M2 - TSI

UE34 Laboratory Report

Image & Signal Processing

Authors:
Arthur Scharf
Andreas Wenzel

February 3, 2017

1 Introduction - Image Processing for Earth Observation

Image processing and the analysis of image data plays an important role especially for Earth Observations, but also in other fields related to space applications that make use of images, e.g. optical navigation or optical attitude control systems. In this report, we present a thorough overview of examples for image processing techniques, especially applying examples in Matlab. We focus on colour spaces for images, fourier transforms of basic figures, erosion and delatation and classification.

1.1 Computer Vision

From a mathematical point of view, an image is an application that associates a value $I(u, v)$ to a pixel (u, v). Pixels are atomic elements arranged in rows and columns, such that the size of an image is given by the number of rows and the number of columns. Therefore, an image can be processed by using a matrix representation, where each matrix element (u, v) represents a pixel containing the value $I(u, v)$ of the pixel.. In the case of multispectral images, i.e. colour images like RGB images (see following chapter), an image is an application that associates saveral values, e.g. $I_1(u, v)$ $I_2(u, v)$ and $I_3(u, v)$ to a pixel (u, v), where the values I_i represent a different colour like red, green and blue in RGB images. Therefore, each matrix element (u, v) contains a vector of values. This matrix notation is the basic of image processing using Matlab.

2 Greyscale and Colour images

The understanding of colour images and the connection to their greyscale representation are fundamental for image analysis and image processing. We illustrate colour images in this section by considering the examples of RGB and HSV colour space. There are also other colour spaces like CYMK (cyan, yellow, magenta, black) or YUV, but we only consider the first two in our examples since these are the most frequently used ones and essential for the understanding of coloured images. CYMK and YUV are mainly based on the first two colour spaces.

2.1 Greyscale Images

Greyscale images are basic images where the value of each pixel e.g. is associated to a value between 0, which is black, to 1 for bright. An example including an optical illusion is presented in image 1. For this image, a 512×1024 matrix was created in Matlab, first being black by associating a zero to each element, but then was filled with increasing values from 0 to 1 along each line. in the middle of the image, a stripe of constant values

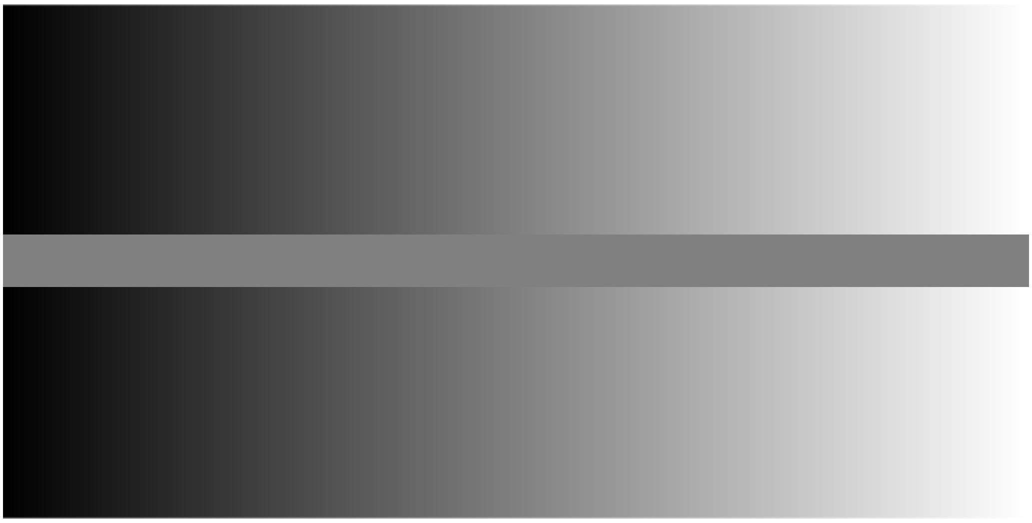

Figure 1: Greyscale image with optical illusion.

2.2 RGB

As mentioned before, in the case of a color image, 3 values are associated to each pixel in an RGB image, representing 8bit unsigned integer values (UINT8) for the colours of red, green and blue. By mixing these values for the colours, one can create any colour for the individual pixel. For the pixel to appear black, all the values for each pixel have to be 0, while they are 1 for a white image, which represents the additive nature of RGB images. As an example, we created the German flag by creating a 300×500 matrix, where the values for each pixel in the first 100 lines are set to 0, to 1 for I_R (red) and 0 for the other two in the lines from 101 to 200, and finally to $I_R = 250$,$I_G = 215$ and $I_B = 0$ in order to achieve a gold colour for the lower lines.

Figure 2: RGB image of the German flag.

2.3 HSV Colour Space

The HSV colour space consists of a value for hue (H), which is a value for the colour or wavelength running from 0 to 1, saturation (S) for that colour also running from 0 to 1, and the value (V) describing the darkness or brightness. To illustrate this, the image 3 was created. A matrix of 512×750 elements was created. For each line, the h value increases from left with 0 to right with 1, resulting in different colours for a spectrum along each line. For each column, the S value varies from 0 at the top of the image to 1 at the bottom. However, for all elements a brightness value of $V = 1$ was chosen.

Figure 3: Illustration of HSV colour space.

Figure 4: Illustration of HSV colour space in a colour wheel.

As an additional illustration, the HSV colour space is represented in the colour wheel shown in figure 4. Inside the wheel, the saturation gradually increases from inside to outside, while the hue value depends on the angle in the circle. Inside the circle, the brightness is chosen as a constant with $V = 1$, while it is set to 0 outside, resulting in the black frame around the circle.

2.4 Filtering

2.5 Image Processing & Analysis

2.6 Pattern Recognition

2.7 Geometry

3 Computer Vision & Morphology

3.1 Erosion & Dilatation

3.2 Morphological Filtering

3.3 Morphological Skeletonization & Segmentation

4 Data Analysis & Processing

After having extensively discussed Image Processing and Image Analysis in the previous chapters we will now move on to Data Analysis.
In general, Data Analysis is an extension or generalisation of Image Analysis - in particular, an image can also be seen as a set of data with a specific topology. However, data analysis is not limited to images but can be used for a vast amount of applications, be it business, science or others, where a set of data is given and information has to be extracted. Ultimately, this is the goal of data analysis: shaping, modelling and analysing the data to gain information or conclusions from given data. Since this course puts an emphasis on analysing and processing images, all the following examples will be performed on and explained by using images as a data set.

The term *data* itself can be interpreted and defined in multiple ways, which raises the need for a definition in our context of data analysis of images. In general data can be considered as an element taken from a set of data-elements. Each data-element can then be seen as a set of different components, attributes, parameters etc. - defining what the data-element is composed of - and are often called *descriptors*. Mathematically speaking, this means that our data $\boldsymbol{x}$ is associated to a vector in $\mathbb{R}^{\boldsymbol{n}}$ containing the descriptors,

$$x = (c_1 ... c_n)^T$$

which is also called the *state space* E of our data. These descriptors are the crucial part when analysing data, since they are defining a set of rules according to which we analyse the raw data.

If this terminology is applied to images, the pixels or a range of pixels in each image can be seen as such an descriptor, thus the image itself as our data-element.

4.1 Classification

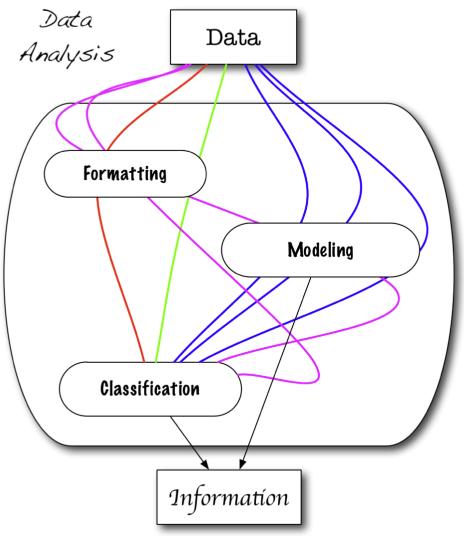

Figure 5: Steps for classification [1]

Having a data set that consists of descriptors essentially allows us to *label* this particular data set. This labelling of data is usually also described as *Classification*, meaning that each data-element or sub-data-element is assigned a particular *class*. In image analysis for space applications typically the pixels in an image have to be classified.

In general, classification of data consists of three steps,

1. Formatting
2. Modelling
3. Classification

whereas these steps are not compulsory. The formatting step usually consists of finding good descriptors, changing the state space (e.g. by re-shaping), pre-processing data (e.g. filtering) and so on.
The modelling step requires to find a model and its optimal parameters to fit the data, but also to fit the model output to the data and to validate the model. In the final Classification part, the task is to find classes and a classification rule according to which distinct data will be labeled.

[1] source: lecture notes by Emmanuel Zenou

There are two main forms of classification, *Supervised* and *Unsupervised*, of which both are described and explained in detail in the following chapters.

4.2 Supervised Classification

4.3 Unsupervised Classification

5