nowadays, there are a lot of applications which let the users generating electronic albums, even creating and sharing these albums on-line. Some examples are Facebook, Flickr, Picasa or iPhoto. However, in most cases is the user who has to select the order and composition of the photos. Consequently, in this point emerges the question: Why is not possible to do this arrangement and composition automatically? This solution would save time to the users when creating electronic albums as well as the possibility to generate more sophisticate compositions of the photos.

This automatic arrangement requires two steps in order to obtain it in a properly way:

- Describe the content of the photos: measures to characterize the photos.
- Arrange the pictures: according to well defined criteria.

A correct description of the photos can be obtained with standard tools like MPEG-7. MPEG-7 allows describing the media content from low to high abstraction. It allows describing photos with lower abstraction levels according to its shape, its size, its texture, its color or its position (where in the image the object can be found?); the highest level would give semantic information: “This is an image of a girl wearing black trousers and a brown dog in a snow-covered landscape”, for instance as can be seen in **Fig. 0.1** Intermediate levels of abstraction may also exist.

Fig. 0.1 Girl and dog in a snow-covered landscape

Once the photos have been properly described is possible to perform an arrangement. This arrangement can be done according to artistic factors, such as: color, people (faces) appearing in the photo, location, the same scene... or a combination of all of them. Also, it would be possible to join synchronized music with the photos, or correct the photos’ framing. Finally, as a result, it is obtained an electronic album.

Taking into account this premise, the objective of this project is to describe and arrange automatically photos only according to its color. What we expect is to be able to cluster a set of images according to its color in order to perform an arrangement, as can be seen in the following figure:

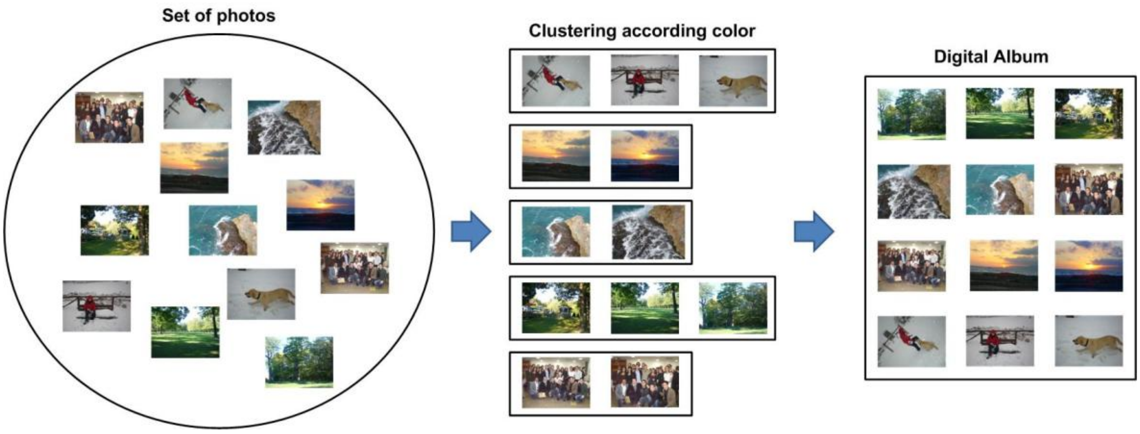

Fig. 0.2 Clustering and ordering photos

It has been selected the color because is the basis of any image and is one of the most important characteristics of any photo.

In order to achieve this objective, MPEG-7 standard has been studied, its visual descriptors and especially the color descriptors. This work is reflected in Chapters 1 and 2 were the context of this project has been introduced.

In Chapters 3 and 4 two color descriptors, the Color Layout Descriptor and the Dominant Color Descriptor, have been studied. For each one, the standard has been followed during the implementation and different tests have been carried out to adjust its performance.

In Chapter 5 it has been integrated the Color Layout Descriptor (CLD) and the Dominant Color Descriptor (DCD) in a common measure in order to obtain the advantages of each one.

Once all the methods to obtain color descriptions have been implemented, it has been decided to evaluate them. To achieve it, it has been decided to perform different tests in order to know if the pictures selected by people taking into account the color are the same than the results obtained from the color descriptors. This work is reflected in Chapter 6.

Finally, in Chapter 7 a web page has been developed and demonstrated that it would be possible to carry out an automatic arrangement of a set of photos using its colors as a reference.

CHAPTER 1. INTRODUCTION TO MPEG-7

This chapter introduces the basic knowledge on how to be able to find similarities between pictures which is the basis for ordering in a specific sequence. To characterize the pictures, some specific measures need to be calculated. These measures are known as descriptors.

Although there are many types of descriptors, in this project it has been decided to use only the MPEG-7 ones, because they are an international standard that guarantees interoperability between several applications. Moreover, these descriptors have already been checked out in an efficient way in other applications.

First, a brief description of MPEG and its different parts is given. Afterwards, it is explained the MPEG-7 standard and it is shown a global vision of its objectives as well as some real applications. Finally, a summary of the different visual descriptors included into MPEG-7 is presented.

1.1 MPEG

The Moving Picture Experts Group (MPEG) [2] is a working group of ISO/IEC in charge of the development of international standards for compression, decompression, processing, and coded representation of moving pictures, audio and their combination. It has created the following standards:

- MPEG-1: The first compression standard for audio and video. It was basically designed to allow moving pictures and sound to be encoded into the bitrate of a Compact Disc. It includes the popular Layer 3 (MP3) audio compression format.
- MPEG-2: Transport, video and audio standards for broadcast-quality television. It is commonly used in digital satellite TV, digital cable television signals or DVD.
- MPEG-4: The standard for multimedia applications. It is a collection of standards oriented to different kind of applications (videoconference, TV, video streaming …) Uses further coding tools with additional complexity to achieve higher compression factors than MPEG-2.
- MPEG-7: The content representation standard for multimedia information search, filtering, management and processing
- MPEG-21: The multimedia framework oriented to management and distribution of audiovisual contents.

Therefore, in a world where information technology, consumer electronics and telecommunication products converge by incorporating increasingly sophisticated technologies and the need for timely available standards is as strong as ever, MPEG provides a proven mechanism to feed research results into standards that promote industry innovation to the benefit of all.

One of these technologies is MPEG-7, which provides an infrastructure to describe contents using key words and semantic (who, what, when, where) or structural (colors, textures, movements) content.

1.2 Description of MPEG-7

There are different references where a detailed revision of the MPEG-7 standard can be found [3] [4] [5] [6]. In this section it is only intended to give a general vision of the standard. Due to MPEG-7 belongs to a standard we have taken the liberty of copying the definitions of the objectives on a direct quotation as the accuracy of the language is very important.

‘Multimedia search and retrieval has become a very active research field due to the increasing amount of audiovisual data that is becoming available, and the growing difficulty to search, filter or manage it. And the trend is clear: in the next few years, users will be confronted with a large number of contents provided by multiple sources.

Therefore, there is a need for the audiovisual content to:

- Manage the information
- Find, select and filter the contents
- Use the contents through computational systems

This context has led to the development of efficient processing tools that are able to create the description of audiovisual material or to support the identification or retrieval of audiovisual documents. MPEG-7, also called “Multimedia Content Description Interface”, standardizes the description of multimedia content supporting a wide range of applications.’ [6]

The MPEG-7 is organized in 8 parts. Part 1 defines the tools needed to prepare MPEG-7 descriptions and part 2 defines its syntax. Parts 3, 4 and 5 define the elements which represent the description of the contents, visual, audio or multimedia respectively, from low to high levels of abstraction. A reference software implementation and guidelines for testing conformance are defined in parts 6 and 7, and part 8 defines the extraction and the use of the descriptions. In the following reference a complete description of the MPEG-7 can be found [7].

In spite of all of these parts, in this project we are only going to consider some of the color descriptors of the image, included in part 3.

Once it has been explained how MPEG-7 is structured and its main parts, it is necessary to go deeply into its objectives and also explain some of the applications where MPEG-7 can be applied.

1.2.1 MPEG-7 Objectives

‘Audiovisual sources will play an increasingly pervasive role in our lives, and there will be a growing need to have these sources processed further. This makes necessary to develop forms of audiovisual information representation that go beyond the simple waveform or sample-based, compression-based (such as MPEG-1 and MPEG-2) or even objects-based (such as MPEG-4) representations. Forms of representation that allow some degree of interpretation of the information meaning are necessary.

Because the descriptive features must be meaningful in the context of the application, they will be different for different user domains and different applications. This implies that the same material can be described using different types of features, tuned to the area of application. To take an example of visual material: a lower abstraction level would be a description of e.g. shape, size, texture, color, movement (trajectory) and position (where in the scene can the object be found?); and for audio: key, mood, tempo, tempo changes, position in sound space. The highest level would give semantic information: ‘This is a scene with a barking brown dog on the left and a blue ball that falls down on the right, with the sound of passing cars in the background. Intermediate levels of abstraction may also exist.’ [6]

The level of abstraction is related to the way the features can be extracted: many low-level features can be extracted in fully automatic ways, whereas high level features need (much) more human interaction.

However, MPEG-7 descriptions do not depend on the ways the described content is coded or stored. It is possible to create an MPEG-7 description of an analogue movie or of a picture that is printed on paper, in the same way as of digitized content.

Therefore, the main objectives of MPEG-7 are [3]:

- Provide a fast and efficient searching, filtering and content identification method.
- Describe main issues about the content (low-level characteristics, structure, models, collections, etc.).
- Index a big range of applications.
- Audiovisual information that MPEG-7 deals is: Audio, voice, video, images, graphs and 3D models.
- Inform about how objects are combined in a scene.
- Independence between description and the information itself.

1.2.2 MPEG-7 Applications

Taking into account the figure **Fig. 1.1** which explains a hypothetical MPEG-7 chain in practice, an audiovisual description from the multimedia content is obtained via manual or semi-automatic extraction. The audiovisual description may be stored or streamed directly. In the MPEG-7 context is possible to define two scenarios related to the use of this content: the pull or push scenario.

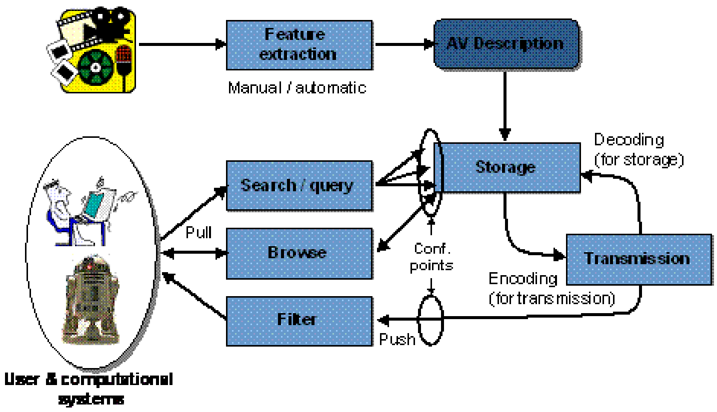

Fig. 1.1 Possible applications using MPEG-7 [3]

On the one hand, if we consider a pull scenario, client applications will submit queries to the descriptions repository and will receive a set of descriptions matching the query for browsing (just for inspecting the description, for manipulating it, for retrieving the described content...).

A few examples of this kind of applications are:

- Play a few notes on a keyboard and retrieve a list of musical pieces similar to the required tune.
- Draw a circle on a screen and find a set of images containing similar shapes, logos, ideograms...
- Select a color and retrieve images where this color also appears.
- Use an excerpt of a U2 song and obtain a list of U2 records, video clips and photographic material.

On the other hand, in a push scenario a filter will select descriptions from the available ones and perform the programmed actions afterwards (such as switching a broadcast channel or recording the described stream).

Some examples of this kind of applications are:

- Filter for receiving only cartoon content on a multimedia stream or IPTV system
- Turn on an alarm when a certain event happens on an audiovisual stream (goals, music...).

In both scenarios, pull or push, all the modules may handle descriptions coded in MPEG-7 formats (either textual or binary) in order to make easier the management of audiovisual content.

1.3 Visual descriptors

The main objective of the MPEG-7 visual standard is to provide standardized descriptions of streamed or stored images or video-standardized header bits (visual low-level descriptors) that help users or applications to identify, categorize or filter images or video. These low-level descriptors can be used to compare, filter or browse images or video purely on the basis of non-textual visual descriptions of the content – or in combination with common text-based queries. They will be used differently for different user domains and different applications environments.

Among this diversity of possible applications the MPEG-7 visual feature descriptors allow users to perform different tasks, for instance it could be possible to draw a square and obtain a set of images containing objects or graphics similar to it; or to select some color structures and obtain images with these color relations. Moreover, it is possible to describe actions, for example “football matches” and get a list of videos in which football actions happen.

The MPEG-7 Visual Descriptors describe basic audiovisual media content on the basis of visual information. For images and video, the content may be described, for example by the shape of objects, object size, texture, color, movement of objects and camera motion.

Once the MPEG-7 descriptors are available, suitable search engines can be employed to search, filter or browse visual material on the basis of suitable similarity measures.

The MPEG-7 descriptors can be broadly classified into general visual descriptors and domain-specific visual descriptors. The former include color, texture, shape, localization and motion features, while the latter are application-dependent and include a face-recognition descriptor. A brief description of the general visual descriptor classes is given below.

- **Color descriptors**: Color is one of the most widely used visual features in image and video retrieval. Color features are relatively robust to the viewing angle, translation and rotation of the regions of interest. The MPEG-7 standard defines five color descriptors which represent different aspects of the color feature, including color distribution, spatial layout of color and spatial structure of color.

- **Texture descriptors**: Texture refers to the visual patterns that have properties of homogeneity or not, that result from the presence of multiple colors or intensities in the image. It is a visual property of any surface, including clouds, trees, bricks, hair and fabric. It contains important structural information of surfaces and their relationship to the surrounding environment.

- **Shape descriptors**: In many image database applications, the shape of the image objects provides a powerful visual clue for similarity matching. MPEG-7 provides region and contour descriptors suitable for a variety of applications as well as a 3-D shape descriptor that is useful for invariant to geometric transformations recognition of object shapes.

- **Motion descriptors**: Description of motion features in video sequences can provide powerful clues regarding its content. MPEG-7 has developed descriptors that capture essential motion characteristics into concise and effective descriptions.

- **Localization descriptors**: The location descriptor enables localization of regions within images or frames as well as to describe spatio-temporal regions in a video sequence, such as moving object regions. Also provides localization functionality.

- **Face descriptor**: Human face perception is an active area in the computer vision community. There are many applications in which automatic face recognition is desirable and different specific techniques were proposed. Among these methods, Principal Component Analysis (PCA) is a popular technique that is widely used and the MPEG-7 face descriptor is based on that.

CHAPTER 2. COLOR DESCRIPTORS

Color is an important visual attribute for both human vision and computer processing. Therefore, a possible way to order elements could be according to its color.

In the previous chapter, the basic concepts of the MPEG-7 standard were explained, while in this chapter some details about the MPEG-7 color descriptors will be presented. They have to allow us to characterize elements according to its color to order them afterwards.

Firstly, the concept of color descriptor and the factors which influence on its selection will be introduced. Secondly, the color spaces specified in the MPEG-7 will be defined, as they are the mathematical model to represent colors. Finally, the different descriptors of the MPEG-7 will be presented.

2.1 Introduction

Color is the most basic quality of the visual contents, therefore it is possible to use colors to describe and represent an image. In the literature there are many possible procedures to describe the color. The MPEG-7 standard has tested the most efficient ones and has selected those that have provided more satisfactory results.

Also, various factors influenced the selection of these color descriptors, such as their ability to characterize the perceptual color similarity or the size of the coded descriptions. It is important to take into account the low complexity of the associated extraction and matching techniques and the scalability and interoperability.

MPEG-7 standard purposes different methods to obtain these descriptors. Particularly, five tools are defined to describe color:

- Dominant Color Descriptor (DCD)
- Scalable Color Descriptor (SCD)
- Group of frame (GoF) or Group-of-pictures (GoP)
- Color Structure Descriptor (CSD)
- Color Layout Descriptor (CLD)

The DCD, SCD and CSD represent the color distribution in an image and the GoF, GoP or CLD ones describe the color relation between sequences or group of images.

Nevertheless, we have to point out that the standard does not define how the extraction of the descriptors and their use in similarity matching must be performed. For each of the descriptors, the MPEG-7 standard provides details of its syntax, descriptor computation and experimental results.

Therefore, an accurately study of each descriptor is needed in order to be able to determine the application to give to each one. In next sections is going to be explained the color spaces used in MPEG-7 standard as well as the different color descriptors.

2.2 Color Spaces

A color space is a mathematical model describing the way colors can be represented. The color spaces used in MPEG-7 are RGB, YCbCr, HSV, HMMD, Monochrome and linear transformation matrix with reference to RGB. The standard describes these color spaces but for each descriptor there is a recommendation of which is the most efficient color space to use. In this section, their main characteristics are briefly explained.

- **RGB:** The RGB color space is one of the most widely used color models. The name of the model comes from the initials of the three additive primary colors: Red, Green, and Blue. It is defined as the unit cube in the Cartesian coordinate system as can be seen in the following figure.

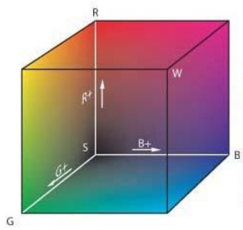

Fig. 2.1 Unit cube in the Cartesian coordinate system [8]

  To indicate in which proportion it is mixed each color, a value is assigned to each of the primary colors where 0 value means that it does not intervene in the mixture. It is quite common to code each primary color with one byte.

  The images used during this project were defined on the RGB color space.

- **YCbCr:** The YCbCr is a legacy color space of the precedent analogue color television and MPEG standards. It is defined by a linear transformation of RGB color space:

Luminance	Y = 0.299*R + 0.587*G + 0.114*B
Blue chrominance	Cb = 0.169*R – 0.331*G + 0.500*B
Red chrominance	Cr = 0.500*R – 0.419*G – 0.081*B

The difference between YCbCr and RGB is that YCbCr represents color as brightness and two color difference signals, while RGB represents color as red, green and blue.

For the Monochrome color representation, Y component alone in the YCrCb is used.

The MPEG-7 standard recommends using this color space for the Color Layout Descriptor.

- **HSV:** The HSV color space attempts to describe perceptual color relationships more accurately than RGB, while remaining computationally simple. It is defined as a cylinder (**Fig. 2.2**) consists of Hue (H), Saturation (S) and Value (V). Hue is represented by the angle from 0 to 360º and specifies one color family from another. Saturation (= [0,1]) specifies how pure a color is and Value (= [0,1]) specifies how bright or dark a color is.

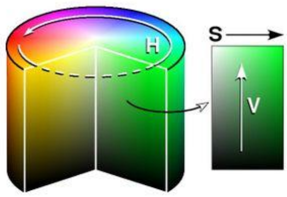

Fig. 2.2 HSV color space [9]

Each component (Hue, Saturation and Value) is defined as:

- $Max = \max\,(R, G, B)$
- $Min = \min\,(R, G, B)$
- $$Hue = \begin{cases} 0, & if \quad Max == Min \\ 60 * \frac{G-B}{Max-Min}, & if \quad Max == R \text{ and } G \geq B \\ 360 + 60 * \frac{G-B}{Max-Min}, & if \quad Max == R \text{ and } G < B \\ 60 * \left(2.0 + \frac{B-R}{Max-Min}\right), & if \quad G == Max \\ 60 * \left(4.0 + \frac{R-G}{Max-Min}\right), & otherwise \end{cases}$$
- $$Saturation = \begin{cases} 0\,, & if \quad Max == 0 \\ \frac{Max-Min}{Max}, & otherwise \end{cases}$$
- $Value = Max$

The differences between RGB and HSV are in the way the model emulate how humans perceive color.

The Group of Frames or Group of Pictures Descriptor and the Scalable Color Descriptor use this color space, as a MPEG-7 standard recommendation.

- **HMMD**: The HMMD (Hue-Max-Min-Diff) color space is closer to a perceptually uniform color space. In the MPEG-7 core experiments had been observed that it was very effective for image retrieval. The double cone shape confines this color space as shown in **Fig. 2.3**.

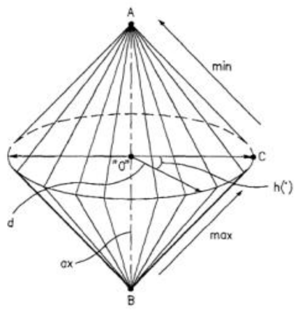

Fig. 2.3 HMMD color space [10]

A total of five components are defined:

- Hue
  - Has the same property as Hue in the HSV color space
- Max = max (R,G,B)
  - Specifies how much black color is present
- Min = min (R,G,B)
  - Specifies how much white color is present
- Diff = Max – Min
  - Specifies how much a color is close to pure colors
- Sum = (Max + Min)/2
  - Specifies the brightness of the color

However, a set of three components {H,Max,Min} or {H,Diff,Sum} is sufficient to form the HMMD color space.

The standard recommends the use of this color space for the Color Structure Descriptor implementation.

2.3 Color Descriptors

In this section the different color descriptors described in MPEG-7 standard are summarized. We will get later into further details for the Dominant Color Descriptor (DCD) and the Color Layout Descriptor (CLD) as they will be selected as the color descriptors implemented in this project.

- **Dominant Color Descriptor**: The Dominant Color Descriptor (DCD) provides a compact description of the representative colors in an image or image region. It allows a specification of a small number of dominant color values as well as their statistical properties such as distribution and variance. Its purpose is to provide an effective, compact and intuitive representation of colors present in a region or image.

- **Scalable Color Descriptor**: The Scalable Color Descriptor is a color Histogram in the HSV Color Space, which is encoded by a Haar transform. It is useful for image-to-image matching and retrieval based on color feature.

- **Group of Frames or Group of Pictures Descriptor**: The Group of Frames or Group of Pictures Descriptor is an extension of the Scalable Color Descriptor to a group of frames in a video or a collection of pictures. This descriptor is based on aggregating the color properties of the individual images or video frames.

- **Color Structure Descriptor**: The Color Structure Descriptor is also based on color histograms, but aims at identifying localized color distribution using a small structuring window. To guarantee interoperability, the Color Structure Descriptor is bound to the HMMD color space.

- **Color Layout Descriptor**: The Color Layout Descriptor captures the spatial layout of the representative colors on a grid superimposed on a region or image. Representation is based on coefficients of the Discrete Cosine Transform. This is a very compact descriptor being highly efficient in fast browsing and search applications. It can be applied to still images as well as to video segments.

After studying the advantages and disadvantages of each color descriptor, the Color Layout Descriptor (CLD) and the Dominant Color Descriptor (DCD) were selected. This decision was carried out to take the advantages that DCD and CLD offers. The DCD lets to describe effectively the dominant colors of an image and the CLD is a compact, resolution-invariant representation that retains the spatial distribution of the color of an image. These characteristics were decisive when the selection of the descriptors was taken.

CHAPTER 3. COLOR LAYOUT DESCRIPTOR (CLD)

One of the main objectives of this project is to be able to characterize images in order to determine their similarity in terms of color perception. This similarity measure will be used to create perceptual pleasant sequence of images, which is defining a presentation order according to certain criteria. As has been seen in previous sections, this characterization could be done with color descriptors. In Chapter 2 the MPEG-7 standardized color descriptors have been described but during the execution of this project only the CLD and the DCD were selected.

In this chapter the Color Layout Descriptor (CLD) will be explained in detail. First of all, its definition is presented and then its extraction procedure is explained. Later, its results will be presented as well as some improvements.

3.1 Definition

The CLD is a very compact and resolution-invariant representation of color for high-speed image retrieval and it has been designed to efficiently represent the spatial distribution of colors. This feature can be used for a wide variety of similarity-based retrieval, content filtering and visualization. It is especially useful for spatial structure-based retrieval applications.

This descriptor is obtained by applying the discrete cosine transform (DCT) transformation on a 2-D array of local representative colors in Y or Cb or Cr color space.

The functionalities of the CLD are basically the matching:

- Image-to-image matching
- Video clip-to-video clip matching

Finally, remark that the CLD is one of the most precise and fast color descriptor.

3.2 Extraction

The extraction process of this color descriptor consists of four stages: image partitioning, representative color detection, DCT transformation and a zigzag scanning. **Fig. 3.1** illustrates this extraction process. Moreover, as the images used during the realization of this project were defined on the RGB color space, a stage of color space conversion was added, as the standard MPEG-7 recommends to use the YCbCr color space for the CLD.

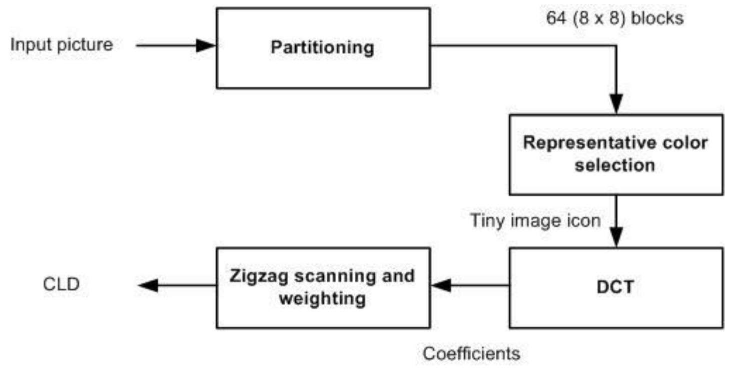

Fig. 3.1 Extraction process of the CLD

In the image partitioning stage, the input picture (on RGB color space) is divided into 64 blocks to guarantee the invariance to resolution or scale.

Fig. 3.2 Step 1: Partitioning

The inputs and outputs of this step are summarized in the following table:

Table 3.1 Inputs and Outputs Stage 1

Input Stage 1	Output Stage 1
Input picture [M x N]	Input picture divided into 64 blocks [M/8xN/8]

After the image partitioning stage, a single representative color is selected from each block. Any method to select the representative color can be applied, but the standard recommends the use of the average of the pixel colors in a block as the corresponding representative color, since it is simpler and the description accuracy is sufficient in general.

The selection results in a tiny image icon of size 8x8. Next figure shows this process. Underline that in the image of the figure, it has been maintained the resolution of the original image only in order to facilitate its representation.

Fig. 3.3 Stage 2: Representative color selection

The inputs and outputs of this stage are summarized in the next table:

Table 3.2 Inputs and Outputs Stage 2

Input Stage 2	Output Stage 2
Input picture divided into 64 blocks [M/8xN/8]	Tiny image icon [8x8]

Once the tiny image icon is obtained, the color space conversion between RGB and YCbCr is applied. This color space conversion can be done in any stage though the MPEG-7 standard recommended to proceed with the conversion in this point in order to reduce the computational load that this process involves. This conversion, as has been explained in section **2.2**, it is defined by a linear transformation of the RGB color space:

Luminance	Y = 0.299*R + 0.587*G + 0.114*B - 128
Blue chrominance	Cb = 0.169*R – 0.331*G + 0.500*B
Red chrominance	Cr = 0.500*R – 0.419*G – 0.081*B

The inputs and outputs of this stage are summarized in the following table:

Table 3.3 Inputs and Outputs Stage 3

Input Stage 3	Output Stage 3
Tiny image icon [8x8] in RGB color space	Tiny image icon [8x8] in YCbCr color space

In the fourth stage, the luminance (Y) and the blue and red chrominance (Cb and Cr) are transformed by 8x8 DCT, so three sets of 64 DCT coefficients are obtained.

To calculate the DCT in a 2D array, the following formulae are used:

$$B_{pq} = \propto_p \propto_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} A_{m\,n} \cos\frac{\pi\,(2m+1)p}{2M} \cos\frac{\pi\,(2n+1)q}{2N}, \qquad \begin{matrix} 0 \le p \le M-1 \\ 0 \le q \le N-1 \end{matrix} \tag{3.1}$$

$$\propto_p = \begin{cases} \frac{1}{\sqrt{M}}, & p = 0 \\ \sqrt{\frac{2}{M}}, & 1 \le p \le M-1 \end{cases} \qquad \propto_p = \begin{cases} \frac{1}{\sqrt{N}}, & q = 0 \\ \sqrt{\frac{2}{N}}, & 1 \le q \le N-1 \end{cases}$$

The inputs and outputs of this stage are summarized in the next table:

Table 3.4 Inputs and Outputs Stage 4

Input Stage 4	Output Stage 4
Tiny image icon [8x8] in YCbCr color space	3 [8x8] matrix of 64 coefficients (DCTY, DCTCb, DCTCr)

A zigzag scanning is performed with these three sets of 64 DCT coefficients, following the schema presented in the next figure:

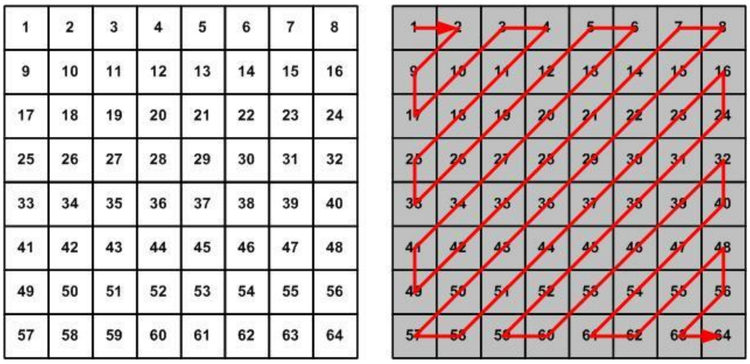

Fig. 3.4 Stage 5: Zigzag scanning

The purpose of the Zig-zag Scan is to group the low frequency coefficients of the 8x8 matrix.

The inputs and outputs of this stage are summarized in the next table:

Table 3.5 Inputs and Outputs Stage 5

Input Stage 5	Output Stage 5
3 matrix of 64 coefficients (DCTY, DCTCb, DCTCr)	3 zigzag scanned matrix (DY, DCb, DCr)

Finally, these three set of matrices correspond to the CLD of the input image.

It should be noted that this descriptor is one of the most compact descriptors in the MPEG-7 visual tools and is quite suitable for applications having limitations on storage and/or bandwidth.

3.3 Matching

The matching process helps to evaluate if two elements are equal comparing both elements and calculating the distance between them. In the case of color descriptors the matching process helps to evaluate if two images are similar. Its procedure is the following:

- Given an image as an input, the application attempts to find an image with a similar descriptor in a data base of images.

If we consider two CLDs:

{DY, DCb, DCr}
{ DY', DCb', DCr' },

The distance between the two descriptors can be computed as:

$$D = \sqrt{\sum_i w_{yi} \left(DY_i - DY_i{}'\right)^2} + \sqrt{\sum_i w_{bi} \left(DCb_i - DCb_i{}'\right)^2} + \sqrt{\sum_i w_{ri} \left(DCr_i - DCr_i{}'\right)^2} \qquad \textbf{(3.2)}$$

The subscript *i* represents the zigzag-scanning order of the coefficients. Furthermore, notice that is possible to weight the coefficients (*w*) in order to adjust the performance of the matching process. These weights let us give to some components of the descriptor more importance than others. In next sections different weights are tested.

Observing the formula, it can be extracted that:

- 2 images are the same if the distance is 0
- 2 images are similar if the distance is near to 0

Therefore, this matching process will let to identify images with similar color descriptors. Since the complexity of the similarity matching process shown above is low, high-speed image matching can be achieved.

3.4 *Implementation*

Once the CLD has been described and explained how it can be used to carry out the matching between images, it is possible to make an implementation to adjust this descriptor to the objectives of this project.

We aim to find images with similar colors, thus, we have to extract the CLD from these images and afterwards compare these descriptors with the matching technique. Consequently, is possible to define two main parts in the implementation of this method:

- Process a database of pictures to obtain its CLD
- Find similarity matching between an input picture and the processed database

Emphasize that a database of pictures in this project means a collection of pictures of different resolution and sizes.

The following figure shows the process of analyzing a database:

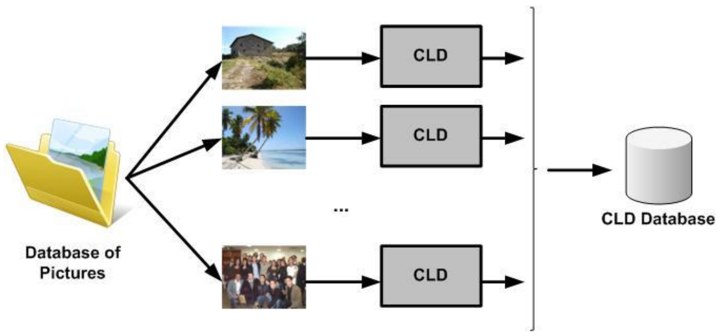

Fig. 3.5 CLD: Process a database

In this process, a database of pictures is analyzed in order to obtain the CLD representing each picture. This process consists of uploading the image into memory and computing the descriptor as explained in the previous section. The final result is a database of CLDs linked to the images that represent.

The following table shows the inputs and outputs of this part of the implementation:

Table 3.6 Inputs and outputs processing a database for CLD

Input	Output
Path of a Database with N pictures	CLD Database {image name$_i$, DY$_i$, DCb$_i$, DCr$_i$},i = 1..N

Once the database of images has been analyzed, the matching between an input image and the database of CLD is carried out. With this process, it will be obtained images with similar colors ordered according to increasing distances. The process of matching and finding similarities is summarized in **Fig. 3.6**.

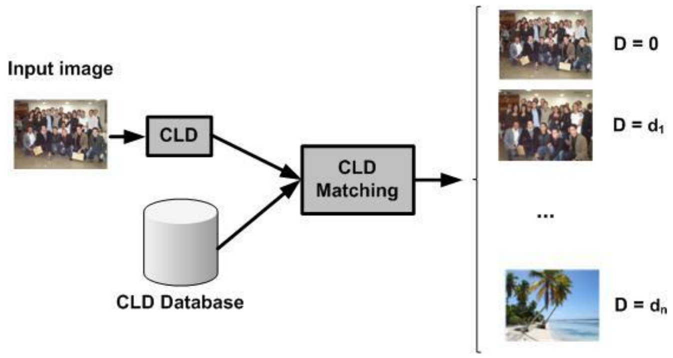

Fig. 3.6 CLD: Find similarity matching

In this case, the input parameters are the input image and the database of the CLD obtained in the previous step. Once the matching process is finished, the images of the database are ordered in increasing distances, where D = 0 means the same color descriptor and $d_1 < d_n$.

Table 3.7 Inputs and Outputs find similarity matching for CLD

Input	Output
Input image and CLD database	Images ordered according to increasing distances

To obtain this implementation, during the execution of this project it has been used the MATLAB environment as it offers the possibility to work in a simple way with images. With MATLAB it is possible to carry out a good implementation of the CLD for later adjusts its performance in order to obtain the desired results. To be precise, the MATLAB version used during this implementation is MATLAB *R2007b 7.5.0.342*.

In addition, the hardware used is a laptop with the following characteristics:

- Laptop: ASUSTeK Computer Inc. A6JM
- OS: Microsoft Windows XP Home Edition
- Processor: Genuine Intel T2400 @ 1.83GHz, 1.83GHz
- RAM: 2,00 GB

3.5 Results

The results obtained with the implementation of the CLD are shown in this section. The tests are carried out with a database of 100 JPG images. The resolutions of the images are different, between a maximum of 1536x1152 pixels and a minimum of 512x384 as they are the ones provided by different commercial cameras. The photos have an average size of 2 MB and there are different scenes (a scene means a set of photos of the same place, such as beach, mountain, a friends meeting...). These images can be consulted in **Annex A**.

As it has been commented previously, the main objective of this project is to be able to group, in an automatic way, images with colors or color structures very similar. In this way, due to the fact that CLD is designed to efficiently represent spatial distribution of colors, it will allow us to detect similar images into a database that will correspond to the same scene since they probably have very similar colors.

To be able to observe this characteristic, the first step is to analyze the database of images. Once this step has been finished and all the CLDs have been obtained, one of the images of the database has been chosen in order to search the similarities among it and the rest of images.

During the realization of this project different tests have been carried out with different photos of the data base. For each one its results were analyzed and the distances obtained were also studied. In **Fig. 3.7** one example can be observed. The results are presented in a MATLAB figure, where the top image is the input photo and the first ten pictures are ordered in increasing distances from left to right and from top to down. Moreover, it is shown the name and the distance for each picture.

Fig. 3.7 Matching results

In this particular example, as on the data base there was a total of six images of the scene of a group of persons, what was tried to obtain was to group the rest of images using the CLD, due to they have very similar colors.

As image with less distance (distance 0) was obtained the input image (it was detected on the database) and inside the 10 first images with less distance all the images of the scene of the group of people were also included. Nevertheless, it can be observed that for this particular example, it has not been managed to group all the images inside the top six.

The obtained results were successful although for these tests it was considered the same weight for all the components of the CLD when the matching process was performed (w equal to 1). Because of this it will be necessary to adjust these parameters to improve these results.

Finally, it is also important to analyze the computing time in order to obtain the descriptors and the results:

- Analyzing the CLDs of the database of 100 pictures: **12 -13 seconds**
- Input image CLD extraction and Matching with 100 CLD: **2 seconds**

This average time has been obtained repeating several times each test. Taking into account the hardware used, the computing time is low.

3.6 Improvements

As it has been seen in the previous section, it is possible to adjust the weights during the matching process in order to improve the results. To adjust these weights, it was decided to do the same tests executed on the Ramon Alujas Tejada PFC [11]. In that project, an exhaustive study of the different weights given to each component of the CLD was carried out. With his results, he managed to make the CLD descriptor robust against brightness and contrast changes on the image. Consequently, in this project the same tests have been repeated and we have used its results.

In order to adjust the weights of the matching process each component of the CLD (three 8x8 matrix, DY, DCb, DCr after the zigzag scanning) could be weighted (***w***) appropriately.

$$D = \sqrt{\sum_i \boldsymbol{w_{yi}}\,(DY_i - DY_i')^2} + \sqrt{\sum_i \boldsymbol{w_{bi}}\,(DCb_i - DCb_i')^2} + \sqrt{\sum_i \boldsymbol{w_{ri}}\,(DCr_i - DCr_i')^2} \qquad \textbf{(3.3)}$$

So, after studying and repeating the different tests realized in the Ramon Alujas Tejada PFC, the following weights (***w***) have been assigned to each coefficient:

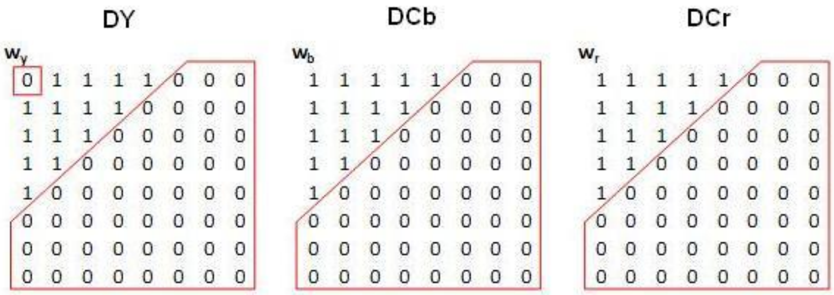

Fig. 3.8 Matching weights

Notice that a zero weight is assigned to the DC coefficient of the DY component in order to reduce the effect of a brightness change in the image. Likewise, it is given also a zero weight to the high frequency coefficients of the DY, DCb and DCr components since the information that they bring compared with the low frequency coefficients have a low effect obtaining the descriptor.

With this configuration it can be observed an improved detection of similar images which have different brightness and contrasts. For instance, **Fig. 3.9** is one sample of these results. The same input image as in the section **3.5** has been selected.

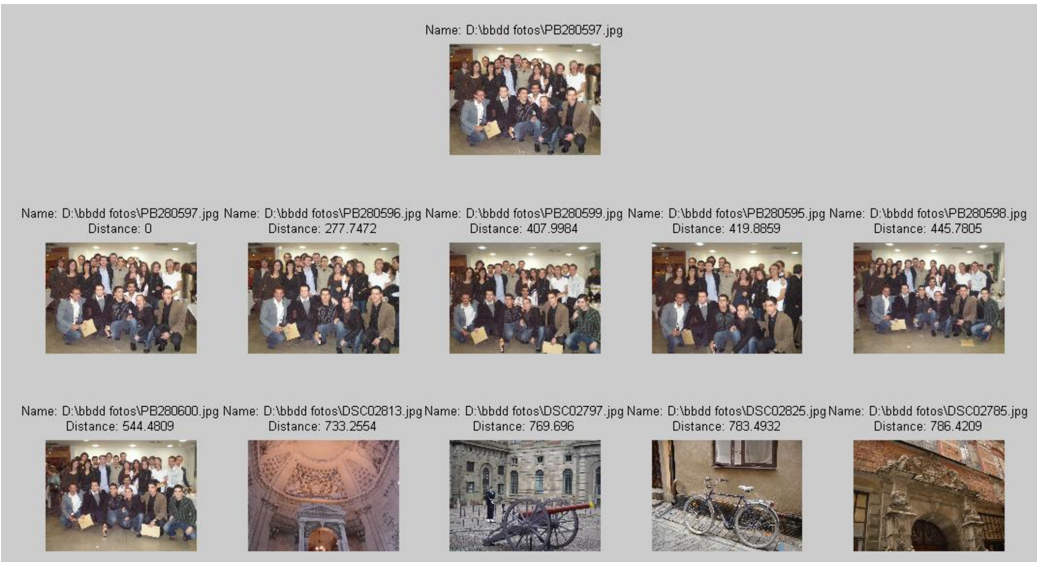

Fig. 3.9 Improved Matching

For this particular case, it can be observed that with the selected weights the six images of the same scene now appear on the top. The same result has been observed in different tests and we have noticed that with these weights the robustness of the CLD against brightness and contrast changes of the images has been improved, as Ramon Alujas Tejada concluded in his PFC.

CHAPTER 4. DOMINANT COLOR DESCRIPTOR (DCD)

MPEG-7 standard proposes different methods to obtain different color descriptors from an image. As has been seen in previous chapters, for the execution of this project the CLD and the DCD have been selected as the color descriptors that we are going to use. In the previous chapter, the CLD has been studied and in this chapter the same will be done with the DCD. The Dominant Color Descriptor is a compact descriptor and will let us to efficiently represent the dominant colors on an image.

In this chapter in particular, it will be explained the DCD in detail. First of all, it will be defined and then its extraction procedure will be explained. Later, its results will be presented as well as some proposals for improvements in our particular application.

4.1 Definition

The DCD provides a compact description of the representative colors in an image or image region.

The main target applications of the DCD are similarity retrieval in image databases and browsing of image databases based on single or several color values.

The DCD is defined to be:

$$F = \{(\boldsymbol{c}_i, p_i, v_i), s\}, \qquad (i = 1, 2, \dots, N) \tag{4.1}$$

where N is the number of dominant colors. Each dominant color value c_i, also called centroid, is a vector of the corresponding color space component values (such as a 3-D vector in the RGB color space). The percentage p_i (normalized to a value between 0 and 1) is the fraction of pixels in the image or image region corresponding to color c_i, and $\sum_i \mathrm{p_i} = 1$.

The optional values are the color variance v_i, which describes the variation of the color values of the pixels in a cluster around the corresponding representative color, and the spatial coherency s, which is a single number that represents the overall spatial homogeneity of the dominant colors in the image.

Even so, in this project the optional values will not be taken into account and only the dominant colors and its percentages will be used.

4.2 Extraction

The extraction procedure of the dominant colors from an image uses a clustering method to cluster the pixel color values. **Fig. 4.1** illustrates the main steps in order to obtain the DCD from an input image:

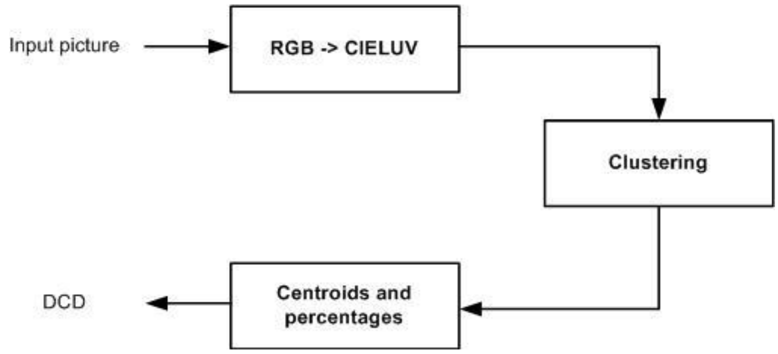

Fig. 4.1 Extraction process of the DCD

The extraction process consists of three stages: a color space conversion, a clustering method and a calculation of the percentages of each centroid.

The first stage is a color space conversion recommended by the MPEG-7 standard, in order to perform the clustering method in a perceptually uniform color space such as the CIE LUV. Due to the images used during this project were defined on the RGB color space it was necessary to perform this color space conversion.

The CIE 1976 (L*, u*, v*) color space, also known as CIE LUV, is a color space adopted by the International Commission on Illumination (CIE) in 1976. It is extensively used for applications such as computer graphics which deal with colored lights.

Its representation is described in **Fig. 4.2**.

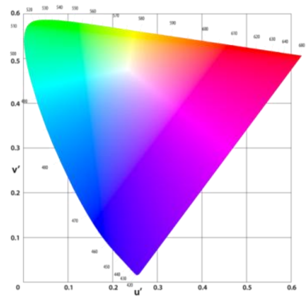

Fig. 4.2 CIELUV color representation [12]

Firstly, the RGB values are converted to the CIE XYZ color space.

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 3.240479 & -1.537150 & -0.498535 \\ -0.969256 & 1.875992 & 0.041556 \\ 0.055648 & -0.204043 & 1.057311 \end{bmatrix} * \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \tag{4.2}$$

When this process has finished, the conversion to the CIE LUV color space is performed.

$$\begin{aligned} L^* &= 116(Y/Y_n)^{1/3} - 16 \\ u^* &= 13L^*(u' - u'_n) \\ v^* &= 13L^*(v' - v'_n) \end{aligned} \tag{4.3}$$

The quantities u'_n and v'_n refer to the reference white point or the light source. In this project the values of $u'_n = 0.1978$ and $v'_n = 0.4683$ were used. Equations for u' and v' are given below:

$$\begin{aligned} u' &= \frac{4X}{X + 15Y + 3Z} = \frac{4x}{-2x + 12y + 3} \\ v' &= \frac{9Y}{X + 15Y + 3Z} = \frac{9y}{-2x + 12y + 3} \end{aligned} \tag{4.4}$$

After the color space conversion, the inputs and outputs of this stage are summarized in the following table:

Table 4.1 Inputs and Outputs Stage 1

Input Stage 1	**Output Stage 1**
Input picture [M x N] in RGB color space	Input picture [M x N] in CIE LUV color space

Once the input image is on the CIE LUV color space, a clustering algorithm is applied in order to extract the dominant colors of the image.

Any clustering method can be applied, however the one selected in this project is the K-means algorithm as it is a method highly tested in the MATLAB environment and also the results obtained using it were satisfactory.

K-means clustering is a method of cluster analysis which aims to divide *n* observations into *k* clusters in which each observation belongs to the cluster with the nearest mean.

One clustering example can be seen on **Fig. 4.3**. Starting from *n* elements, the creation of two clusters is expected. By applying K-means, two clusters are obtained, Cluster 1 in red and Cluster 2 in blue. Also the corresponding centroids are calculated.

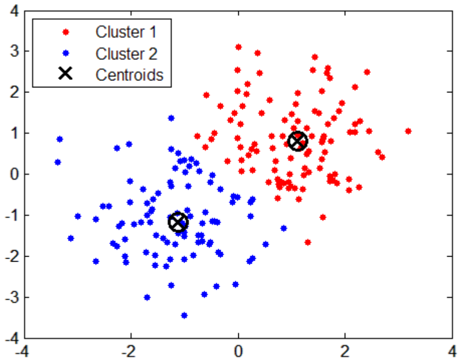

Fig. 4.3 K-means clustering example

In the case of images, with K-means we will be able to group the *n* pixels of the image into *k* clusters in order to obtain pixels with similar colors.

This algorithm follows a sequence of centroid calculations and clustering steps until the number of desired clusters is met. In [13] [14] it can be obtained more detailed information about the K-means algorithm.

As a result, the following figure is an example of a color clustering using K-means for the DCD extraction of an input image. As the standard recommends the use of three or four dominant colors for the DCD, in this particular example four colors have been selected. Underline that in the image of the figure it has been maintained the RGB color space when the K-means algorithm is applied in order to facilitate its comprehension:

Fig. 4.4 K-means: RGB clustering with 4 clusters

It can be observed as only four dominant colors of the original image have been selected, which corresponds to the four centroids of the K-means clusters.

When the clustering process is finished, the percentage of pixels in the image belonging to each of the clusters is calculated. Taking into account the previous example, after processing the image, the DCD obtained (four centroids and each percentage) are summarized in the following table:

Table 4.2 Example of DCD

Centroids (c_i)			Percentage (p_i)
L*	u*	v*	
54.2495	70.0140	12.3690	0.3501 %
80.3203	24.7781	18.8156	0.1321 %
66.4478	49.2724	17.7287	0.2456 %
34.6152	36.4142	-15.7432	0.2722 %

These centroids and its corresponding percentages are the DCD of the image. The inputs and outputs of this step are summarized in the **Table 4.3**:

Table 4.3 Inputs and Outputs Stage 2

Input Stage 2	Output Stage 2
Input picture [M x N] in CIE LUV color space	$F = \{(\boldsymbol{c}_i, p_i)\}, \quad (i = 1, 2, \dots, K)$

4.3 Matching

As it has been said in section **3.3**, the similarity matching is a tool that helps to evaluate if two elements are equal. Therefore compares both elements and calculates the distance between them.

If we consider two DCDs:

$$F_1 = \{(\boldsymbol{c}_{1i}, p_{1i})\}, \quad (i = 1, 2, \dots, N_1)$$
$$F_2 = \{(\boldsymbol{c}_{2i}, p_{2i})\}, \quad (i = 1, 2, \dots, N_2)$$

The distance $D^2(F_1, F_2)$ between the two descriptors can be computed as:

$$D^2(F_1, F_2) = \sum_{i=1}^{N_1} p_{1i}^2 + \sum_{j=1}^{N_2} p_{2j}^2 - \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} 2a_{1i,2j} p_{1i} p_{2j} \tag{4.5}$$

The subscripts 1 and 2 in all variables stand for descriptions F_1 and F_2, respectively, and $a_{k,l}$ is the similarity coefficient between two colors c_k and c_l,

$$a_{k,l} = \begin{cases} 1 - \frac{d_{k,l}}{d_{max}} & d_{k,l} \leq T_d \\ 0 & d_{k,l} > T_d \end{cases} \quad \textbf{(4.6)}$$

Where $d_{k,l} = \|c_k - c_l\|$ is the Euclidean distance between two colors c_k and c_l, T_d is the maximum distance for two colors to be considered similar and $d_{max} = \propto T_d$. In particular, this means that any two dominant colors from one single description are at least T_d distance apart. The MPEG-7 standard recommends a value for T_d between 10 and 20 in the CIELUV color space and for $\propto$ between 1.0 and 1.5.

Observing the formula, it can be concluded that if the distance is 0, the two DCD are the same and if two images are similar the distance will be near 0. As a result, the matching process will let us identify images with similar color descriptors.

4.4 Implementation

Once the DCD has been described and also how it can be used to carry out the matching between images, it is possible to make an implementation to adjust this descriptor to the objectives of this project.

The DCD will let us order images according to its dominant colors. To achieve this objective, firstly the DCD has been extracted from these images to afterwards compare these descriptors with the matching technique. This means that, as in the case of the CLD, it is possible to define two main parts in the implementation of this method:

- Process a database of pictures to obtain its DCD
- Find similarity matching between an input picture and the processed database

Fig. 4.5 shows the process of analyzing a database using the DCD:

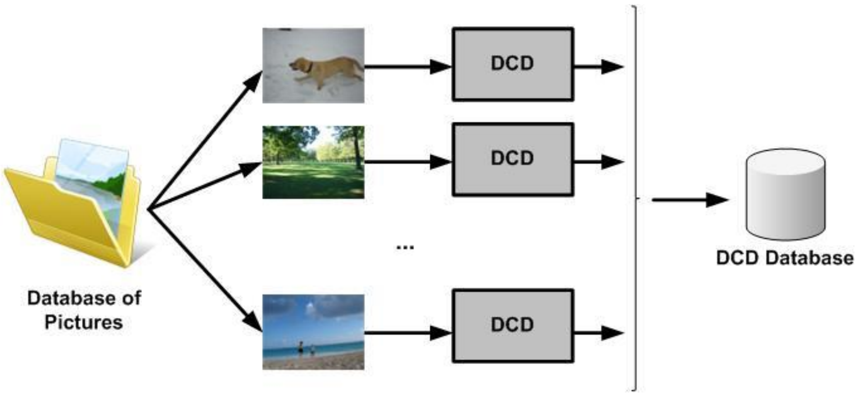

Fig. 4.5 DCD: Process a database

The process is the same that in the case of the CLD. First, an image of the database is uploaded into memory and its descriptor is computed as previously explained. This process is repeated for each image in the database and as a final result a database of DCDs linked to the images that represent is obtained.

The following table shows the inputs and outputs of this stage:

Table 4.4 Inputs and Outputs process a database for DCD

Input	**Output**
Path of a Database with N pictures	DCD Database {image name$_i$, F_i = {(c,p)} i = 1..N

The similarity matching finding process is represented in the following figure:

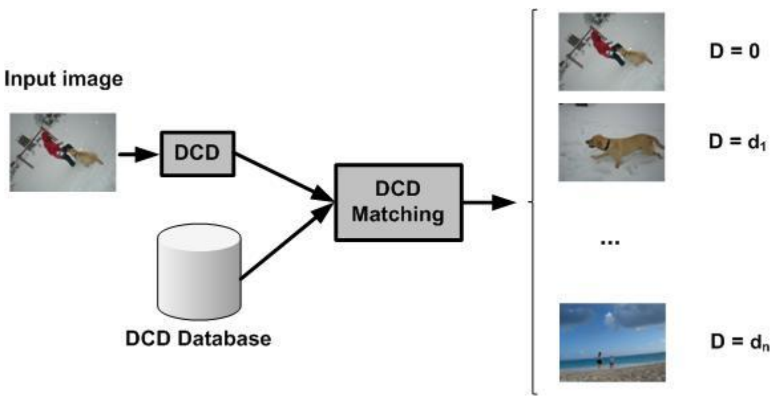

Fig. 4.6 DCD: Find similarity matching

Once the database of images has been analyzed, the matching between an input image and the database of DCD is carried out. With this process we will order the images according to increasing distance taking into account its dominant colors.

Here the process is the same as in the case of the previous descriptor. The input parameters are the input image and the database of DCD obtained in the previous step. Once the process of matching is finished, the images of the database are ordered into increasing distances, where D = 0 means the same color descriptor and $d_1 < d_n$.

Table 4.5 Inputs and Outputs find similarity matching for DCD

Input	**Output**
Input image and DCD database	Images ordered according to increasing distances

The standard recommends the use of three or four dominant colors for the DCD to provide a good characterization of the region colors. In this project we have decided to select four dominant colors. The reason is that after carrying out different test, it has been observed that four colors were sufficient to properly describe an image.

Also, the necessary parameters for the matching process, T_d and $\propto$, have been defined with the following values:

- $T_d = 10$
- $\propto = 1.2$.

These values have been chosen due to the results obtained with them were satisfactory as well as they accomplish the interval (see section **4.3**) recommended by the MPEG-7 standard.

The MATLAB environment has been used to obtain this implementation as well as the same laptop than in the case of the CLD.

4.5 Results

The results obtained with the implementation of the DCD are shown in this section. The tests were carried out with the same database of 100 pictures as in the case of the CLD (these images can be consulted in **Annex A**).

The DCD provides a compact description of the representative colors in an image. Therefore, it is expected that the DCD will be able to group images that have similar dominant colors. This property is interesting if we want to fulfill the objective of this project: order images according to its color.

The first step in order to observe this characteristic is to analyze the database of images. Once this step has been finished and all the DCDs have been obtained, one of the images of the database has been chosen in order to search the similarities among it and the rest of images.

As in the case of the CLD, different tests have been carried out with different photos of the database in order to evaluate its performance. In this case, it is expected to obtain images with similar dominant colors ordered according to increasing distances. An example of the system performance is represented in **Fig. 4.7** where the dominant colors of the input image are grey and white (a snow-covered landscape). The results are also presented in a MATLAB figure where the first ten pictures are ordered in increasing distance.

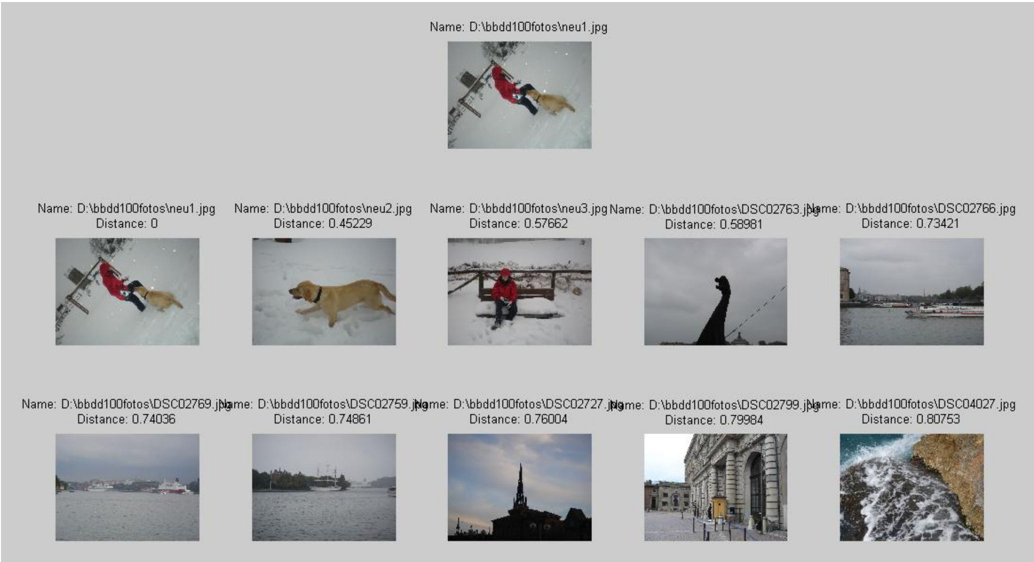

Fig. 4.7 DCD Results

In this particular example it can be observed that the image with less distance (distance 0) is the input image (it is detected in the database) and the following ones are the other images with similar dominant colors. Furthermore, the same results have been obtained using, as an input to the system, images with different dominant colors, such as green, blue, brown…

The obtained results were satisfactory. However, in this case the computing time when the descriptors were extracted was very high:

- Analyzing DCDs of the database of 100 pictures: **33 min**
- Input image DCD extraction and Matching with 100 DCD: **45 seconds**

It is necessary to decrease this computing time if we want to use the results of this descriptor in a near real time environment as the one suggested in this project.

After studying different possibilities we decided to apply a low pass filtering step and decrease the resolution of the images in order to reduce the number of pixels. As a result, a decrease of the time to converge of the K-means algorithm is obtained. This work is explained in the next section.

4.6 Improvements

As it has been seen in the previous section, it was necessary to carry out a pre-process step before the DCD extraction in order to reduce its computing time.

The extraction procedure was analyzed and it was obtained that the precise point where the extraction of this descriptor was slower was during the process of clustering the dominant colors, so using the K-means algorithm. We have observed that the K-means, for high resolutions images, spends a high time to converge in order to determine the different color clusters.

As the modification of the K-means algorithm in order to reduce its time to converge was a very complex process, we decided to add a pre-process step before the DCD extraction. This pre-process step consists on a low pass filter applied to the input image and a reduction of its resolution.

The low pass filtering step is done in order to reduce sudden colors changes since low pass filtering makes colors smoother. It has been decided to use a FIR filter using a 5x5 template filtering which averages the pixel values of the image.

Once the low pass filtering step is done, it has been decided to decrease the resolution of the images in order to reduce the number of pixels. After extensive tests on several resolutions, we decided to reduce the resolution of the images up to a CIF size (352x288 pixels). We tested this resolution and successful results were obtained.

Therefore, the DCD extraction process is the following:

Fig. 4.8 DCD with the pre-process step

With this pre-process step added to our implementation several tests were performed and the obtained results were satisfactory. The computational burden to obtain the DCD was considerably reduced (practically in a 3 factor).

- Analyzing DCDs of the database of 100 pictures: **9 min**
- Input image DCD extraction and Matching with 100 DCD: **17 seconds**

Nevertheless, as well as a faster extraction of the descriptor, it was also important to obtain similar results to the ones obtained in the previous section. Several tests were performed using different input images in order to compare the distances obtained with or without the pre-process step. In all the tests the differences between the distances were very small. One example of this system performance with the pre-process step is represented in the following figure:

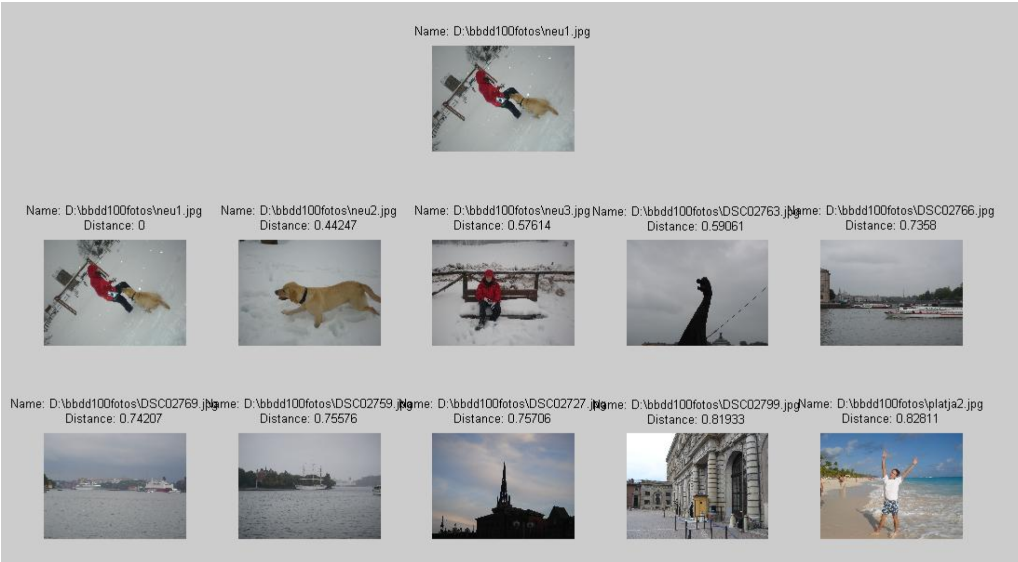

Fig. 4.9 DCD Results with computational improvements

For this particular case, it can be observed that the differences between distances is almost imperceptible as well as the dominant colors of the images are the same for all of them. For instance, the first five distances obtained in the results with and without the pre-process step can be observed on **Table 4.6**.

Table 4.6 Distances without pre-process and with pre-process

	Distances without pre-process	**Distances with pre-process**
1	0	0
2	0.45229	0.44247
3	0.57662	0.57614
4	0.58981	0.59061
5	0.73421	0.7358

Similar results have been observed in many different tests performed during the execution of this project.

CHAPTER 5. INTEGRATION OF CLD AND DCD IN A COMMON MEASURE

On the one hand, the CLD is a very compact and resolution-invariant representation of color for high-speed image retrieval and it is designed to efficiently represent the spatial distribution of colors. On the other hand, the DCD provides a compact description of the representative colors in an image or image region. Consequently, why do not we integrate the CLD and the DCD in a common measure in order to obtain the advantages of each one?

In this chapter the integration between the Color Layout Descriptor and the Dominant Color Descriptor in a common measure is described. First of all it will be justified why both descriptors can be integrated. Finally, the implementation of the algorithm and some results will be presented.

5.1 Justification

One of the main objectives of this project is to characterize images in order to create an automatic sequence that is perceptually pleasant. During the execution of this project we have selected the CLD and the DCD as color descriptors. This decision was carried out to take the advantages that DCD and CLD offers. The DCD lets to describe effectively the dominant colors of an image and CLD is a compact, resolution-invariant representation that retains the spatial distribution of the color of an image.

Therefore, an implementation of the CLD and of the DCD using the MATLAB environment has been performed. For each one, it has been analyzed the different tests carried out in order to improve its performance. First, we have improved the robustness of the CLD against brightness and contrast changes of images. Next, a pre-process step before the DCD extraction was added in order to reduce its computing time.

Once the two color descriptors have been properly tested and improved, it has been decided to integrate its results in a common measure in order to obtain the advantages that they bring to us. In first place, we need to extract the CLD and the DCD of an image and then modify the matching process to integrate it in a common measure. In the next section this process will be explained.

5.2 Implementation

We aim to use the integration of the CLD and the DCD in a common measure in order to perform an image sequencing according to its colors. Because of this, as in the case of the CLD and of the DCD, is possible to define two main parts for this method:

- Process a database of pictures to obtain its CLD and DCD
- Find similarity matching between an input picture and the processed database

The extraction procedure of the color descriptions is the same as in the case of the previous descriptors. The following figure shows the process of analyzing a database:

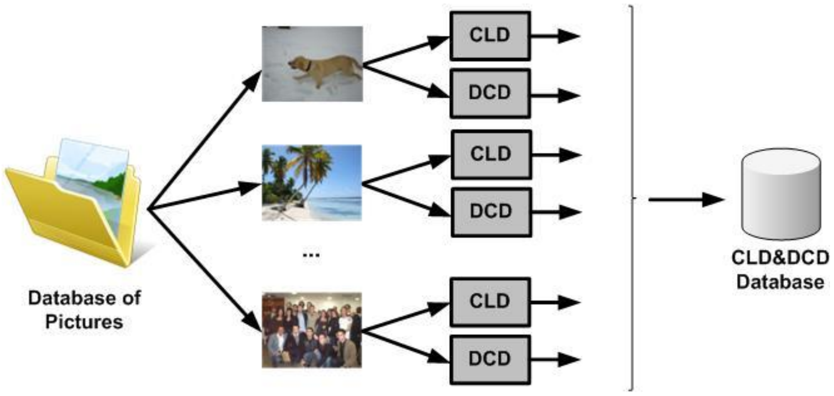

Fig. 5.1 CLD and DCD: Process a database

In this process, a database of pictures is analyzed in order to obtain the CLD and DCD representing each picture. This process consists of uploading the image into memory and computing both descriptors separately as explained in previous chapters. The final result is a database of CLDs and DCDs linked to the images that represent.

The similarity matching finding process is represented in the following figure:

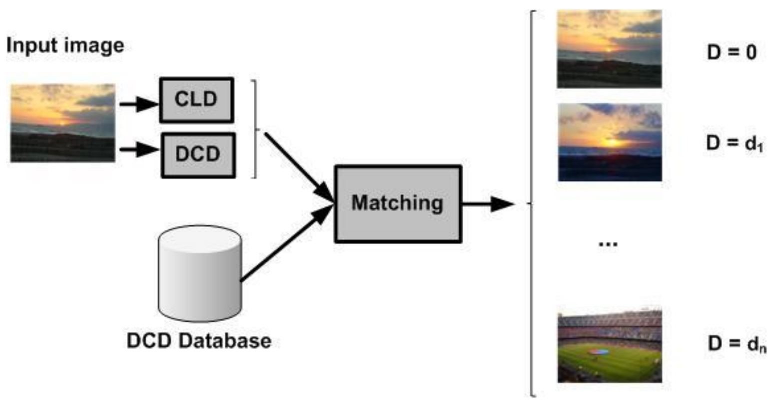

Fig. 5.2 CLD and DCD: Find similarity matching

In this case, the matching process has changed compared with the find similarity matching of the CLD or the DCD. If we consider the input image with its corresponding CLD and DCD and one picture of the database, the matching process can be described as:

- Input image:
  - CLD_1: {DY, DCb, DCr}
  - DCD_1: $F_1 = \{(\boldsymbol{c}_{1i}, p_{1i})\}, \quad (i = 1, 2, \ldots, N_1)$
- CLD and DCD linked to one photo from the database:
  - CLD_2: { DY', DCb', DCr' }
  - DCD_2: $F_2 = \{(\boldsymbol{c}_{2i}, p_{2i})\}, \quad (i = 1, 2, \ldots, N_2)$

Firstly, the D_{CLD} is computed as described in section 3.3:

$$D_{CLD} = \sqrt{\sum_i w_{yi}\,(DY_i - DY_i')^2} + \sqrt{\sum_i w_{bi}\,(DCb_i - DCb_i')^2} + \sqrt{\sum_i w_{ri}\,(DCr_i - DCr_i')^2} \quad \textbf{(5.1)}$$

Secondly, the D_{DCD} is also computed as described in section 4.3:

$$D_{DCD}^2(F_1, F_2) = \sum_{i=1}^{N_1} p_{1i}^2 + \sum_{j=1}^{N_2} p_{2j}^2 - \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} 2a_{1i,2j} p_{1i} p_{2j} \quad \textbf{(5.2)}$$

Once both distances are obtained, they are weighed by the square root of the sum of all the distances squared, $\sqrt{\sum_i D_{CLDi}^2}$ or $\sqrt{\sum_i D_{DCDi}^2}$ respectively, in order to obtain values from 0 to 1.

Finally, the common distance $D_{CLD\&DCD}$ is computed as:

$$D_{CLD\&DCD} = \beta * D_{CLD} + (1 - \beta) * D_{DCD} \quad \textbf{(5.3)}$$

Observing the formula, it can be extracted that if the distance is 0, as in the case of the previous descriptors, the two CLD and DCD are the same and if two images are similar the distance will be near to 0.

As a result, once the matching process is finished, all images of the database are ordered in increasing distances, where $D_{CLD\&DCD}$ = 0 means the same color descriptor and $d_1 < d_n$.

With this distance, $D_{CLD\&DCD}$, it is obtained a common measure that let us obtain the advantages of each descriptor.

In this implementation it is selected β = 0.5 in order to give the same weigh to CLD as to DCD, due to we do not know previously if we have to give more importance to one descriptor than another.

5.3 Results

The results obtained with this implementation are shown in this section. The tests were carried out using the same database of 100 pictures as in the case of the previous descriptors. Different tests have been carried out with different photos of the database in order to evaluate its performance. Only some examples are given below. It was considered interesting to use the same input images as the ones used in the examples given in the CLD and DCD sections, in order to show to the reader its performance.

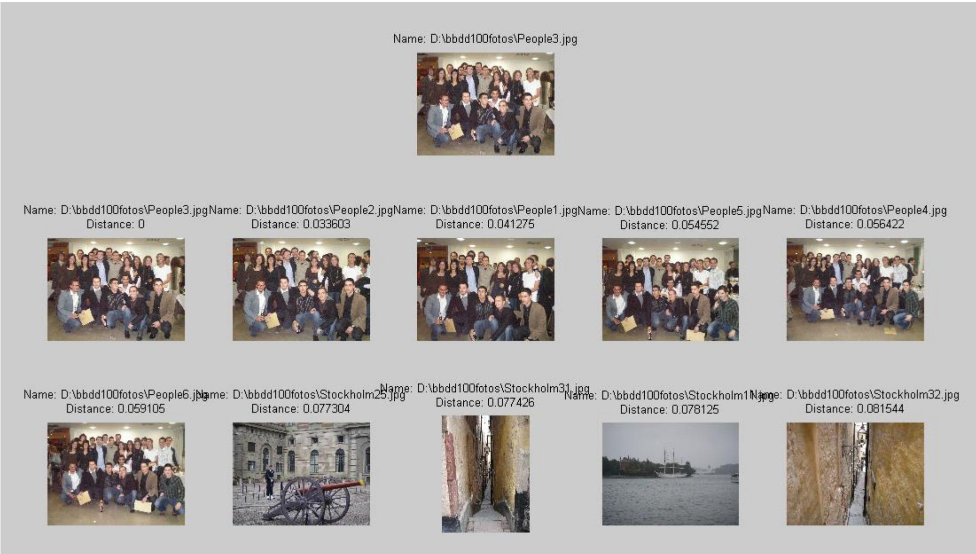

Fig. 5.3 Integration results 1

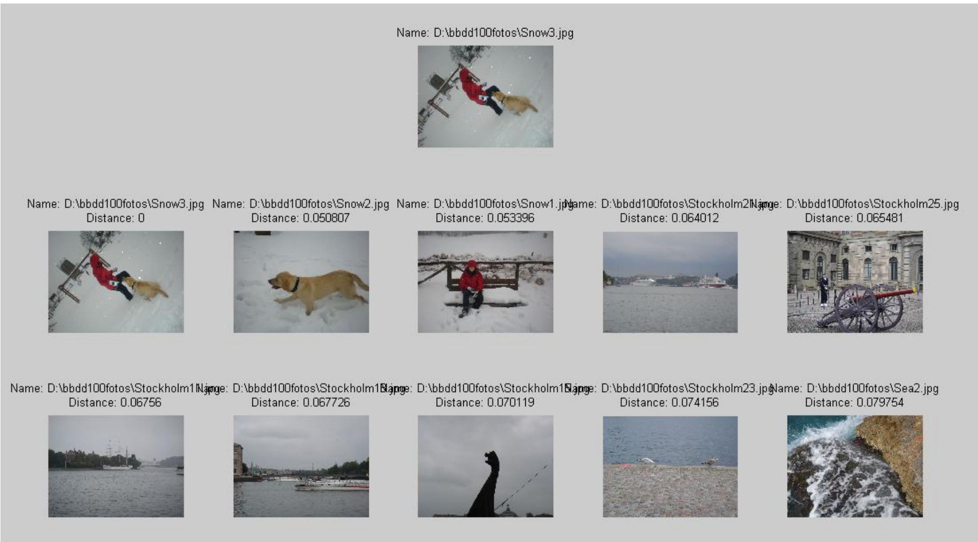

Fig. 5.4 Integration results 2

For these particular two examples, we have obtained similar results compared to the ones using each descriptor alone.

Therefore, after analyzing the different tests performed, we considered the obtained results of this integration satisfactory, especially if we take into account that they are the integration of two color descriptors.

Furthermore, it is important to analyze the computing time in order to obtain the descriptors and the results:

- Analyzing the CLDs and DCD of the database of 100 pictures: **9 min minutes 20 seconds**
- Input image CLD&DCD extraction and Matching with 100 descriptors: **20 seconds**

In this case, this computing time is approximately the total of the computing time of the CLD and DCD; this means that this integration does not imply an increase of this time.

Finally, as we want to use the results obtained on this project in order to perform an automatic sequencing of different pictures according to its colors, we thought about how we could evaluate which of the results obtained from the different methods, the CLD, the DCD and the integration of both results, is more similar to an arrangement performed by a person. This work is reflected in next chapter.

CHAPTER 6. EVALUATION AND RESULTS

During the execution of this project, two color descriptors, the CLD and the DCD, have been studied, analyzed and tested in order to describe the colors of an image. Furthermore, both descriptors have integrated in a common measure to obtain the advantages that each one brings. Nevertheless, as one of the objectives of this project was to create an electronic album, it has been decided to evaluate if the three methods implemented are similar to the perceptual vision of the people.

The similarity classification between images based on colors is difficult to evaluate since there is no ground truth of what the ideal results should be. However, it has been defined an evaluation method in order to know if the results obtained by the descriptors are similar to the results obtained by human beings.

In this chapter will be explained this evaluation method and how it has been used. Finally, the results obtained are presented as well as some conclusions.

6.1 Evaluation method

Color is an important visual attribute for both human vision and computer processing. Therefore, it has been considered important to evaluate if the results obtained with the implemented descriptors in this project are similar to the perceptual vision of the people.

Different evaluation methods, in order to obtain this evaluation, were discussed. In a first approach, it was decided to perform an arrangement of all the 100 photos in the database using the three methods: the CLD, the DCD and CLD&DCD (the combination of the CLD and the DCD). Then, different people were asked to perform an arrangement of the same 100 pictures of the database in order to compare both sequences. However, after performing the test to three different people, we realized that the problem of this method was the time that each person spent to perform the arrangement. Consequently, it was decided to perform another evaluation method. Even so, the results obtained with this method can be seen in the **Annex B**.

After analyzing other possibilities, finally it was decided to perform a test that let us to know if pictures selected by people, taking into account the color, are the same than the results obtained from the color descriptors. A score protocol was defined and a final value is obtained for each color descriptor.

One example of these tests can be seen in the following figure:

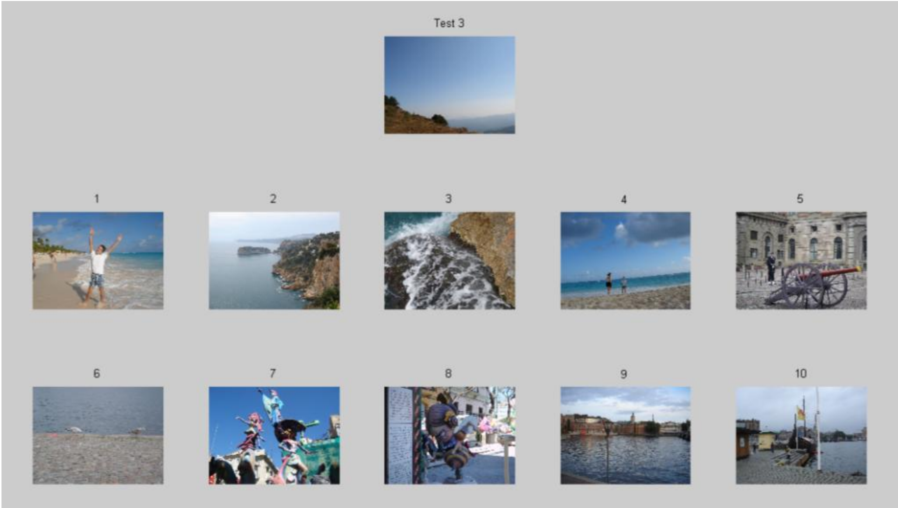

Fig. 6.1 Test

In order to quantify the similarity between people's selection and descriptors' results, for each human test the top image is the reference image and below 10 pictures numbered from 1 to 10 are added. Then, it was asked to people to select the top three images similar to the reference taking into account the color and distribution in the image.

With this test it is possible to know if the top three photos selected by the people are similar to the results of the descriptors. This is due to the fact that these 10 photos are selected according to the results of the CLD, the DCD and the CLD&DCD. In order to obtain them, the matching process of each method using as input image the reference one was applied. Then, the three images with less distance were selected. For example, on the previous test, pictures 1, 3 and 4 corresponds to the three images with less distance for the integration of the CLD and DCD. The following figure shows this photo selection for the previous example test:

Fig. 6.2 Selection of the photos of the test

We have used random pictures in order to complete the 10 pictures, in the previous example they are numbers 5 and 8.

The marks defined in order to quantify the similarity between people selection and descriptors results are the following. We have to underline that we only have taken into account the first image with less distance for each descriptor (in the case of the previous example the photo number 7 for the CLD, the 10 for the DCD and the 4 for the CLD&DCD) to facilitate the evaluation and because these pictures are the most similar for this color descriptor. The marks used are described below:

- If the first photo selected by the person is the same of the descriptor: 6 points
- If the second photo selected by the person is the same of the descriptor: 4 points
- If the third photo selected by the person is the same of the descriptor: 2 points
- If the selected photo by the person is not the same of the descriptor: 0 points

An example score obtained from a person after performing the test of **Fig. 6.1** can be observed in the following table:

Table 6.1 Marks obtained from one person

		Test 3		
	CLD	**DCD**	**CLD&DCD**	**Person**
	7	**10**	**4**	**4**
Marks	**2**	**0**	**6**	**2**
				7

The column called *Person* corresponds to the position in the test of the top three of the person; in this case the first photo selected by a person is number 4, in second place number 2 and finally number 7.

Below each color descriptor name appears the position of the image with less distance respect the reference image; in this case for the CLD is photo number 7, for the DCD number 10 and for the CLD&DCD photo number 4.

Then, the marks are reflected in the row called *Marks*. In this particular example, the CLD&DCD method gets 6 points as the person also selects the picture number 4 in first place. The CLD gets 2 points as it matches with the third picture selected by the person. Finally, DCD gets 0 points as its image with less distance does not appears on the top 3 of the person.

Using these marks, information about similarities between descriptors' selection and people's selection can be obtained.

15 tests were performed to each person. Therefore 15 photos from the database were selected and for each one the same procedure was followed. All the tests can be seen on the **Annex C**.

Once all the tests have been performed, all the marks of each test were added in order to obtain the final score. In the example, each color descriptor has obtained a final score represented in the following table:

Table 6.2 Final score of one person

TOTAL person		
CLD	DCD	CLD&DCD
54	44	56

For this particular case, the method that obtains a higher score is the CLD&DCD, then the CLD and finally the DCD. Analyzing these results, we can determine the similarity between the selection of the person and the results of the method. In the example, for this person the CLD&DCD results are more similar to its selection.

Finally, during the execution of this project the 15 tests have been performed to 14 different people. After analyzing the results all the scores were added and the following table shows the marks obtained for each descriptor:

Table 6.3 Final score of 14 people

	TOTAL	
CLD	DCD	CLD&DCD
726	530	758

For these 14 people in particular, it can be observed that the method that obtains a highest score is the integration of the CLD and the DCD, in second place the CLD and afterwards the DCD. The results of all the test of each person can be consulted on the **Annex D**.

6.2 Conclusions

In this chapter it has been defined an evaluation method in order to know if the results obtained by the descriptors are similar to the ones obtained by people. We have performed these tests to 14 different people. Nevertheless, it is necessary to stand out that the obtained results are very specific and can't be generalized easily as only 15 photos and our small database have been used.

The only objective of this evaluation was to get a perception of how the methods developed and if they were working correctly or not.

Therefore, after carrying out the tests what we can extract is that the two methods implemented in this project, the CLD and the DCD, are good comparing them with the subjective selection carried out by persons. Analyzing each test, we can observe that in most of the cases the selections performed by people are the same as the selection of the descriptors. Also, we can extract that the CLD seems to get better results than the DCD. Moreover, the integration of the two color descriptors in a common measure can obtain better results to the visual perception of persons; even if the weights of the integration of both methods are selected giving the same weigh to each descriptor. As future work it could be interesting to perform several tests with different weighs in order to obtain better results.

In conclusion, after analyzing the results obtained and performing these tests, it can be extracted that the 3 methods developed during the execution of this project work quite satisfactory when compared to the results obtained by the visual perception of persons.

CHAPTER 7. APPLICATION: AN E-ALBUM

One of the objectives of this project is to create an electronic album taking into account the colors of the images. To reach this objective, first it has been studied different color descriptors. Then, we have implemented two of them: the CLD and the DCD, as well as we have integrated the results of them in a common measure. Finally, the three methods comparing their results to the perceptual vision of people have been evaluated. We have concluded that the results were satisfactory, so it has been decided to apply the results to a real application, to the creation of an electronic album.

In this chapter it is attempted to give a real application to the image classification provided by color descriptors studied in this project. The application will be an electronic album (also called e-album) where the user can upload images and then are ordered according on its color.

7.1 Justification

In this project different color descriptors have been studied in order to be able to organize images. First, the CLD has been studied and its performance has been improved. Afterwards the same has been done with the DCD. Finally, both results have been joined in a common measure and its performance has been evaluated.

In this point it has been attempted to give a real use to the study carried out. Because of that, a web application which allows classifying uploaded images depending on its color has been developed. In other words, an e-album where the order of the images is determined by the color has been created.

At this point, we studied how we could arrange the photos. This arrangement can be done according to different aspects, such as artistic factors. For example, it could be possible to create an electronic album with a color coherency, or it could be interesting to detect similar images and put them in different positions. As different options were possible, for the execution of this project it has been decided to perform color-based arrangement. In other words, to create an electronic album starting with some picture and arrange the rest taking into account the colors. This process is explained in the following section.

7.2 Implementation

It has been decided to arrange the photos of the album according to its color. The method selected to obtain the sequences of ordered pictures is the CLD&DCD, in other words, the integration of the CLD and DCD descriptors to obtain the electronic album. It has been decided to use their results as they were the ones that have obtained a better score during the evaluation.

To create the album, it has been performed the matching process starting with a photo (the first alphabetical) in order to obtain the following with less distance. Then, it has been repeated the same process with this new photo until the last one. The following figure represents this process:

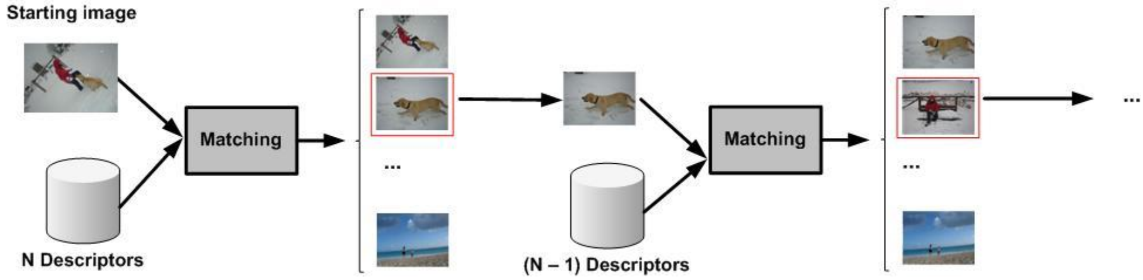

Fig. 7.1 Arrangement of a database

The MATLAB compiler [15] has been used to generate one executable of the code of the CLD&DCD developed in this project. The MATLAB compiler prepares M-file(s) for deployment outside of the MATLAB environment. In particular a standalone application has been generated as it allows creating MATLAB applications that take advantage of the mathematical functions used on MATLAB, yet do not require that end users own MATLAB. Standalone applications were selected because are a convenient way to package the power of MATLAB and to distribute a customized application to the users.

The input of this standalone application is the path of the folder where the images are, and the outputs are the path of the images classified according to the CLD&DCD.

Once the executable has been generated, a Java application has been developed with the *Netbeans* [16] program in order to use this executable.

Finally, using the Java Server Pages (JSP) technology [17], a web page has been developed. JSP is a server side Java technology that allows software developers to create dynamically generated web pages, with HTML, XML, or other document types, in response to a Web client request to a Java Web Application container.

In this case, the Web client request is to obtain an electronic album ordinate according to its colors, and the server side Java technology is to execute the MATLAB standalone application which returns the appropriate arrangement.

Dreamweaver (a web development application [18]) has been used in order to design the web page. The following section shows this development.

7.3 Results

The results of this web application can be seen in the following figures.

Fig. 7.2 is the starting page, where it is explained the process that has to be followed to obtain the e-album as well as the possibility to access to another web page where the functioning of the CLD&DCD is briefly explained.

Fig. 7.2 Start page

The following figure shows the web page where the user can upload its images.

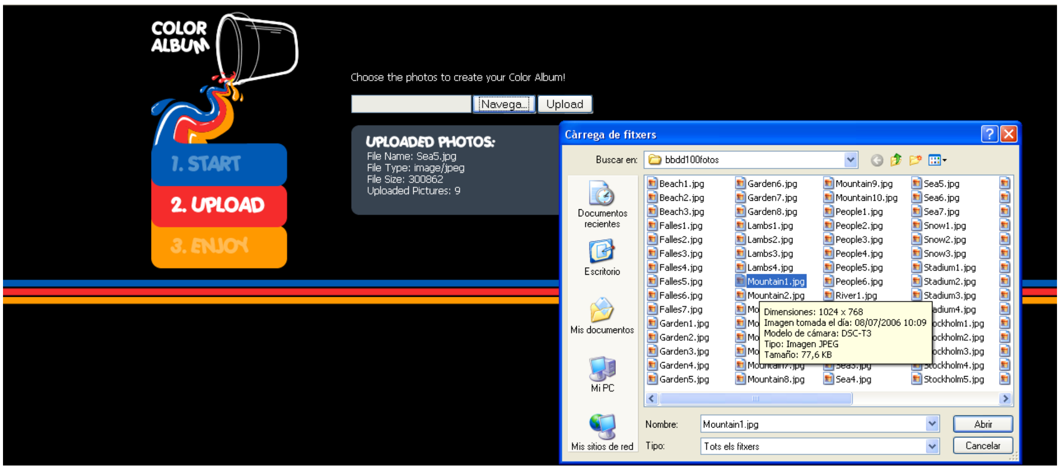

Fig. 7.3 Upload images page

These images are uploaded to the server. Finally, the picture sequence obtained by the CLD&DCD is shown using a Flash viewer [19] as the electronic album:

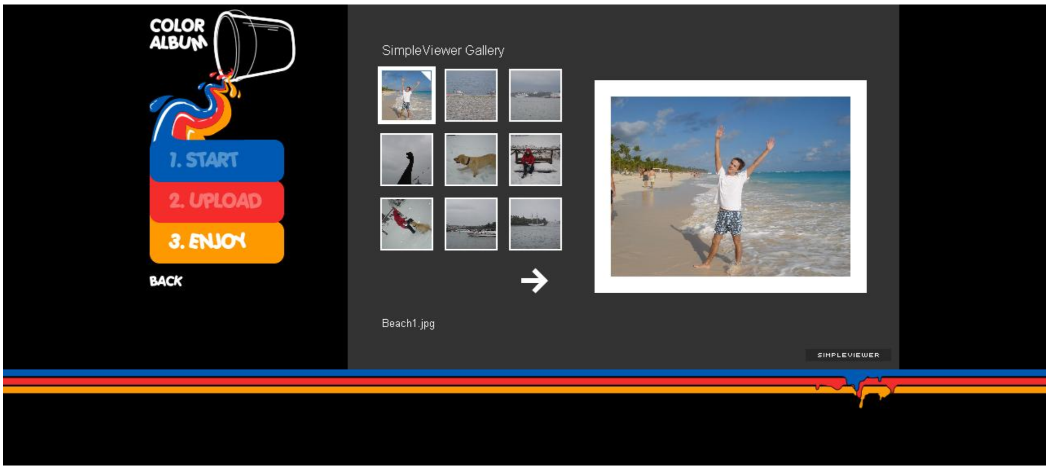

Fig. 7.4 Electronic Album page

It is important to highlight that the results obtained with the standalone application are exactly the same than the obtained using the MATLAB program directly. Therefore, the computing time of the database is approximately the same. For example, for 100 pictures this time would be around 9 minutes. We know that this time is high for a web application (if the user wants to classify 100 pictures), but stand out that this is only a proof of concept. The processing can be highly accelerated if the algorithms are directly programmed in C++.

CHAPTER 8. CONCLUSIONS

The objectives set out at the beginning of this project have been fulfilled in a satisfactory way. It has been used the MPEG-7 low level descriptors to characterize the colors of an image. In particular two color descriptors have been implemented, the CLD and the DCD. Then, these descriptors have been used in order to detect if two images were similar according to its color as well as to be able to group photos with similar colors.

Once the descriptors have been tested and their results improved, we have integrated them in a common measure to take the advantages that DCD and CLD offer. Afterwards, it has been defined an evaluation method in order to know if the results obtained by the descriptors are similar to the ones obtained by people. After analyzing the results, it has been concluded that the 3 methods (the CLD, the DCD and the CLD&DCD) developed during the execution of this project work successfully.

Finally, to demonstrate its use in a real application, it has been developed a web page in order create an electronic album.

It is also necessary to comment the possible environmental impacts of this project. Remark that the use of electronic resources to distribute photos (like electronic albums) implies a significant save in photographic paper as well as in the use of impression inks cartridges, highly polluting.

As future work, this project has to be a starting point to implement other image descriptors in order to be able to include them in a real application of electronic album. A good option would be to detect objects, structures or people faces in the photos. Also, could be interesting to integrate music with the photo composition as well as generate a video with the obtained results.

Another future line could be to adjust the weights of the CLD&DCD in order to improve the obtained results.

To summarize, this work has to be considered as a first step to achieve an application or web page where users can upload their photos and as a result they obtain the photos ordered in an automatic way.

Finally, point out that during the execution of this project I have also learned how to write properly a report in English. We have to thank to the school the opportunity that is given to us in order to improve in this kind of reports. Therefore, it is necessary to remark that the reader can detect in the first chapters a different level of English or in the fluency in the writing respect to the rest of the chapters. We do not want to change the explanations to a more personal style when the standard is defined due to the importance of the accuracy of the language. However, on the rest of chapters I have attempted to use my personal style of English in order to explain what I had made instead of copy literally texts from different sources. This perhaps can be felt in the fluency but I thought that it was more important learning and use my own words.

CHAPTER 9. BIBLIOGRAPHY

[1] Image warehouses on the Web:

http://www.techcrunch.com/2009/04/07/who-has-the-most-photos-of-them-all-hint-it-is-not-facebook/

[2] MPEG: http://www.mpeg.org/

[3] MPEG-7: http://www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.htm

[4] MPEG-7: http://www.m4if.org/resources.php#section40

[5] MPEG-7: http://www.multimedia-metadata.info/

[6] Manjunath B.S., Salembier P., and Sikora T., *Introduction to MPEG-7: Multimedia Content Description Interface*, 1st edition 2002

[7] MPEG-7: http://www.chiariglione.org/mpeg/

[8] RGB image: http://de.wikipedia.org/wiki/RGB-Farbraum

[9] HSV image: http://www.virtual-drums.com/images/HSV_cylinder.jpg

[10] HMMD image: http://www.freepatentsonline.com/6633407-0-large.jpg

[11] Alujas Tejada R., "Eines software per la classificació i autocatalogació de material audiovisual", Projecte Fi de Carrera, 2006

[12] CIELUV image:

http://upload.wikimedia.org/wikipedia/commons/8/83/CIE_1976_UCS.png

[13] K-means:

http://people.revoledu.com/kardi/tutorial/kMean/NumericalExample.htm

[14] K-means:

http://www.mathworks.com/access/helpdesk/help/toolbox/stats/index.html?/access/helpdesk/help/toolbox/stats/kmeans.html&http://www.mathworks.com/cgi-bin/texis/webinator/search/?db=MSS&prox=page&rorder=750&rprox=750&rdfreq=500&rwfreq=500&rlead=250&sufs=0&order=r&is_summary_on=1&ResultCount=10&query=kmeans&submitButtonName=Search

[15] MATLAB compiler:

http://www.mathworks.com/access/helpdesk/help/toolbox/compiler/index.html?/access/helpdesk/help/toolbox/compiler/mcc.html&http://www.google.es/search?q=matlab+mcc+-mv&ie=utf-8&oe=utf-8&aq=t&rls=org.mozilla:ca:official&client=firefox-a

[16] Netbeans: http://www.netbeans.org/

[17] JSP: http://java.sun.com/products/jsp/

[18] Dreamweaver: http://www.adobe.com/es/products/dreamweaver/

[19] Flash viewer:

http://www.simpleviewer.net/simpleviewer/

[20] Matlab documentation:

http://www.mathworks.com/access/helpdesk/help/techdoc/index.html?/access/helpdesk/help/techdoc/&http://www.google.es/search?q=help+matlab&ie=utf-8&oe=utf-8&aq=t&rls=org.mozilla:ca:official&client=firefox-a

[21] Notes from Francesc Tarrés (EPSC), “Tractament digital del senyal” subject

Escola Politècnica Superior de Castelldefels

UNIVERSITAT POLITÈCNICA DE CATALUNYA

ANNEX

TITLE: Color Based Image Classification and Description

MASTER DEGREE: Master in Science in Telecommunication Engineering & Management

AUTHOR: Sergi Laencina Verdaguer

DIRECTOR: Francesc Tarrés Ruiz

DATE: October 20th 2009

ANNEX A. DATABASE OF PICTURES

Database of 100 JPG images used during the execution of this project in alphabetical order:

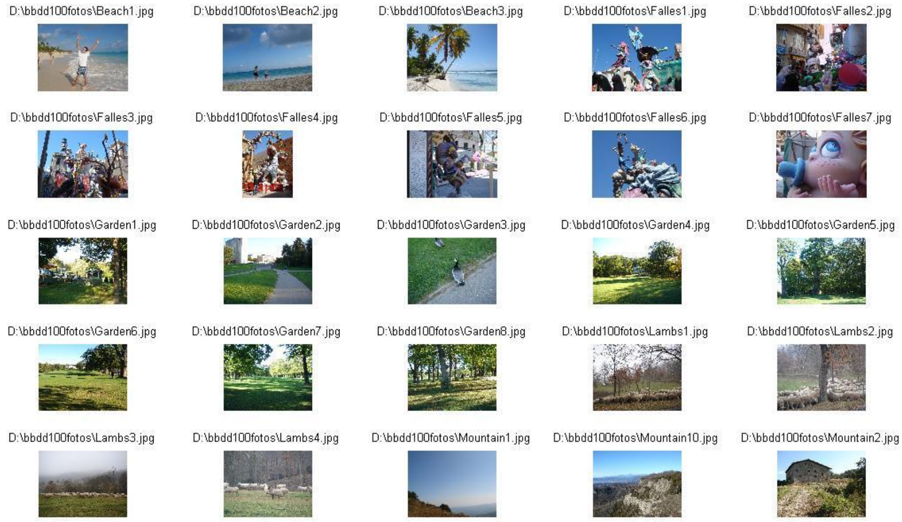

Fig. 1 Photos 1 - 25

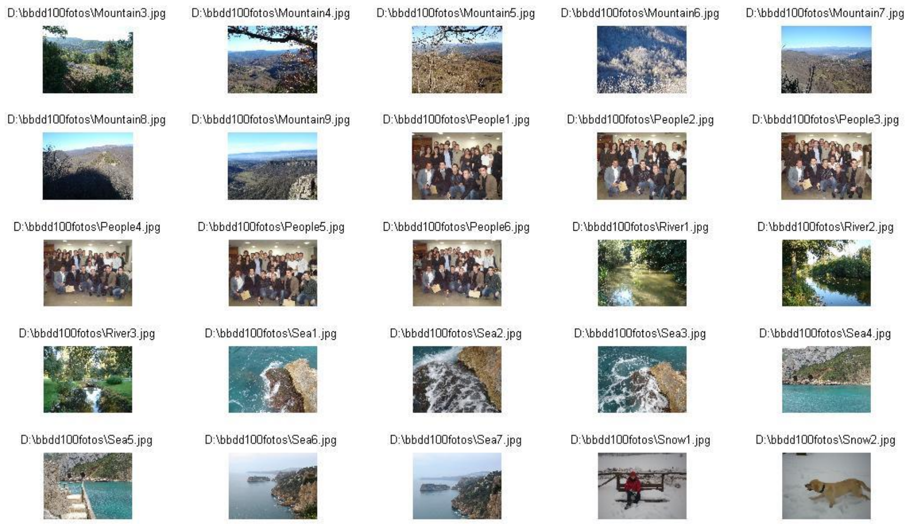

Fig. 2 Photos 26 - 50

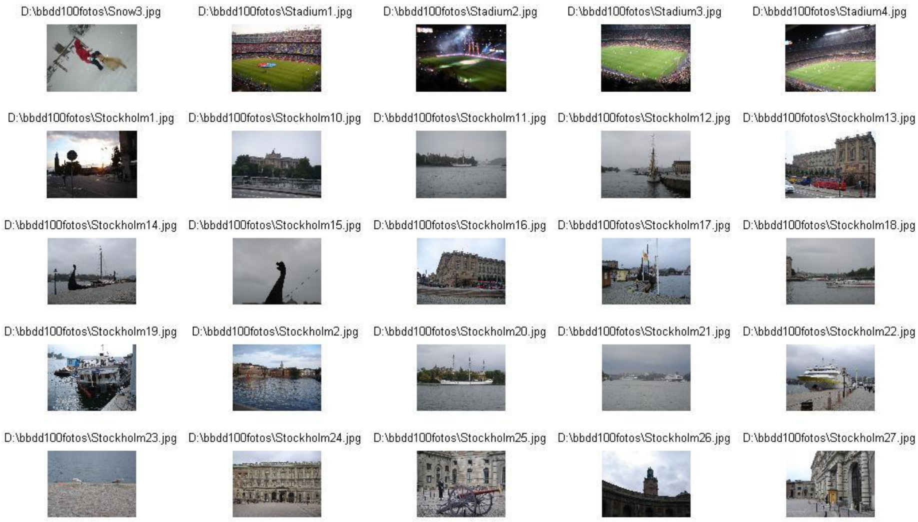

Fig. 3 Photos 51 - 75

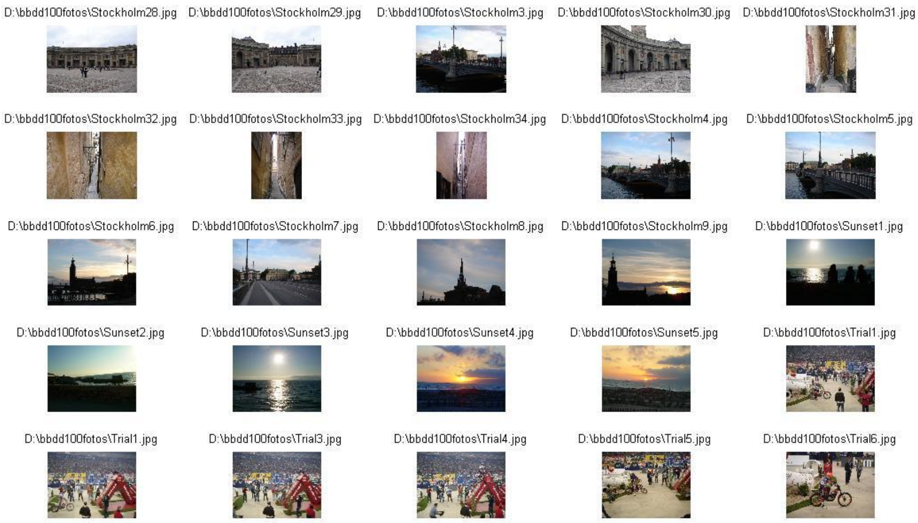

Fig. 4 Photos 76 - 100

ANNEX B. ARRANGEMENT OF 100 PICTURES

In this Annex first can be observed the arrangement performed by the three different methods. Then the arrangement of three different persons is shown and finally the comparison of the different results is presented.
To carry out the image similarity test using the descriptors, a mechanism has been implemented in order to obtain it in an automatic way. The first step is to analyze the database of pictures in order to obtain the corresponding CLD or DCD. Once this process is finished, it is chosen one color descriptor from the database.
This initial color descriptor represents the starting reference image. From this descriptor, the matching is calculated with the rest of the descriptors of the database and the results were classified according to its distance. The result with less distance (D = 0) will represent the input image itself, and it will be deleted from the database because it will not be used anymore in this classification. To continue with the process, the second image with less distance will be taken in order to continue with the classification.
In order to facilitate the comprehension of the results, the images of the database has been renamed according to the scene that appears in it. The chosen names are: Sunset, Mountain, Stadium, Stockholm, Garden, River, Lambs, Sea, People, Trial, Snow, Beach and Falles.

B.1. CLD arrangement

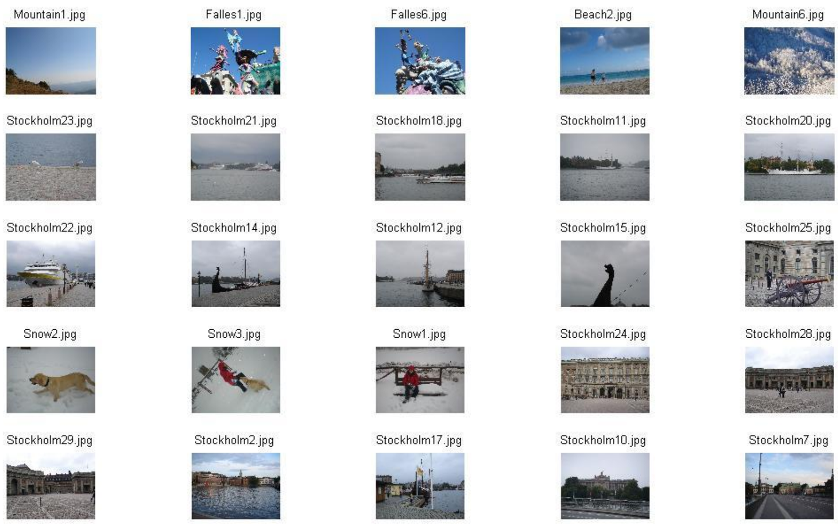

Fig. 1 CLD 1 – 25

Fig. 2 CLD 26 – 50

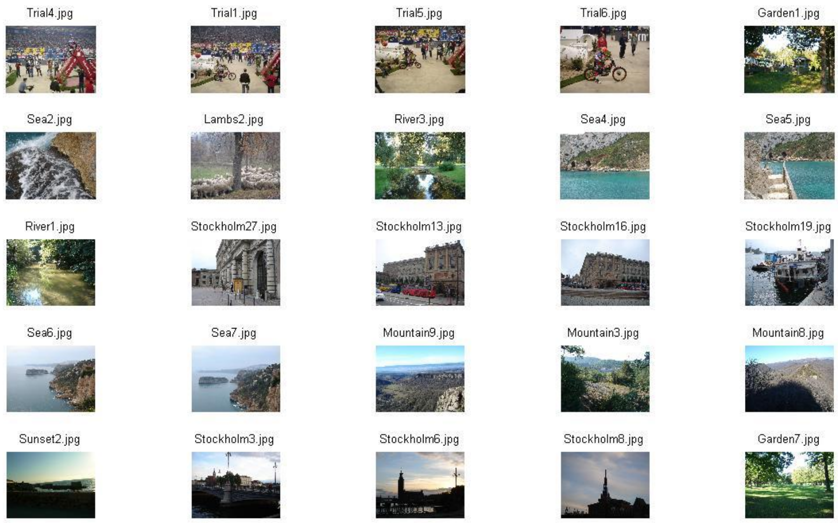

Fig. 3 CLD 51 – 75

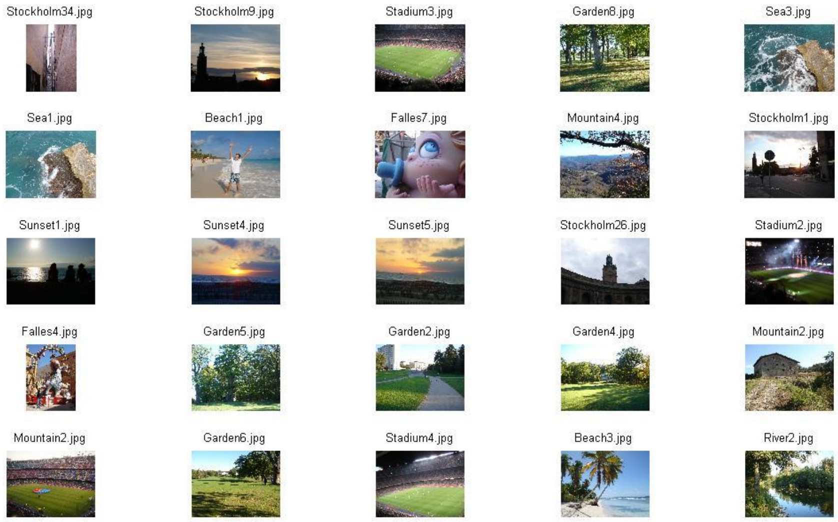

Fig. 4 CLD 75 – 100

B.2. DCD arrangement

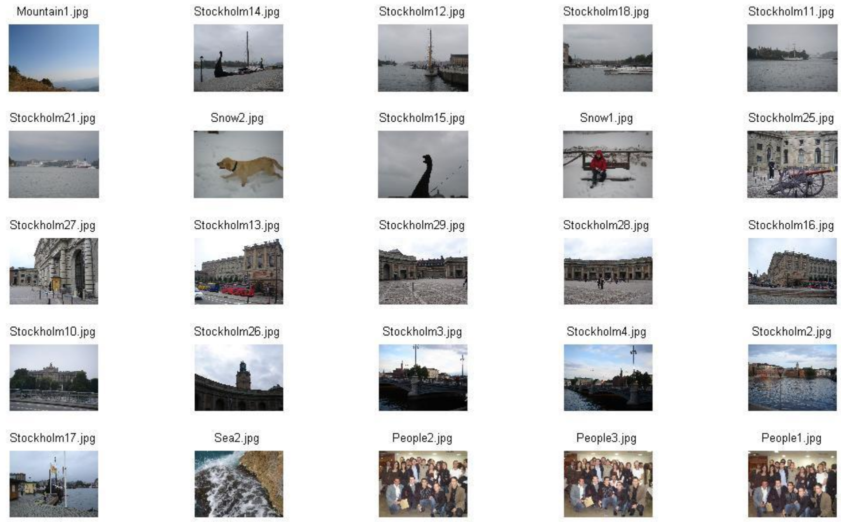

Fig. 5 DCD 1 – 25

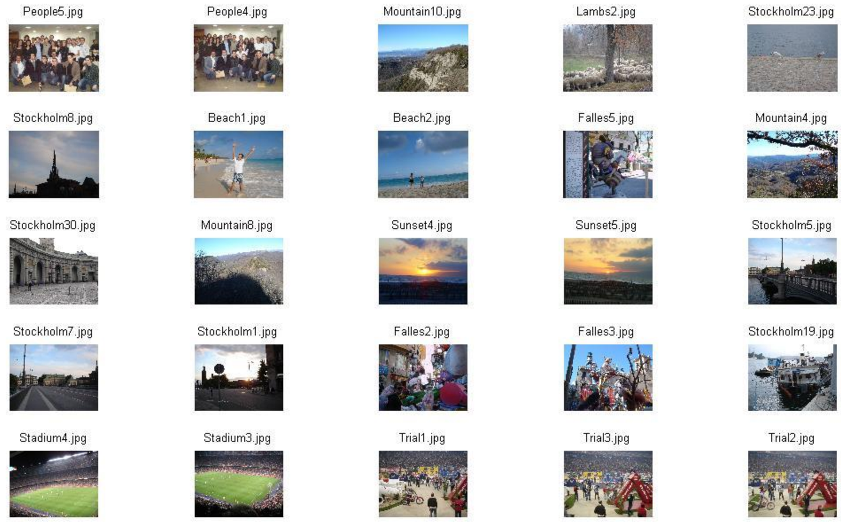

Fig. 6 DCD 26 – 50

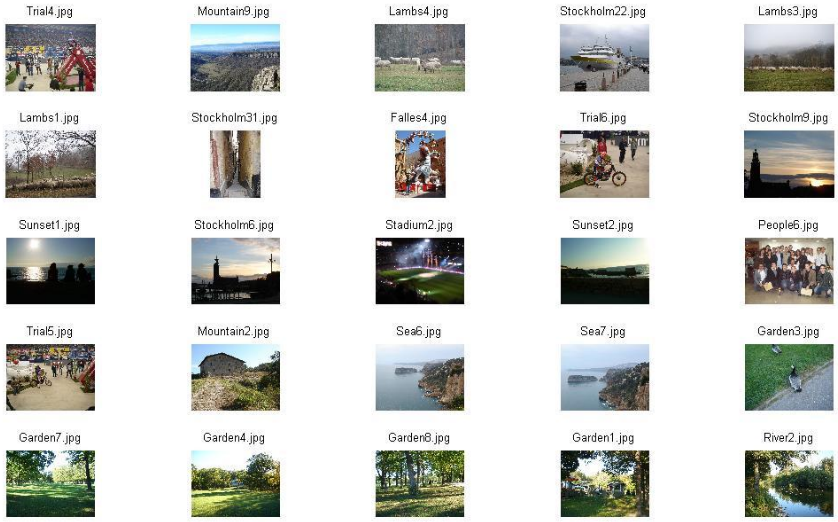

Fig. 7 DCD 51 – 75

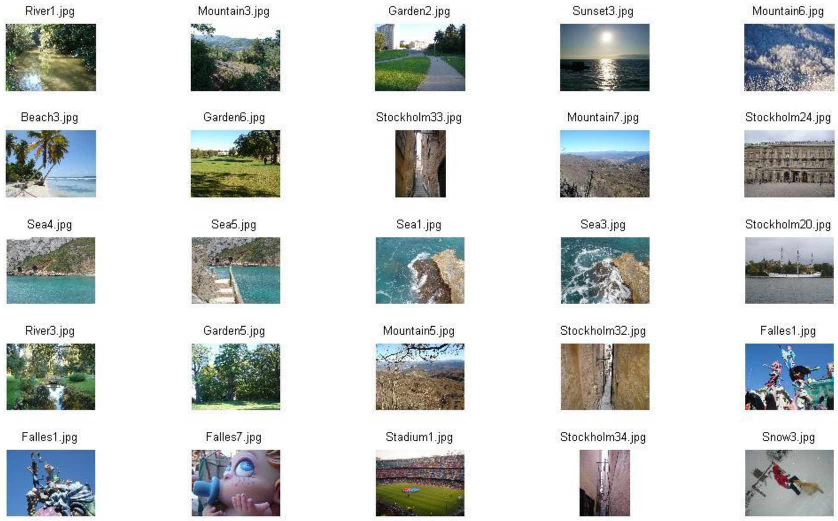

Fig. 8 DCD 76 – 100

B.3. CLD&DCD arrangement

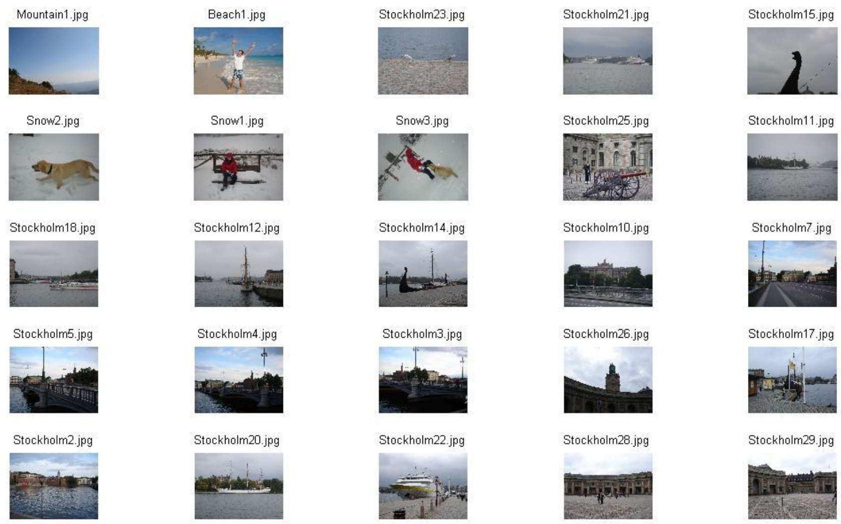

Fig. 9 CLD&DCD 1 – 25

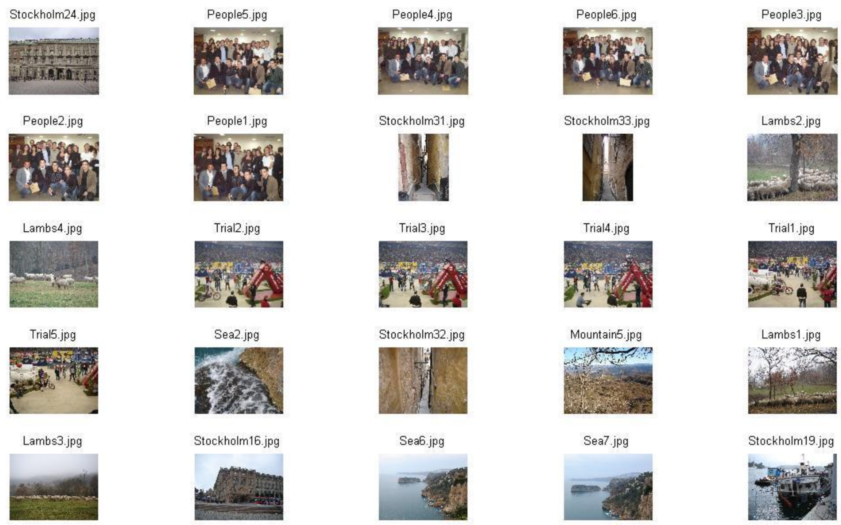

Fig. 10 CLD&DCD 26 – 50

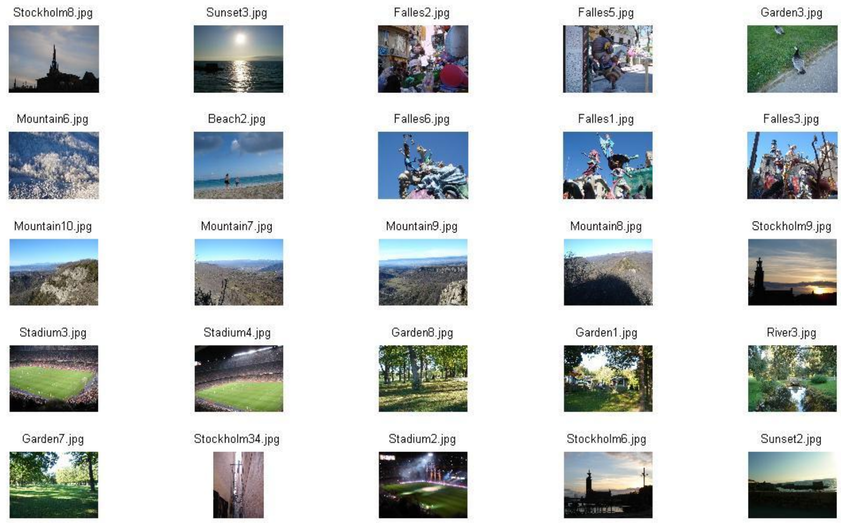

Fig. 11 CLD&DCD 51 – 75

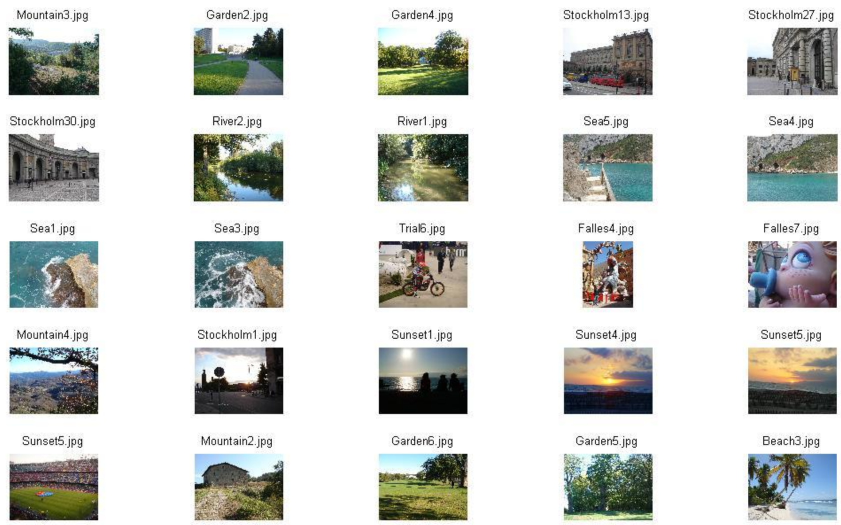

Fig. 12 CLD&DCD 76 – 100

B.4. People's arrangement:

The classification done by the people has been carried out using the same database of images and starting with the same image than the one used in the implementation of the descriptors. Also, it has been asked to classify the pictures according of the colors of the images. Finally, to facilitate the classification, the images have been printed.

Emphasize two aspects: The resolutions of the printed images were small (3x4 cm approximately), in order to facilitate it's use, and the quality of the printed images were medium-low, so maybe some colors are not exactly the real ones.

The applied procedure has been the following one:

- The printed images have been enumerated on the back following the classification given by the CLD or the DCD.
- Then, each person classifies the pictures according to its criteria.
- When the pictures have been classified, the numbers written in the back of each picture has been introduced into a table.

The time that each person approximately spends to do the classification of all the pictures is the following:

- Person 1: 30 min
- Person 2: 25 min
- Person 3: 30 min

B.4.1. Person 1 arrangement

1	Mountain1	21	Stockholm17	41	Trial1	61	Sea1	81	River2
2	Beach2	22	Stockholm19	42	Trial6	62	Sea4	82	River3
3	Beach3	23	Stockholm2	43	Falles4	63	Sea5	83	Garden5
4	Beach1	24	Stockholm4	44	Snow1	64	Falles3	84	Mountain3
5	Mountain6	25	Stockholm7	45	Snow3	65	Falles5	85	Mountain2
6	Mountain4	26	Stockholm5	46	Snow2	66	Falles1	86	Lambs4
7	Mountain5	27	Stockholm3	47	Stockholm15	67	Falles6	87	Lambs3
8	Mountain8	28	Stockholm10	48	Stockholm23	68	Falles2	88	Lambs1
9	Mountain10	29	Stockholm16	49	Stockholm8	69	Stadium2	89	Lambs2
10	Mountain9	30	Stockholm30	50	Stockholm6	70	Stadium1	90	Stockholm32
11	Mountain7	31	Stockholm29	51	Stockholm26	71	Garden2	91	Stockholm33
12	Sea7	32	Stockholm28	52	Stockholm1	72	Stadium4	92	Stockholm31
13	Sea6	33	Stockholm24	53	Stockholm9	73	Stadium3	93	Stockholm34
14	Stockholm22	34	Stockholm27	54	Sunset4	74	Garden4	94	Falles7
15	Stockholm12	35	Stockholm25	55	Sunset5	75	Garden6	95	People1
16	Stockholm21	36	Stockholm13	56	Sunset2	76	Garden7	96	People3
17	Stockholm11	37	Trial2	57	Sunset3	77	Garden3	97	People2
18	Stockholm18	38	Trial3	58	Sunset1	78	Garden8	98	People6
19	Stockholm20	39	Trial4	59	Sea2	79	River1	99	People5
20	Stockholm14	40	Trial5	60	Sea3	80	Garden1	100	People4

Fig. 13 Person 1 classification

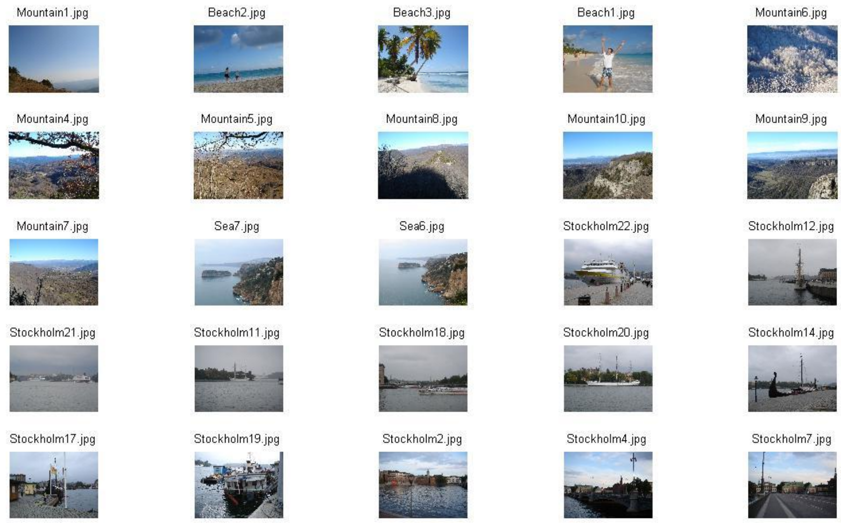

Fig. 14 Person 1: 1 – 25

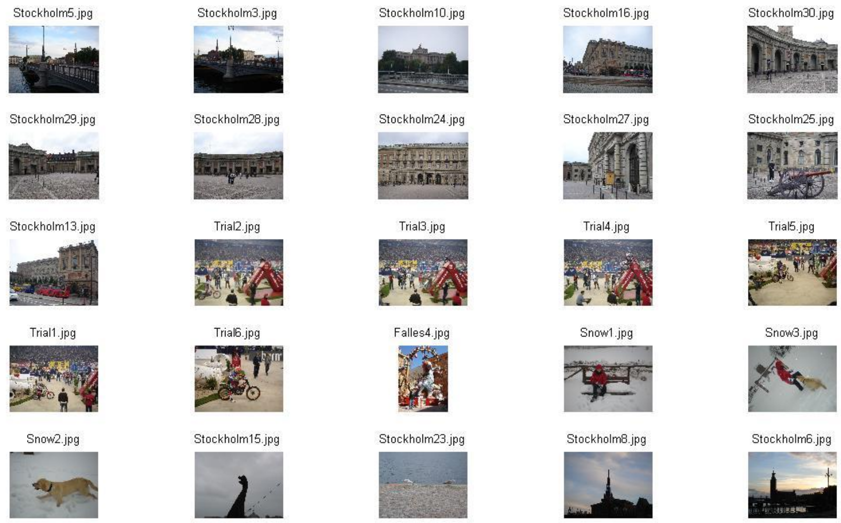

Fig. 15 Person 1: 26 – 50

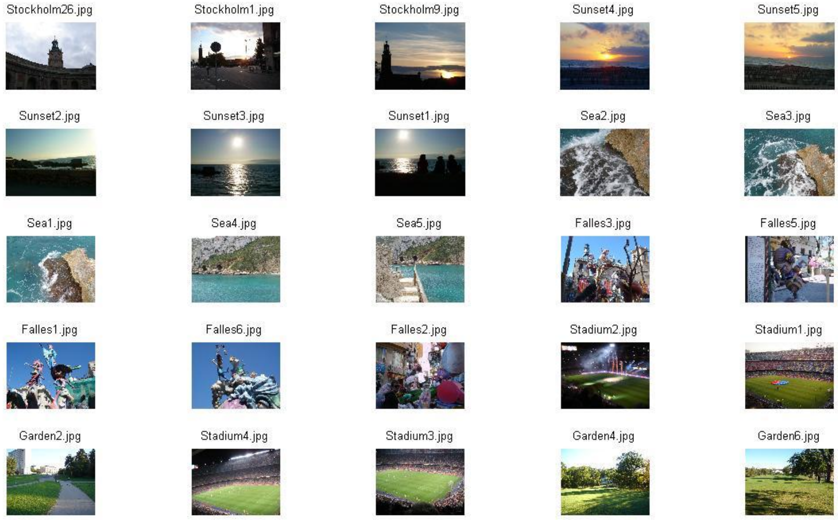

Fig. 16 Person 1: 51 – 75

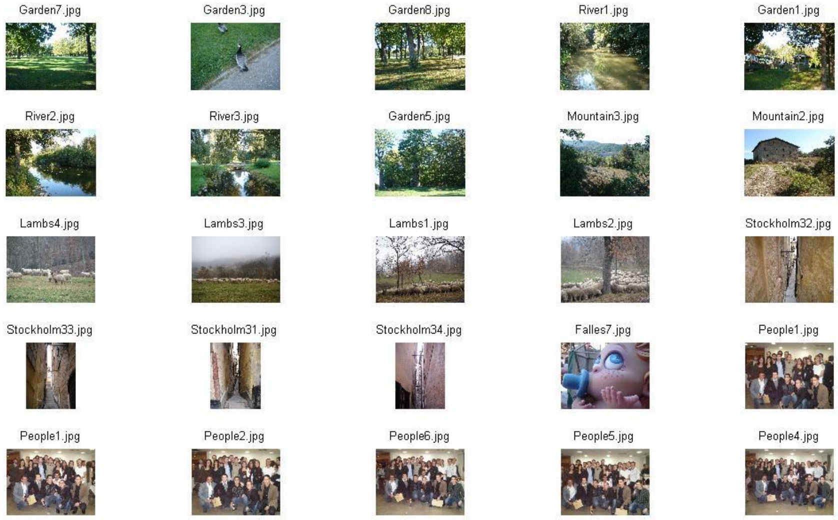

Fig. 17 Person 1: 76 – 100

B.4.2. Person 2 arrangement

1	Mountain1	21	Stockholm7	41	Stockholm29	61	Trial6	81	Lambs3
2	Beach2	22	Stockholm6	42	Stockholm28	62	Stockholm33	82	Lambs2
3	Mountain6	23	Stockholm22	43	Stockholm30	63	People1	83	Lambs4
4	Beach3	24	Stockholm20	44	Stockholm27	64	People3	84	Mountain3
5	Falles1	25	Stockholm18	45	Stockholm25	65	People5	85	Mountain2
6	Falles6	26	Stockholm11	46	Snow1	66	People6	86	Garden5
7	Beach1	27	Stockholm14	47	Snow3	67	People4	87	Garden8
8	Sea4	28	Stockholm15	48	Snow2	68	People2	88	Garden4
9	Sea5	29	Stockholm21	49	Stockholm17	69	Trial1	89	Garden6
10	Sea1	30	Stockholm23	50	Stockholm2	70	Trial5	90	Garden1
11	Sea3	31	Stockholm8	51	Stockholm5	71	Trial4	91	Stadium1
12	Sea2	32	Stockholm12	52	Sunset2	72	Trial2	92	Stadium4
13	Sea7	33	Stockholm4	53	Sunset1	73	Trial3	93	Stadium3
14	Sea6	34	Stockholm26	54	Sunset3	74	Falles2	94	River1
15	Mountain10	35	Stockholm3	55	Stockholm9	75	Falles7	95	River2
16	Mountain5	36	Stockholm1	56	Sunset4	76	Falles4	96	River3
17	Mountain7	37	Stockholm10	57	Sunset5	77	Falles3	97	Garden7
18	Mountain9	38	Stockholm16	58	Stockholm34	78	Falles5	98	Garden3
19	Mountain8	39	Stockholm13	59	Stockholm31	79	Mountain4	99	Garden2
20	Stockholm19	40	Stockholm24	60	Stockholm32	80	Lambs1	100	Stadium2

Fig. 18 Person 2 classification

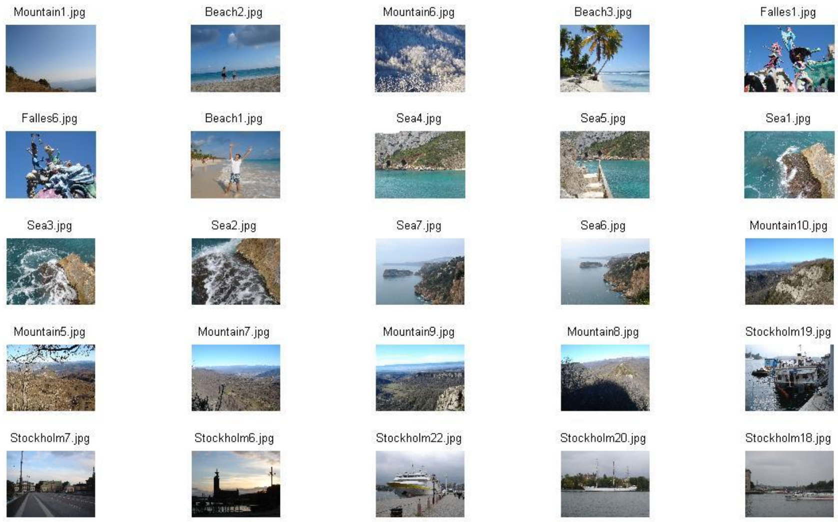

Fig. 19 Person 2: 1 – 25

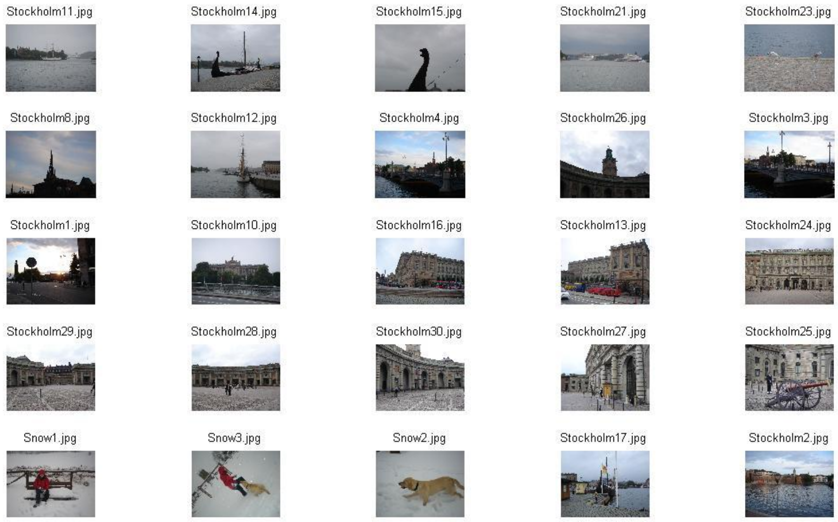

Fig. 20 Person 2: 26 – 50

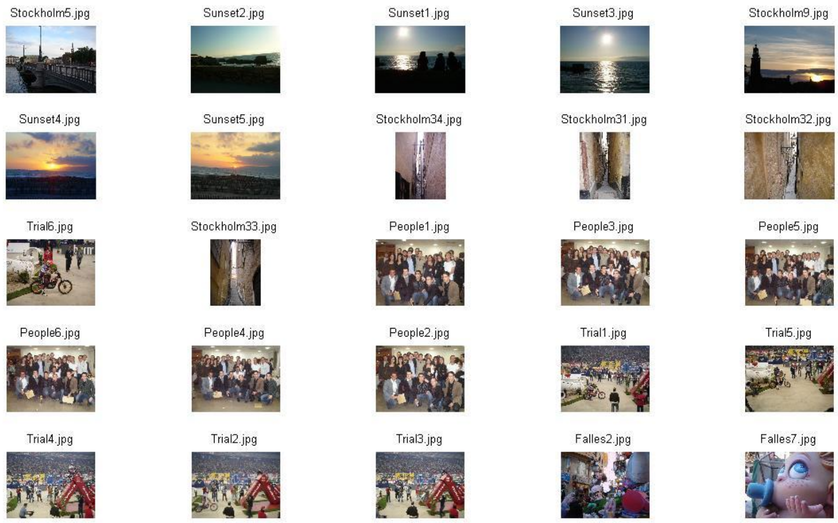

Fig. 21 Person 2: 51 – 75

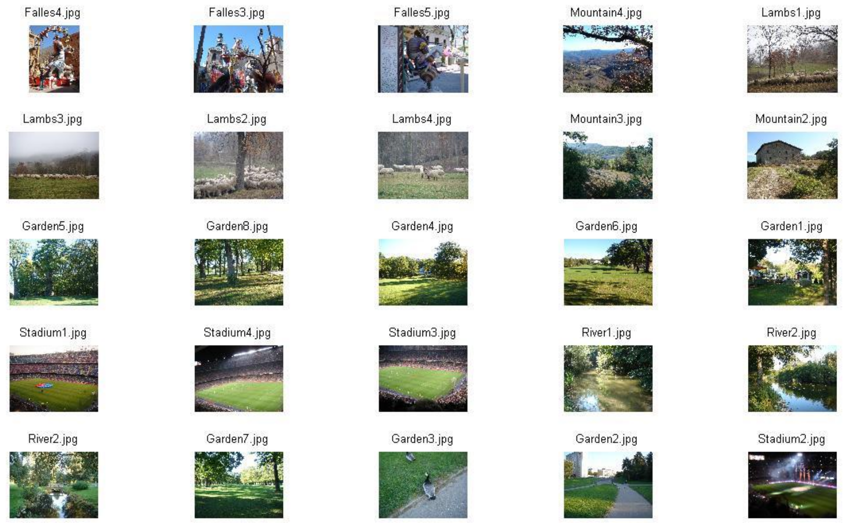

Fig. 22 Person 2: 76 – 100

B.4.3. Person 3 arrangement

1	Mountain1	**21**	Stockholm7	**41**	Stockholm8	**61**	Stadium3	**81**	People4
2	Beach2	**22**	Stockholm5	**42**	Stockholm6	**62**	Stadium4	**82**	Trial6
3	Mountain10	**23**	Stockholm4	**43**	Stockholm3	**63**	Stadium1	**83**	People6
4	Mountain7	**24**	Stockholm26	**44**	Stockholm1	**64**	Stadium2	**84**	People1
5	Beach3	**25**	Sea7	**45**	Sunset1	**65**	River3	**85**	People2
6	Beach1	**26**	Sea6	**46**	Sunset3	**66**	Garden8	**86**	People5
7	Falles1	**27**	Stockholm14	**47**	Sunset4	**67**	Garden3	**87**	People3
8	Falles6	**28**	Stockholm21	**48**	Sunset5	**68**	Mountain2	**88**	Trial5
9	Mountain6	**29**	Stockholm23	**49**	Stockholm9	**69**	Lambs2	**89**	Trial4
10	Sea3	**30**	Stockholm15	**50**	Lambs3	**70**	Lambs4	**90**	Trial1
11	Sea1	**31**	Stockholm17	**51**	Lambs1	**71**	Stockholm25	**91**	Trial2
12	Sea5	**32**	Stockholm20	**52**	Mountain3	**72**	Stockholm24	**92**	Trial3
13	Sea4	**33**	Stockholm22	**53**	River1	**73**	Stockholm30	**93**	Snow1
14	Sunset2	**34**	Stockholm10	**54**	Garden5	**74**	Stockholm27	**94**	Falles4
15	Mountain9	**35**	Stockholm16	**55**	River2	**75**	Stockholm32	**95**	Stockholm13
16	Mountain4	**36**	Stockholm12	**56**	Garden1	**76**	Stockholm33	**96**	Falles5
17	Mountain8	**37**	Stockholm18	**57**	Garden2	**77**	Stockholm34	**97**	Stockholm19
18	Sea2	**38**	Stockholm11	**58**	Garden6	**78**	Stockholm31	**98**	Falles7
19	Mountain5	**39**	Stockholm28	**59**	Garden4	**79**	Snow3	**99**	Falles3
20	Stockholm2	**40**	Stockholm29	**60**	Garden7	**80**	Snow2	**100**	Falles2

Fig. 23 Person 3 classification

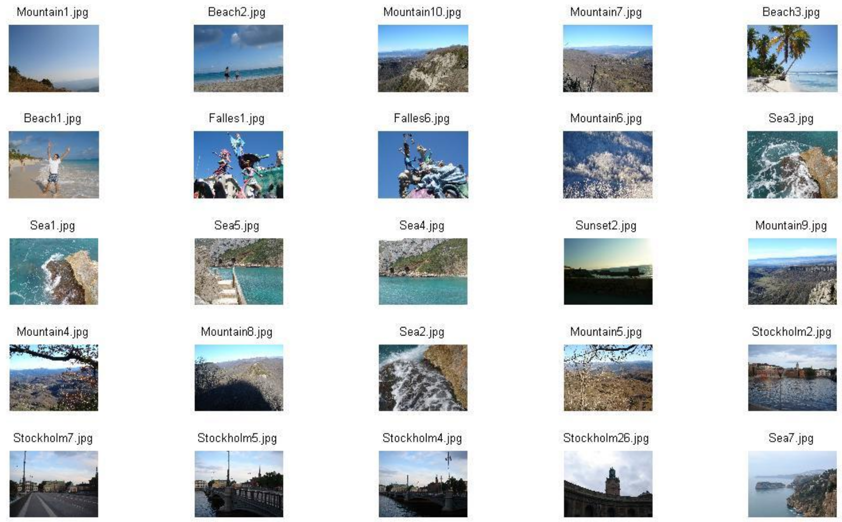

Fig. 24 Person 3: 1 – 25

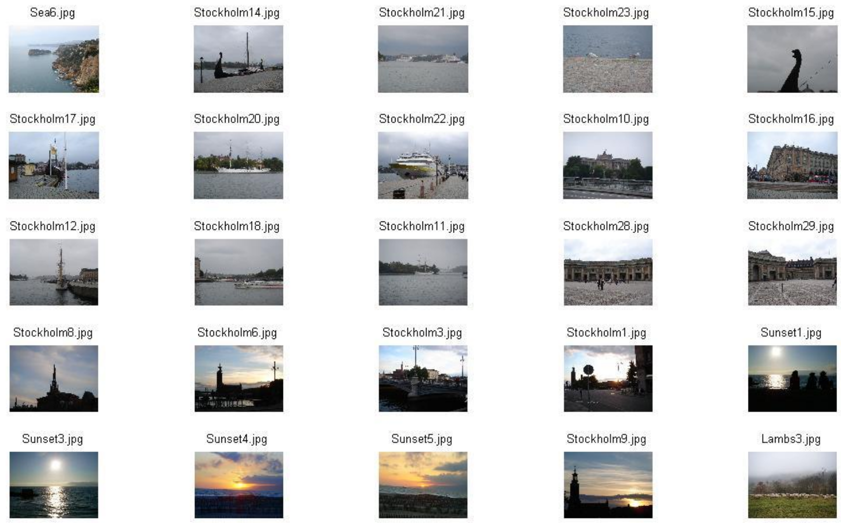

Fig. 25 Person 3: 26 – 50

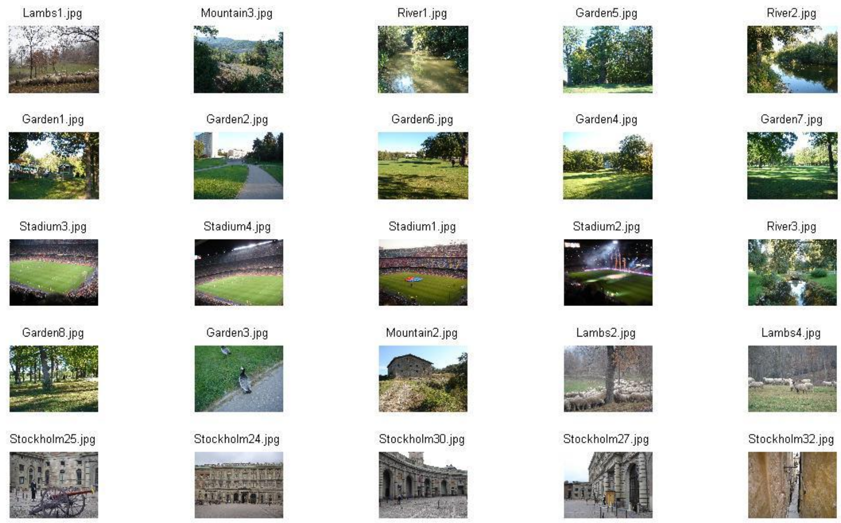

Fig. 26 Person 3: 51 – 75

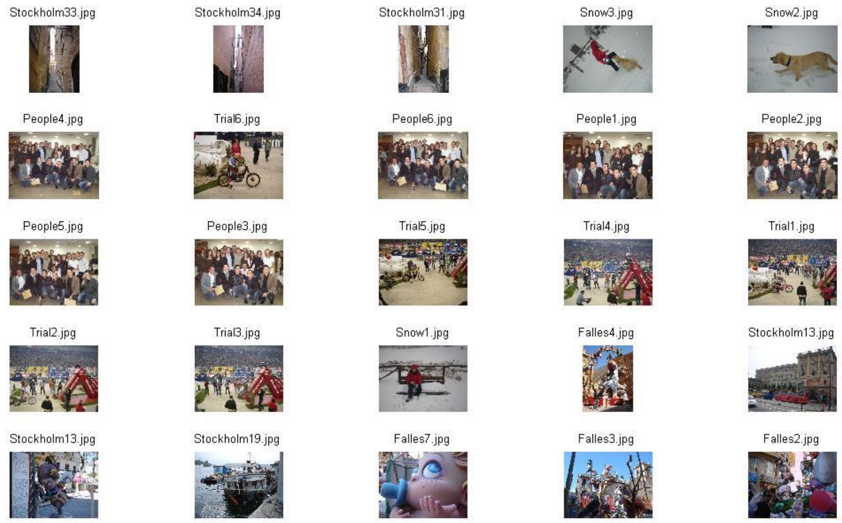

Fig. 27 Person 3: 76 – 100

B.5. Analysis

The following figures compare the results of the arrangements of the descriptors with the arrangements carried out for the three people. The meaning of the tables is the following:

- The numeration from 1 to 100 (in gray) indicates the classification of the person.
- Each cell number corresponds to the result compared with the CLD, DCD or CLD&DCD. For example, if in the position 4 appears the 82 number, this means that for the CLD classification the *Beach1* corresponds to the position 82 whereas for the person 1 corresponds to position 4.
- In red color are marked the coincidences and in orange color the images that are in the middle of a series of coincidences.

B.5.1. CLD comparison

1	1	21	23	41	52	61	81	81	100
2	4	22	65	42	54	62	59	82	58
3	99	23	22	43	91	63	60	83	92
4	82	24	26	44	18	64	33	84	69
5	5	25	25	45	17	65	46	85	95
6	84	26	27	46	16	66	2	86	48
7	30	27	72	47	14	67	3	87	28
8	70	28	24	48	6	68	35	88	29
9	32	29	64	49	74	69	90	89	57
10	68	30	45	50	73	70	96	90	37
11	31	31	21	51	89	71	93	91	36
12	67	32	20	52	85	72	98	92	38
13	66	33	19	53	77	73	78	93	76
14	11	34	62	54	87	74	94	94	83
15	13	35	15	55	88	75	97	95	39
16	7	36	63	56	71	76	75	96	41
17	9	37	49	57	34	77	47	97	40
18	8	38	50	58	86	78	79	98	44
19	10	39	51	59	56	79	61	99	42
20	12	40	53	60	80	80	55	100	43

Fig. 28 CLD: Person 1 classification

With these results, it is observed that for the classification done by the person 1 there are 49% of coincidences compared with the CLD classification.

1	1	21	25	41	21	61	54	81	28
2	4	22	73	42	20	62	36	82	57
3	5	23	11	43	45	63	39	83	48
4	99	24	10	44	62	64	41	84	69
5	2	25	8	45	15	65	42	85	95
6	3	26	9	46	18	66	44	86	92
7	82	27	12	47	17	67	43	87	79
8	59	28	14	48	16	68	40	88	94
9	60	29	7	49	23	69	52	89	97
10	81	30	6	50	22	70	53	90	55
11	80	31	74	51	27	71	51	91	96
12	56	32	13	52	71	72	49	92	98
13	67	33	26	53	86	73	50	93	78
14	66	34	89	54	34	74	35	94	61
15	32	35	72	55	77	75	83	95	100
16	30	36	85	56	87	76	91	96	58
17	31	37	24	57	88	77	33	97	75
18	68	38	64	58	76	78	46	98	47
19	70	39	63	59	38	79	84	99	93
20	65	40	19	60	37	80	29	100	90

Fig. 29 CLD: Person 2 classification

In this case there are a 55% of coincidences compared with the CLD classification.

1	1	21	25	41	74	61	78	81	43
2	4	22	27	42	73	62	98	82	54
3	32	23	26	43	72	63	96	83	44
4	31	24	89	44	85	64	90	84	39
5	99	25	67	45	86	65	58	85	40
6	82	26	66	46	34	66	79	86	42
7	2	27	12	47	87	67	47	87	41
8	3	28	7	48	88	68	95	88	53
9	5	29	6	49	77	69	57	89	51
10	80	30	14	50	28	70	48	90	52
11	81	31	23	51	29	71	15	91	49
12	60	32	10	52	69	72	19	92	50
13	59	33	11	53	61	73	45	93	18
14	71	34	24	54	92	74	62	94	91
15	68	35	64	55	100	75	37	95	63
16	84	36	13	56	55	76	36	96	46
17	70	37	8	57	93	77	76	97	65
18	56	38	9	58	97	78	38	98	83
19	30	39	20	59	94	79	17	99	33
20	22	40	21	60	75	80	16	100	35

Fig. 30 CLD: Person 3 classification

Observing these results, there are 47% of coincidences compared with the CLD classification.

B.5.2. DCD comparison

1	1	21	21	41	48	61	88	81	75
2	33	22	45	42	59	62	86	82	91
3	81	23	20	43	58	63	87	83	92
4	32	24	19	44	9	64	44	84	77
5	80	25	41	45	100	65	34	85	67
6	35	26	40	46	7	66	95	86	53
7	93	27	18	47	8	67	96	87	55
8	37	28	16	48	30	68	43	88	56
9	28	29	15	49	31	69	63	89	29
10	52	30	36	50	62	70	98	90	94
11	84	31	13	51	17	71	78	91	83
12	69	32	14	52	42	72	46	92	57
13	68	33	85	53	60	73	47	93	99
14	54	34	11	54	38	74	72	94	97
15	3	35	10	55	39	75	82	95	25
16	6	36	12	56	64	76	71	96	24
17	5	37	50	57	79	77	70	97	23
18	4	38	49	58	61	78	73	98	65
19	90	39	51	59	22	79	76	99	26
20	2	40	66	60	89	80	74	100	27

Fig. 31 DCD: Person 1 classification

It is observed that there are 56% of coincidences compared with the CLD classification.

1	1	21	41	41	13	61	59	81	55
2	33	22	62	42	14	62	83	82	29
3	80	23	54	43	36	63	25	83	53
4	81	24	90	44	11	64	24	84	77
5	95	25	4	45	10	65	26	85	67
6	96	26	5	46	9	66	65	86	92
7	32	27	2	47	100	67	27	87	73
8	86	28	8	48	7	68	23	88	72
9	87	29	6	49	21	69	48	89	82
10	88	30	30	50	20	70	66	90	74
11	89	31	31	51	40	71	51	91	98
12	22	32	3	52	64	72	50	92	46
13	69	33	19	53	61	73	49	93	47
14	68	34	17	54	79	74	43	94	76
15	28	35	18	55	60	75	97	95	75
16	93	36	42	56	38	76	58	96	91
17	84	37	16	57	39	77	44	97	71
18	52	38	15	58	99	78	34	98	70
19	37	39	12	59	57	79	35	99	78
20	45	40	85	60	94	80	56	100	63

Fig. 32 DCD: Person 2 classification

In this case there are a 51% of coincidences compared with the CLD classification.

1	1	**21**	**41**	**41**	31	**61**	**47**	**81**	27
2	33	**22**	**40**	**42**	62	**62**	**46**	**82**	59
3	28	**23**	19	**43**	18	**63**	98	**83**	65
4	84	**24**	17	**44**	42	**64**	63	**84**	**25**
5	81	**25**	**69**	**45**	61	**65**	91	**85**	**23**
6	32	**26**	**68**	**46**	79	**66**	73	**86**	**26**
7	**95**	**27**	2	**47**	**38**	**67**	70	**87**	**24**
8	**96**	**28**	6	**48**	**39**	**68**	67	**88**	66
9	80	**29**	30	**49**	60	**69**	29	**89**	**51**
10	**89**	**30**	8	**50**	**55**	**70**	53	**90**	**48**
11	**88**	**31**	21	**51**	**56**	**71**	10	**91**	**50**
12	**87**	**32**	90	**52**	**77**	**72**	85	**92**	**49**
13	**86**	**33**	54	**53**	**76**	**73**	36	**93**	9
14	64	**34**	**16**	**54**	92	**74**	11	**94**	58
15	52	**35**	**15**	**55**	**75**	**75**	94	**95**	12
16	35	**36**	**3**	**56**	**74**	**76**	83	**96**	34
17	37	**37**	**4**	**57**	**78**	**77**	99	**97**	**45**
18	22	**38**	**5**	**58**	82	**78**	57	**98**	97
19	93	**39**	**14**	**59**	**72**	**79**	100	**99**	**44**
20	20	**40**	**13**	**60**	**71**	**80**	7	**100**	**43**

Fig. 33 DCD: Person 3 classification

Observing these results, for person 3 can be extracted a total of 40% of coincidences compared with the CLD classification.

Underline that whereas person 1 gets a 49% of coincidences compared with CLD, with the comparison between DCD the same person obtains a 56%. This means that for person 1 there is more important the dominant colors than the color structure.

Finally, underline that with the tests carried out in this chapter there are more coincidences with the CLD than with the DCD (50.33% vs. 49%). Emphasize that the difference is very small but maybe this is due to persons tends to follow more the structure of the images than the dominant colors but, for example, for person 1 it’s the opposite.

B.5.3. CLD&DCD comparison

1	1	21	20	41	40	61	86	81	82
2	57	22	50	42	88	62	85	82	70
3	100	23	21	43	89	63	84	83	99
4	2	24	17	44	7	64	60	84	76
5	56	25	15	45	8	65	54	85	97
6	91	26	16	46	6	66	59	86	36
7	44	27	18	47	5	67	58	87	46
8	64	28	14	48	3	68	53	88	45
9	61	29	47	49	51	69	73	89	35
10	63	30	81	50	74	70	96	90	43
11	62	31	25	51	19	71	77	91	34
12	49	32	24	52	92	72	67	92	33
13	48	33	26	53	65	73	66	93	72
14	23	34	80	54	94	74	78	94	90
15	12	35	9	55	95	75	98	95	32
16	4	36	79	56	75	76	71	96	30
17	10	37	37	57	52	77	55	97	31
18	11	38	38	58	93	78	68	98	29
19	22	39	39	59	42	79	83	99	27
20	13	40	41	60	87	80	69	100	28

Fig. 34 CLD&DCD: Person 1 classification

For person 1, it is observed a 51% of coincidences compared with the CLD%DCD classification.

1	1	21	15	41	25	61	88	81	46
2	57	22	74	42	24	62	34	82	35
3	56	23	23	43	81	63	32	83	36
4	100	24	22	44	80	64	30	84	76
5	59	25	11	45	9	65	27	85	97
6	58	26	10	46	7	66	29	86	99
7	2	27	13	47	8	67	28	87	68
8	85	28	5	48	6	68	31	88	78
9	84	29	4	49	20	69	40	89	98
10	86	30	3	50	21	70	41	90	69
11	87	31	51	51	16	71	39	91	96
12	42	32	12	52	75	72	37	92	67
13	49	33	17	53	93	73	38	93	66
14	48	34	19	54	52	74	53	94	83
15	61	35	18	55	65	75	90	95	82
16	44	36	92	56	94	76	89	96	70
17	62	37	14	57	95	77	60	97	71
18	63	38	47	58	72	78	54	98	55
19	64	39	79	59	33	79	91	99	77
20	50	40	26	60	43	80	45	100	73

Fig. 35 CLD&DCD: Person 2 classification

In this case, the coincidences reach the 60%, compared with the CLD&DCD classification.

1	1	21	15	41	51	61	66	81	28
2	57	22	16	42	74	62	67	82	88
3	61	23	17	43	18	63	96	83	29
4	62	24	19	44	92	64	73	84	32
5	100	25	49	45	93	65	70	85	31
6	2	26	48	46	52	66	68	86	27
7	59	27	13	47	94	67	55	87	30
8	58	28	4	48	95	68	97	88	41
9	56	29	3	49	65	69	35	89	39
10	87	30	5	50	46	70	36	90	40
11	86	31	20	51	45	71	9	91	37
12	84	32	22	52	76	72	26	92	38
13	85	33	23	53	83	73	81	93	7
14	75	34	14	54	99	74	80	94	89
15	63	35	47	55	82	75	43	95	79
16	91	36	12	56	69	76	34	96	54
17	64	37	11	57	77	77	72	97	50
18	42	38	10	58	98	78	33	98	90
19	44	39	24	59	78	79	8	99	60
20	21	40	25	60	71	80	6	100	53

Fig. 36 CLD&DCD: Person 3 classification

In this case there are a 46% of coincidences compared with the CLD&DCD classification.

Once the three CLD&DCD comparisons have been carried out, it can be extracted that there are an average of 52.33% of coincidences between the people classifications and the one done by d the CLD%DCD. Point out that person 2 gets a 60% of coincidences compared with CLD&DCD, which is the higher result.

Finally, comparing these results with the CLD and DCD ones, the union of CLD&DCD gets better results in this particular test than the CLD or DCD alone.

ANNEX C. TESTS

All the tests performed during the execution of this project are presented below:

C.1. Tests

Fig. 1 Test 1

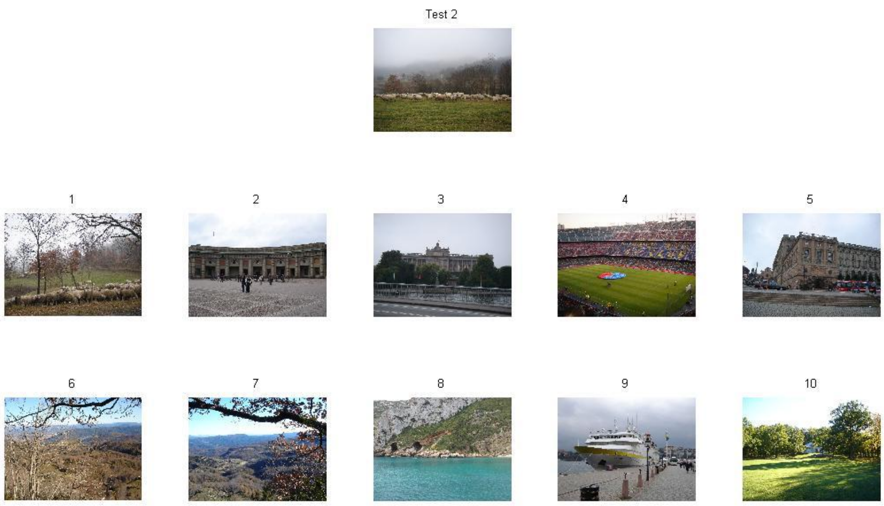

Fig. 2 Test 2

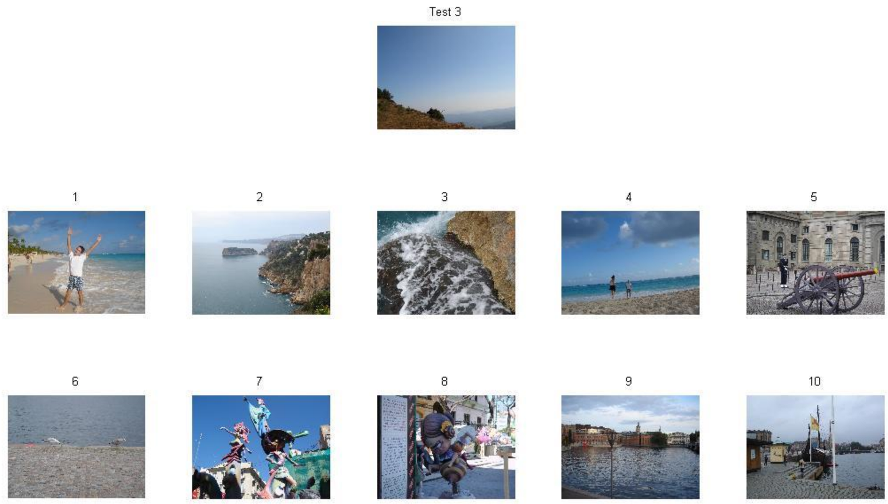

Fig. 3 Test 3

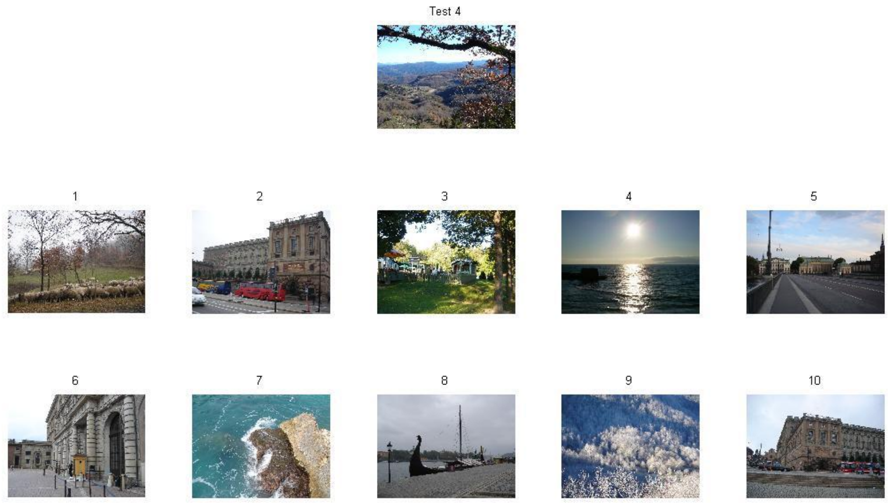

Fig. 4 Test 4

Fig. 5 Test 5

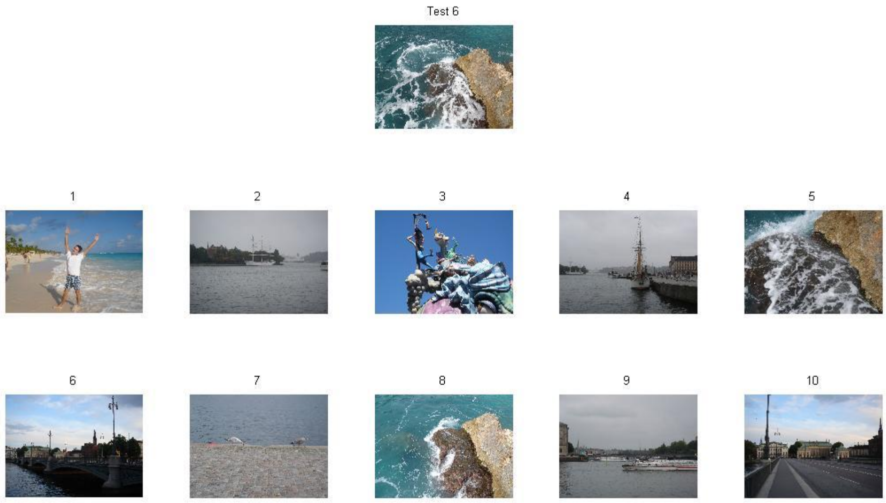

Fig. 6 Test 6

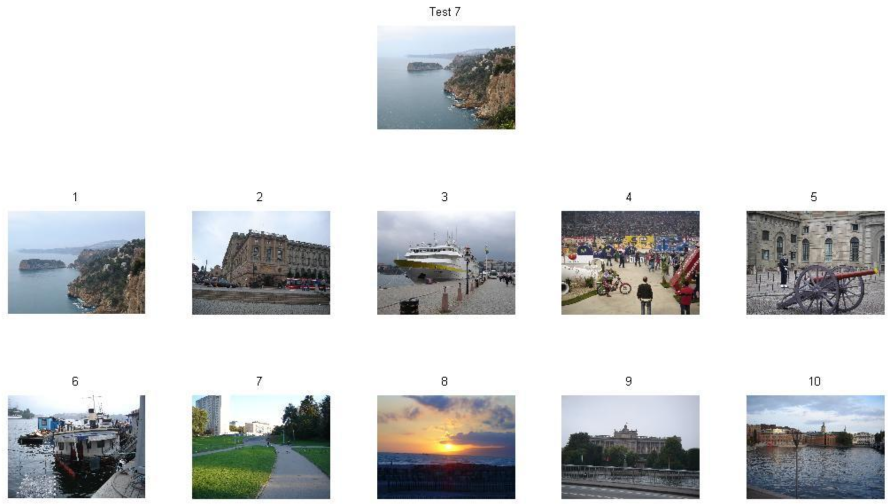

Fig. 7 Test 7

Fig. 8 Test 8

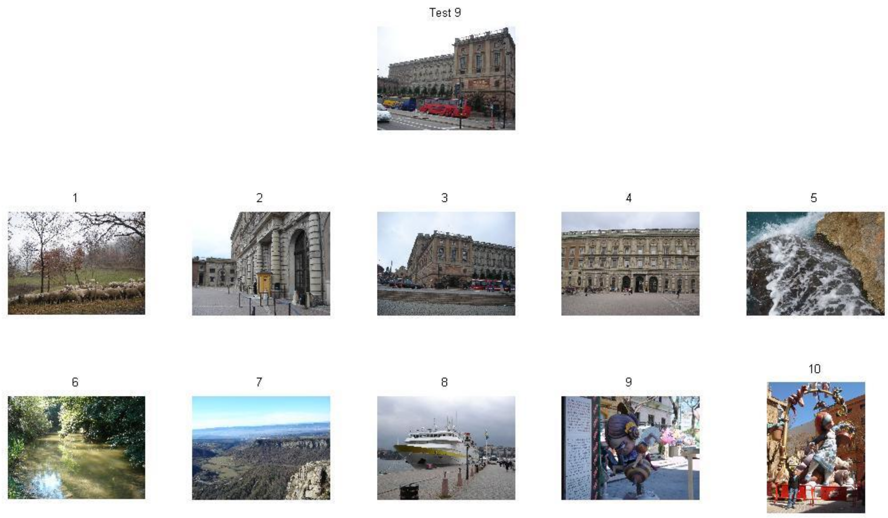

Fig. 9 Test 9

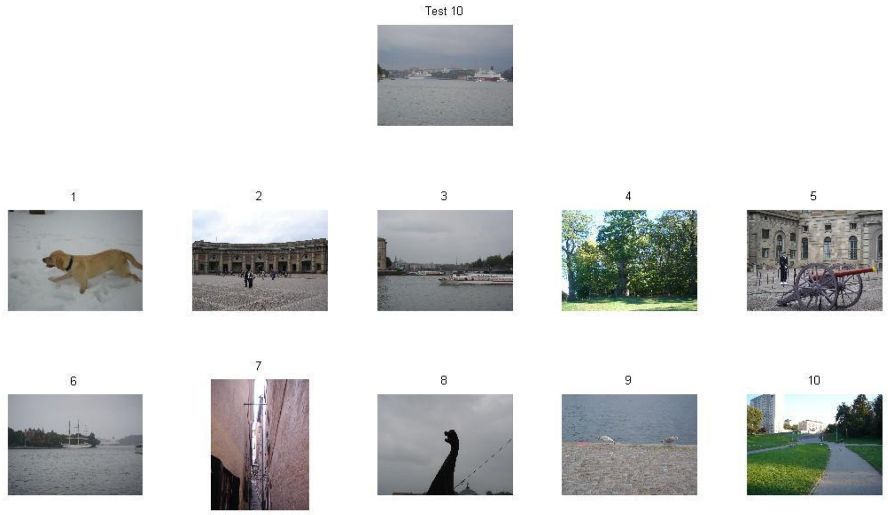

Fig. 10 Test 10

Fig. 11 Test 11

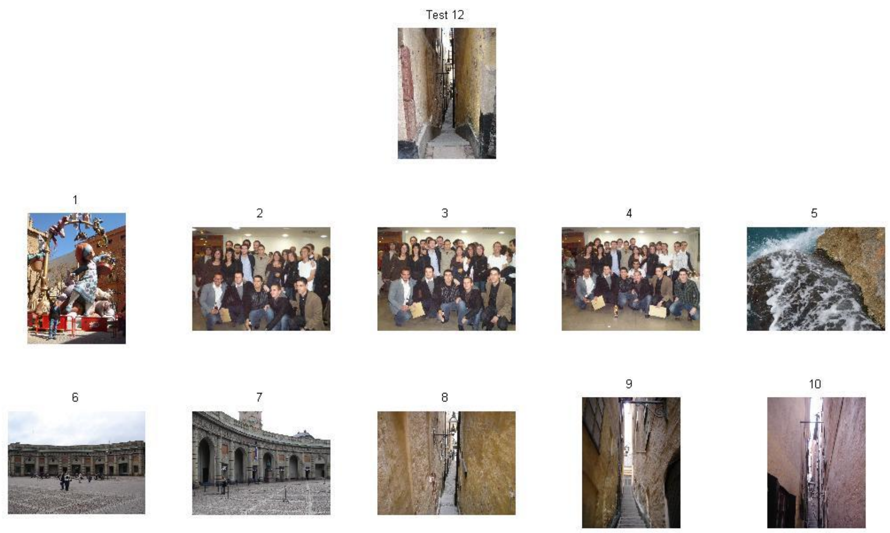

Fig. 12 Test 12

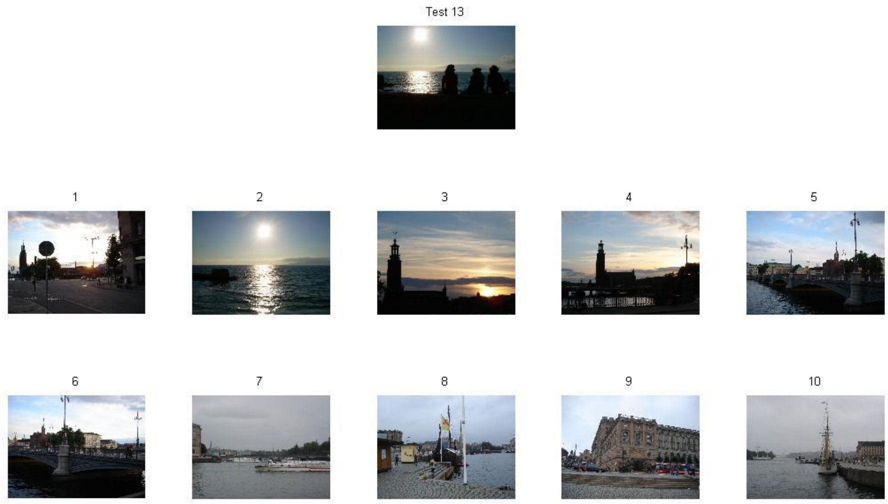

Fig. 13 Test 13

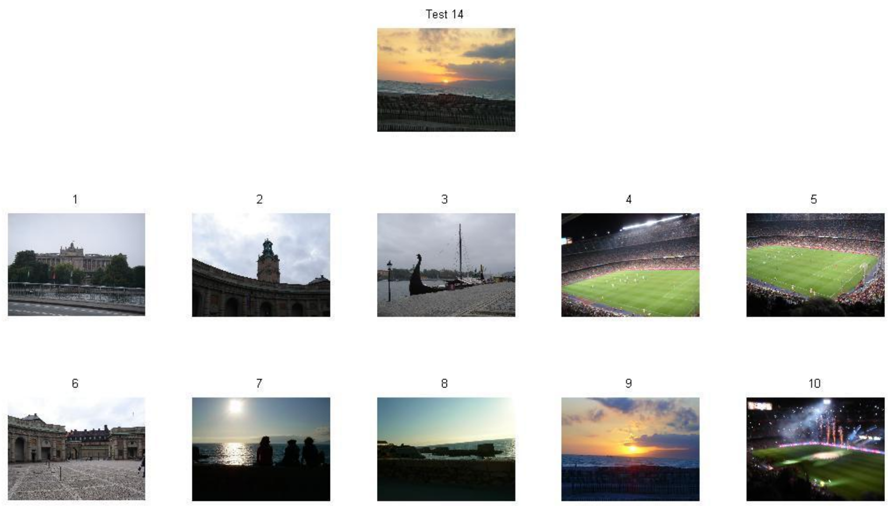

Fig. 14 Test 14

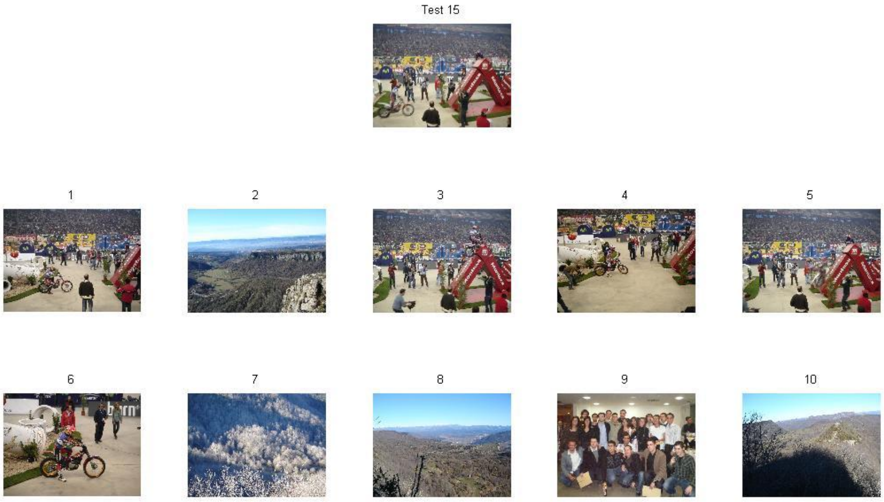

Fig. 15 Test 15

ANNEX D. RESULTS

This ANNEX is linked to an EXCEL spreadsheet file called *Evaluation.xls*. The structure of this EXCEL is the following:

- Sheet name: ***People Marks and Final Score***
  - In this sheet is presented the people marks corresponding to the 15 different tests. Also it can be observed the final score, with all the marks added.
- Sheet name: ***Tests People***
  - The selection performed by the people of the different tests
- Sheet name: ***Top 3 Descriptors***
  - The 3 images with less distance from each descriptor using the reference image of each test
- Sheet name: ***Test Order***
  - The order of the images of each tests