Department of Computer Science
Technical University of Cluj-Napoca

Image Processing
Content-Based Image Retrieval

Name: Paul-Adrian Kovacs
Group: 30232
Email: kovacs_paul_adrian@yahoo.com

March 2023

Contents

1 Introduction

1.1 Background

With the increasing availability of digital images, there is a growing need for efficient and accurate methods to search and retrieve images based on their content. Content-Based Image Retrieval (CBIR) is one approach that uses features such as color, texture, and shape to find similar images. However, there are still challenges in developing CBIR systems that can handle large datasets and provide accurate results.

1.2 Problem Statement

The main problem that this project seeks to address is the limited performance of CBIR systems when dealing with large-scale image datasets. Existing CBIR methods may not be able to accurately and efficiently search through a large number of images based on their content due to the high computational cost of feature extraction and matching algorithms, as well as the difficulty of dealing with variations in image quality and noise.

1.3 Significance

Developing an image retrieval system that accurately retrieves images based on their visual features has significant practical applications in fields such as medical imaging, surveillance, and data mining. The proposed system has the potential to save time and improve efficiency by enabling users to easily find similar images in large datasets.

1.4 Objectives

The specific objectives of this project are to (1) develop an image retrieval system that uses color histograms and Pearson correlation coefficient, (2) evaluate the performance of the system on a large dataset of color images, and (3) propose improvements to the system to enhance its accuracy and scalability.

1.5 Relevance

This project is highly relevant to the field of computer science, specifically in the area of image processing and retrieval. The results of this project will be useful to researchers and industry professionals who are working on CBIR systems and related applications.

1.6 Scope

This project will focus specifically on the development and evaluation of an image retrieval system using color histograms and Pearson correlation coefficient. The system will use a set of images, given by the user at initialization, as a database and will allow users to select an input image to find the most similar images in the database. The system will also calculate the histograms on the Hue channel of the HSI color space for the selected images and display them along with their correlation coefficients. The project will not cover other image processing techniques.

2 Bibliographic Study

2.1 CBIR - Content-based image retrieval

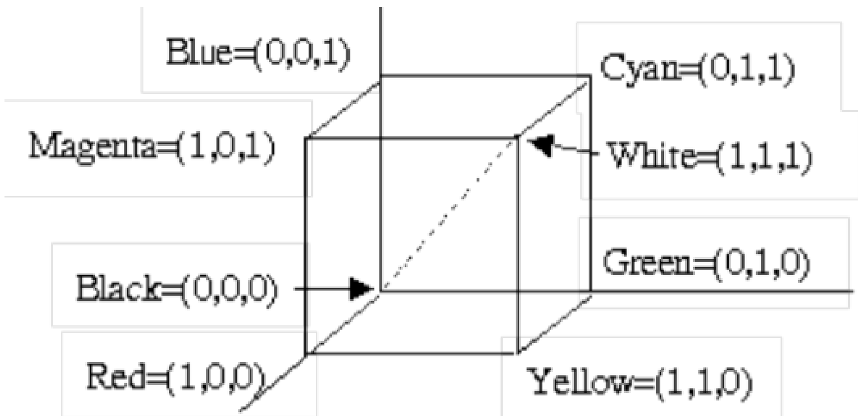
Content-based image retrieval is an approach in which similar images are retrieved for a particular query image depending on the image content similarity. One of the challenges of CBIR is to determine the similarity between color images, which may vary in hue, saturation and brightness. A common approach to measure color similarity is to use color histograms, which represent the distribution of colors in an image.

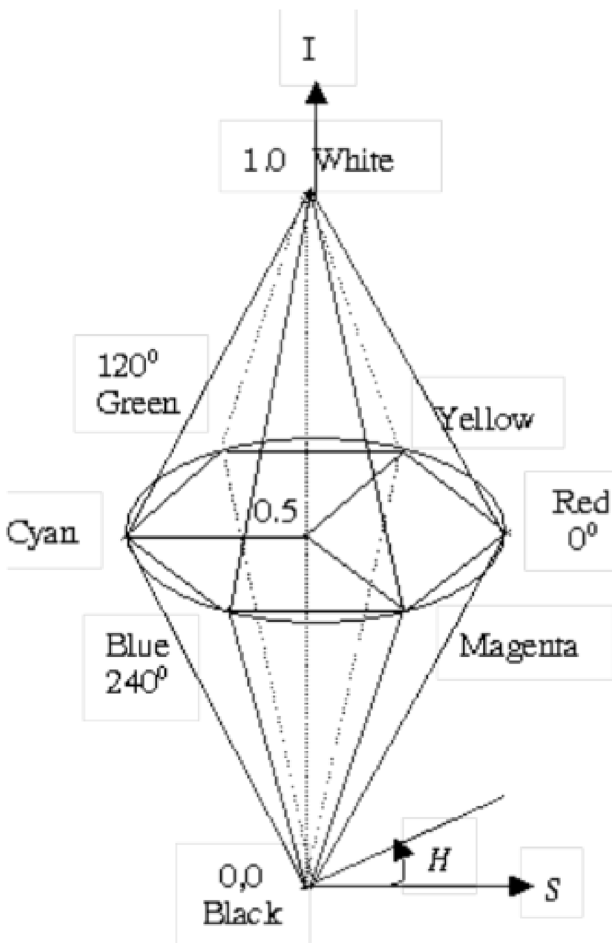
Color is one of the key visual features that can be used to determine the similarity between images. In content-based image retrieval, color similarity between images can be analyzed by examining various aspects of the color distribution in the images, such as color histograms, color correlograms, and color coherence vectors. These features can be compared using various distance measures, such as Euclidean distance, Manhattan distance, or cosine similarity, to determine the similarity between the color images.

2.2 Color spaces[1]

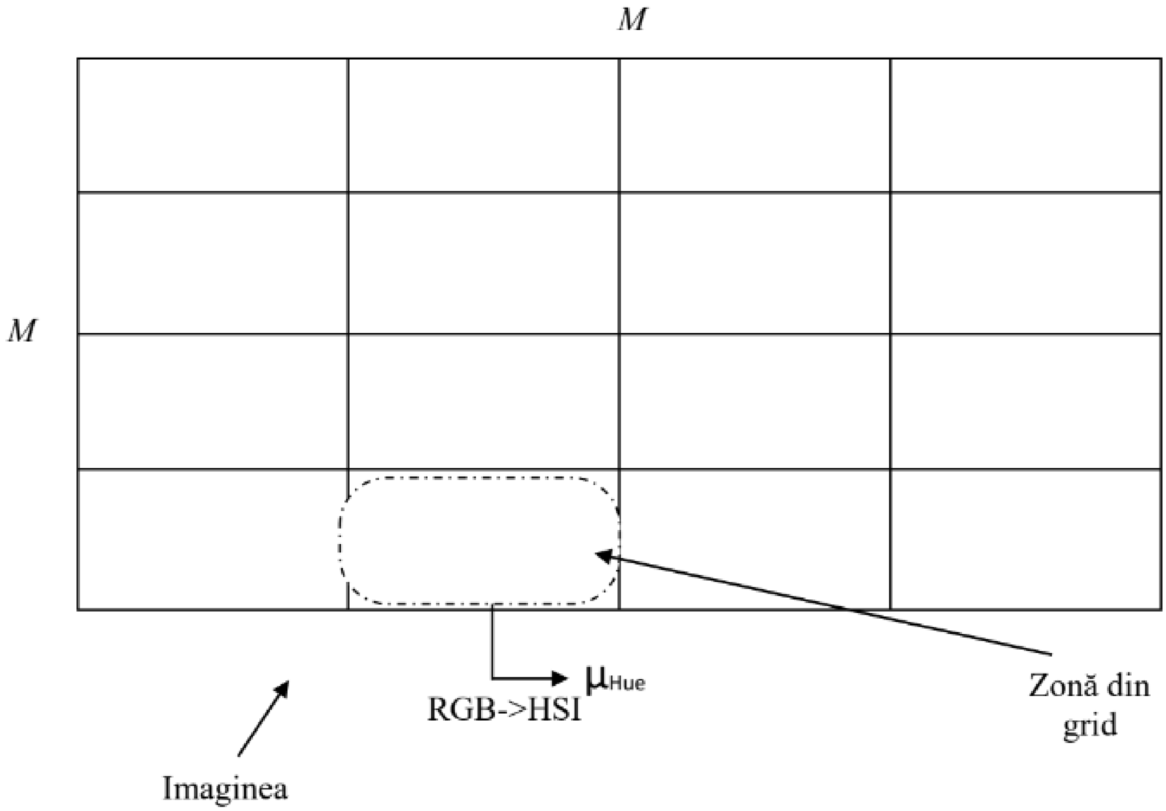
$$I = \frac{R+G+B}{3}; S = \left(I \neq 0 ? 1 - \frac{\min(R,G,B)}{I} : 0\right)$$

$$h = \left(S \neq 0 ? \operatorname{acos} \frac{(R-G)+(R-B)}{2\sqrt{(R-G)^2+(R-B)(G-B)}} : 0\right); H = (B > G ? 360° - h : h)$$





2.3 Determining similarity based on color distribution[2]



The image will be divided into an M*M grid to increase accuracy.

On each grid zone, an RGB -> HSI conversion is performed and the average Hue value ($\mu_{Hue} \in [0, 2\pi)$) is calculated for that zone, ignoring points for which S = 0. This value will be stored in a vector V of size M*M. If all points in a grid zone have S = 0, then the average value is meaningless, and the corresponding element in V will be assigned the value -1, indicating that it should be ignored in subsequent calculations.

The resulting vector V obtained by traversing all zones will be the image representation vector.

If we want to see the similarity between this image and a new image, the corresponding V vector for the new image will be calculated. The degree of differentiation between the two images is given by the formula:

$$D = \frac{1}{C} * \sqrt{\sum_{i=1}^{M*M} (min(|V_1[i] - V_2[i]|, 2\pi - |V_1[i] - V_2[i]|)^2,\ if\ V_1[i] \geq 0\ and\ V_2[i] \geq 0)}$$

where:
V_1, V_2 - the arrays corespoding to the 2 images
$C = \sqrt{N}\pi$ - normalization constant such that D has values in the range [0,1]
V_1, V_2 - number of pairs i which met $V_1[i] \geq 0$

The degree of similarity between the two images is given by the formula:

$$S = 1 - D, if N > 0 or S = 0, if N = 0$$

If a percentage representation is used, S will be multiplied by 100. A similarity threshold T (in percentage) can be chosen:

$$S > T => the\ images\ are\ similar$$

$$S <= T => the\ images\ are\ not\ similar.$$

2.4 Gaussian filter[3]

Gaussian noise removal must be performed using a filter with adequate shape and size, correlated to the amount of the Gaussian noise that corrupts the image. The filter size w of such a filter is usually 6σ (for example, for a Gaussian noise with $\sigma = 0.8 => w = 4.8 \approx 5$ Constructing the elements of such a kernel/Gaussian filter G will be performed using the following equation:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x-x_0)^2+(y-y_0)^2}{2\sigma^2}}$$

where:
X_0, Y_0 - are the coordinates of the central column and row of the kernel
σ - standard deviation

2.5 Histograms[4]

Many applications use the HSI color model. Machine vision uses HSI color space in identifying the color of different objects. Image processing applications, such as histogram operations, intensity transformations, and convolutions, operate on only an image's intensity. These operations are performed much easier on an image in the HSI color space. For the HSI is modeled with cylindrical coordinates. The hue (H) is represented as the angle, varying from 0o to 360o. Saturation (S) corresponds to the radius, varying from 0 to 1. Intensity (I) varies along the z axis with 0 being black and 1 being white.

2.6 Pearson correlation coefficient[5]

In statistics, the Pearson correlation coefficient is a measure of linear correlation between two sets of data. It is the ratio between the covariance of two variables and the product of their standard deviations; thus, it is essentially a normalized measurement of the covariance, such that the result always has a value between -1 and 1. As with covariance itself, the measure can only reflect a linear correlation of variables, and ignores many other types of relationships or correlations.

$$\rho_{X,Y} = \frac{\operatorname{cov}(X,Y)}{\sigma_X \sigma_Y}$$

where

- cov is the covariance
- σ_X is the standard deviation of X
- σ_Y is the standard deviation of Y.

The formula for ρ can be expressed in terms of mean and expectation. Since

$$\operatorname{cov}(X,Y) = \mathbb{E}[(X-\mu_X)(Y-\mu_Y)],$$

the formula for ρ can also be written as

$$\rho_{X,Y} = \frac{\mathbb{E}[(X-\mu_X)(Y-\mu_Y)]}{\sigma_X \sigma_Y}$$

where

- σ_Y and σ_X are defined as above
- μ_X is the mean of X
- μ_Y is the mean of Y
- $\mathbb{E}$ is the expectation.

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i-\bar{y})^2}}$$

where

- n is sample size
- x_i, y_i are the individual sample points indexed with *i*
- $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$ (the sample mean); and analogously for $\bar{y}$.

Rearranging gives us this formula for r_{xy}:

$$r_{xy} = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{\sqrt{n\sum x_i^2 - \left(\sum x_i\right)^2}\ \sqrt{n\sum y_i^2 - \left(\sum y_i\right)^2}}.$$

3 Implementation

3.1 Converting images to vectors of predominant hues

3.2 Finding the most similar images

3.3 Conversion to HSI color space

3.4 Gaussian filter

3.5 Computing histogram

3.6 Pearson correlation coefficient

4 Conclusions

References

[1] https://drive.google.com/file/d/1n-kH2kb8LKC8hTaiCUWGsaMBhnS-dj74/view *accessed on March 23, 2023*

[2] https://drive.google.com/file/d/1d1_VbqRfohH9AArnJ93tFQyTNYgtk-KJ/view *accessed on March 23, 2023*

[3] https://users.utcluj.ro/ rdanescu/pi_c09.pdf *accessed on March 23, 2023*

[4] https://biblioteca.utcluj.ro/files/carti-online-cu-coperta/625-8.pdf *accessed on March 23, 2023*

[5] https://en.wikipedia.org/wiki/Pearson_correlation_coefficient *accessed on March 23, 2023*

Images:

[6] https://en.wikipedia.org/wiki/Pearson_correlation_coefficient *accessed on March 23, 2023*

Listings

List of Figures

List of Tables