AGH University of Science and Technology

Department of Telecommunication

Image classification

based on visual descriptors

Introduction to image processing using Matlab and Open View

Theoretical fundamentals

*Image classification based on visual descriptors such as **SIFT**, **SURF**, **ORB**, **MPEG-7***

Expand of the technology forced new tools to describe photos, people need to compare or find photos and here comes visual descriptors. Visual descriptors are descriptions of the visual features of the contents in images or algorithms or applications that produce such descriptions. They describe elementary characteristics such as the shape, the color or the texture. Descriptors have a good knowledge of the objects and events found in image and they allow the quick and efficient searches of the visual content.

One of the most important feature of descriptors is that they should be invariant (ie. values for the same feature for different images, for example obtained using different camera orientations, should be the same), uniqueness (ie. Different features should have different descriptors), stability (ie. changes of the descriptor caused by small alterations of the described feature should be small) and the independence (ie. the individual components of the description should be functionally independent).

Descriptors are the first step to find out the connection between pixels contained in a digital image and what humans recall after having observed an image or a group of images after some minutes. In this instruction we will talk about only few descriptors such as SIFT, SURF and ORB.

MPEG-7 is a multimedia content description standard. It was standardized in ISO/IEC 15938 (Multimedia content description interface).This description will be associated with the content itself, to allow fast and efficient searching for material that is of interest to the user. MPEG-7 is formally called Multimedia Content Description Interface. Thus, it isnot a standard which deals with the actual encoding of moving pictures and audio, like MPEG-1, MPEG-2 and MPEG-4. It uses XML to store metadata, and can be attached totimecode in order to tag particular events, or synchronise lyrics to a song, for example.It was designed to standardize:

1) a set of Description Schemes ("DS") and Descriptors ("D")

2) a language to specify these schemes, called the Description Definition Language ("DDL")

3) a scheme for coding the description

The combination of MPEG-4 and MPEG-7 has been sometimes referred to as MPEG-47.

Texture browsing descriptor

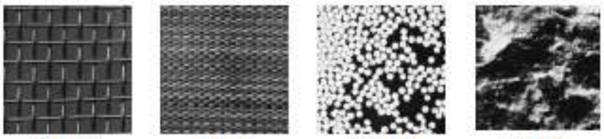
The texture information of an image is a fundamental visual feature, which has been studied during the last decade to analyze images in the areas of medical imaging and satellite imaging, etc.
This contains structureness, regularity, directionality and roughness of images, which are important properties of the content-based indexing of the image.
Previous works such as probability distribution of pixels, directional filtering and Markov random field have been studied. More recently, spatial Gabor filters and wavelet transformation have been studied to extract texture information. In, Gabor, Pyramid structured Wavelet Transform (PWT), Tree structured Wavelet Transform (TWT), and Multiresolution Simultaneous Autoregressive Model (MRSAR) methods have been compared.
The MPEG-7 homogeneous texture descriptor we invented is efficient not only for computing texture features but also in representing texture information.

$$M = \sum_{i=1}^{3} N_i * V_i$$

Ni is numbers of candidate projections

$$\begin{aligned}
vl &= 0 \quad \text{if} \quad M < 5, \\
vl &= 1 \quad \text{if} \quad 5 \le M < 10, \\
vl &= 2 \quad \text{if} \quad 10 \le M < 20, \\
vl &= 3 \quad \text{if} \quad M \ge 20
\end{aligned}$$

For bigger V1, structure of the texture is more regular. The picture below presents example of textures in dependent of V1



v1=3 v1=2 v1=1 v1=0

You should run the program from command line, supplying the necessary input arguments and the descriptors will be written to the output file you specified.

Firstly open command line by cmd, next using cd.. find location of mpeg7 folder.

Now the program is run as follows:

MPEG7Fex.exe **featureType featureParameters images outputFile**

feature Type how you can try:

- **CSD**
- **SCD**
- **CLD**
- **DCD**
- **HTD**
- **EHD**

CSD (Color Structure Descriptor) **-** a descriptor that uses no more storage than an ordinary colour histogram but that substantially out-performs it.

SCD (Scalable Color Descriptor) - is derived from a color histogram defined in the HueSaturation-Value (HSV) color space with fixed color space quantization. It uses a Haar transform coefficient encoding, allowing scalable representation of description, as well as complexity scalability of feature extraction and matching procedures.

CLD (Color Layout Descriptor) - is designed to capture the spatial distribution of color in an image. The feature extraction process consists of two parts; grid based representative color selection and discrete cosine transform with quantization.

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

HTD (Homogeneous Texture Descriptor) - is an important visual primitive for searching and browsing through large collections of similar looking patterns. If an image can be partitioned into a set of homogeneous texture regions, then the texture features associated with the regions can index the image data. Examples of homogeneous textured patterns are viewed from a distance parking lots with cars parked at regular intervals, or aagricultural areas and vegetation patches in aerial and satellite imagery.

EHD (Edge Histogram Descriptor) - Homogeneous texture is an important visual primitive for searching and browsing through large collections of similar looking patterns. If an image can be partitioned into a set of homogeneous texture regions, then the texture features associated with the regions can index the image data. Examples of homogeneous textured patterns are viewed from a distance parking lots with cars parked at regular intervals, or aagricultural areas and vegetation patches in aerial and satellite imagery.

For example:

MPEG7Fex.exe CSD 64 images.txt CSD.txt

You can use **image.txt** file with prepared pictures to do exercise.

The output file, CSD.txt (see descriptors/ directory), is generated as follows, one descriptor on each line.

```
imageFile    descriptorValues (64 values for CSD of size 64)
----------   ----------------------------------------------
Ordekler.jpg 61 0 0 0 3 0 0 0 86 75 2 21 0 0 2 0 181 96 0 0 0 0 5 0 119 144 96 76 98 175 23 0 113 204 59 2 31 26 0 0 140 114 47 15 131 87 18 3 112 94 20 3 59 42 3 3 11
ari-cicek2.jpg 4 44 0 0 0 0 0 0 19 11 114 84 98 4 0 0 0 0 0 0 109 108 39 8 121 164 124 51 46 121 92 0 8 18 27 0 69 161 168 78 104 118 60 11 86 94 62 0 35 56 56 0 59 88
agustos_bocegi.jpg 124 163 79 0 0 0 0 5 130 68 121 46 96 42 0 0 0 0 0 0 0 0 14 24 141 106 62 53 89 49 27 16 6 13 5 0 37 59 43 16 77 45 31 60 102 35 33 51 33 19 9 29 35
```

Default descriptor size is 64, allowable descriptor size values are **32, 64, 128, 256.**

In folder *code files* you can find program codes.

Output files documentation:

1) The output, DCD.txt (see descriptors/DCD.txt), is shown above.

2) Default descriptor size is 128, allowable descriptor size values are 16, 32, 64, 128, 256.

The output, SCD.txt (see descriptors/SCD.txt), is as follows (descriptor size=32):

Ordekler.jpg -148 75 0 49 -14 -12 9 21 -16 -13 -2 14 9 14 19 22 -2 5 -3 6 -1 5 0 0 -2 2 2 0 -3 5 1 -4
ari-cicek2.jpg -182 71 18 57 -3 3 17 21 -5 2 6 14 9 14 19 22 3 -2 -1 2 -1 5 0 0 -2 2 2 0 -3 5 1 -4
agustos_bocegi.jpg -167 77 5 52 -7 -7 14 23 -9 -8 3 16 9 14 19 22 3 -4 0 2 -1 5 0 0 -2 2 2 0 -3 5 1 -4

3) Defaults, numberOfYCoeff = 64, numberOfCCoeff = 28.Allowable parameters (numberOfYCoeff, numberOfCCoeff) are 3, 6, 10, 15, 21, 28, 64
The output file (CLD.txt) contains the CLD descriptors, assuming numberOfYCoeff = 6, numberOfCCoeff = 3, one per line, in the following order:
imageFile YCoeff_DC(1) YCoeff_AC(5) CbCoeff_DC(1) CbCoeff_AC(2) CrCoeff_DC(1) CrCoeff_AC(2) (vector of length 12)
First the DC coefficient, then the AC coefficients for 3 channels (Y, Cb, Cr) are written consecutively.
Example (see descriptors/CLD.txt, run as MPEG7Fex.exe CLD 6 3 imageList.txt descriptors/CLD.txt)

Ordekler.jpg 15 18 17 13 16 11 35 19 11 22 12 16
ari-cicek2.jpg 26 13 17 18 17 20 23 11 7 33 10 8
agustos_bocegi.jpg 47 25 30 25 9 25 15 13 28 29 17 21

4) Parameters:
+ normalizationFlag: if 1, normalize the values to MPEG-7 ranges (color:0-32, variance:0,1, weight:0-32) if 0, do not normalize (color:RGB values[0-255], variance:as computed, weight:0-100 percent) default: 1
+ varianceFlag: if 1, compute and write variance; if 0, do not compute/write variance, default: 1.
+ spatialFlag: if 1, compute and write spatial coherence; if 0, do not compute/write spatial coherence; default: 1
+ numBin1, numBin2, numBin3: number of bins for color quantization, defaults: 32, 32, 32.
Output file (DCD.txt) contains the DCD descriptors, one per line for each image, in the following order.
imageFile numberOfDominantColors spatialCoherency percentage_1 centroid_1_channel1 centroid_1_channel2 centroid_1_channel3 variance_1_channel1 variance_1_channel2 variance_1_channel3 percentage_2 centroid_2_channel1 centroid_2_channel2 centroid_2_channel3 ...
Example (see descriptors/DCD.txt)
Ordekler.jpg 7 14 8 9 15 19 0 0 0 4 3 4 4 0 0 0 4 9 12 9 0 0 0 2 24 22 19 1 0 1 3 8 11 13 0 0 0 5 8 8 7 0 1 0 2 11 17 17 0 0 1
ari-cicek2.jpg 8 4 2 4 4 3 1 0 1 10 12 18 10 0 0 0 4 16 9 15 1 0 0 2 13 13 6 1 0 1 2 12 11 11 0 0 1 4 22 16 22 0 0 1 2 22 21 13 1 0 1 2 17 17 16 1 0 1
agustos_bocegi.jpg 8 9 3 3 4 3 1 0 0 14 31 31 31 0 0 0 2 10 15 5 1 0 1 1 23 12 7 1 1 1 2 18 29 5 0 0 1 2 7 9 4 1 0 1 2 14 9 8 1 1 1 2 13 20 5 1 1 1
There might be up to 8 dominant colors in an image, or only one! Hence the descriptors for each image may differ in size.That is why "the number of dominant colors" is saved as the first value for the ease of reading..

5) layerFlag: if 1, full layer (compute & write both energy and deviation); if 0, base layer (compute & write energy, but not energy deviation) default: 1
Output file (HTD.txt) contains the HTD descriptors, one per line for each image, in the following order.
imageFile mean (1) std(1) energy (30) energyDeviation(30) (a vector of length 62 if full layer)
Example (see descriptors/HTD.txt)
Ordekler.jpg 86 61 201 179 163 142 169 195 194 150 135 121 137 157 156 125 103 105 94 121 119 73 58 85 58 58 96 16 0 54 0 5 201 181 165 137 170 192 187 148 133 118 125 146 146 119 94 92 85 116 117 58 56 73 55 53 104 8 0 48 0 1
agustos_bocegi.jpg 176 195 238 239 235 235 202 191 216 200 202 206 208 208 183 193 159 191 161 190 153 161 156 167 161 176 149 141 136 142 132 141 237 239 233 232 197 185 206 192 202 202 204 204 166 186 141 177 144 185 141 134 153 149 157 176 141 117 111 137 102 123

6) The output file (EHD.txt) contains EHD descriptors of length 80 for each image.
Example (see descriptors/HTD.txt)
Ordekler.jpg 0 6 2 5 0 1 3 5 2 0 1 4 2 4 0 0 1 1 1 0 0 6 2 2 3 2 4 5 3 2 1 4 5 6 4 1 6 5 5 1 1 7 2 3 5 2 4 7 6 4 3 4 5 6 5 2 5 2 4 3 0 6 0 1 1 1 7 1 3 1 2 6 3 5 4 1 7 2 6 2
ari-cicek2.jpg 0 0 0 0 0 2 3 5 4 1 0 5 1 1 2 0 2 1 1 0 2 0 2 2 1 4 4 6 6 4 2 4 7 5 3 3 3 6 7 0 3 1 5 5 2 5 2 6 6 5 4 3 7 6 4 6 2 7 4 3 2 1 3 4 2 6 1 5 6 5 6 1 5 6 4 6 1 6 5 4
agustos_bocegi.jpg 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 3 1 2 1 5 4 2 5 3 3 3 5 5 0 4 0 1 0 1 3 6 0 2 3 2 6 4 7 3 5 3 6 4 2 2 2 6 2 2 4 6 2 4 5 0 4 7 5 1 1 1 7 3 1 2 1 7 4

SIFT

The SIFT detector extracts from an image a collection of frames or keypoints. These are oriented disks attached to blob-alike structures of the image. As the image translates, rotates and scales, the frames track these blobs and thus the deformation.

SIFT transofmation consist of five steps:

1) detection characteristic area in both fields, scale and placement in picture
2) localization of keypoints
3) match the scale
4) match orientation of angle
5) designate descriptor for keypoint

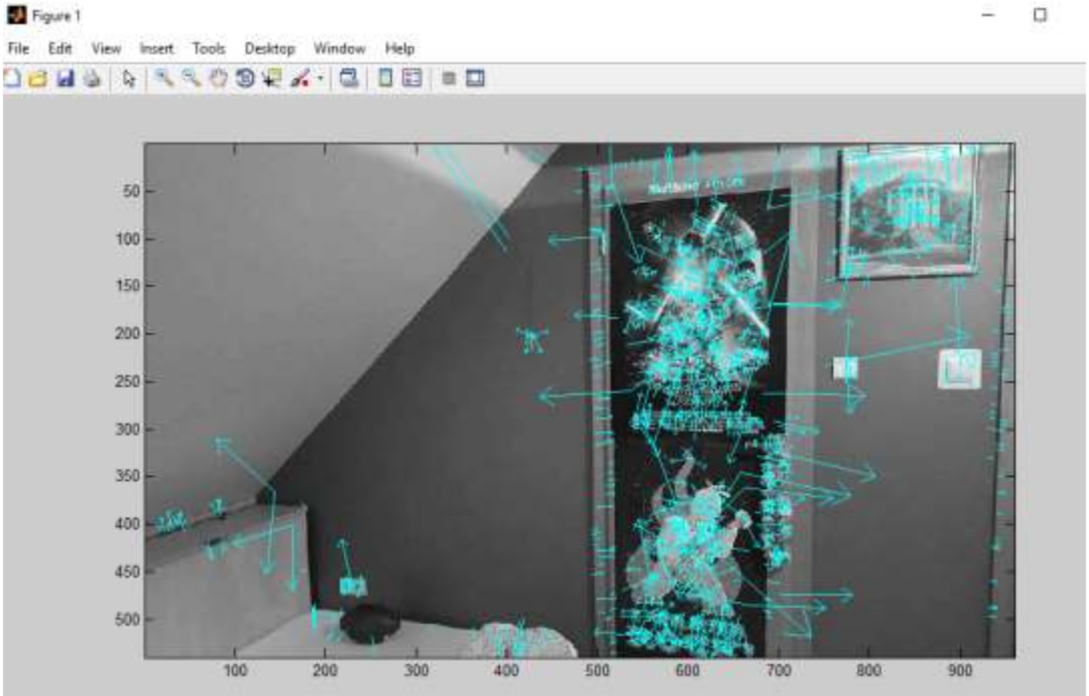
We begin by detecting points of interest, which are termed keypoints in the SIFT framework. The image is convolved with Gaussian filters at different scales, and then the difference of successive Gaussian-blurred images are taken. Keypoints are then taken as maxima/minima of the Difference of Gaussians (DoG) that occur at multiple scales. Specifically, a DoG image $D(x, y, \sigma)$ D is given by

$$L(x,y,\sigma) = G(x,y,\sigma) * I(x,y)$$
$$G(x,y,\sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}$$

The SIFT detector and the SIFT descriptor are invoked by means of the function sift, which provides a unified interface to both.

Unzip package ***sift.rar*** and import m-files to matlab.
Run Matlab in the current directory and execute the following commands. The "sift" command calls the appropriate binary to extract SIFT features and returns them in matrix form.
You can change ratio in match.m-file.
default ratio: 0.5
Use "showkeys" to display the keypoints superimposed on the image:

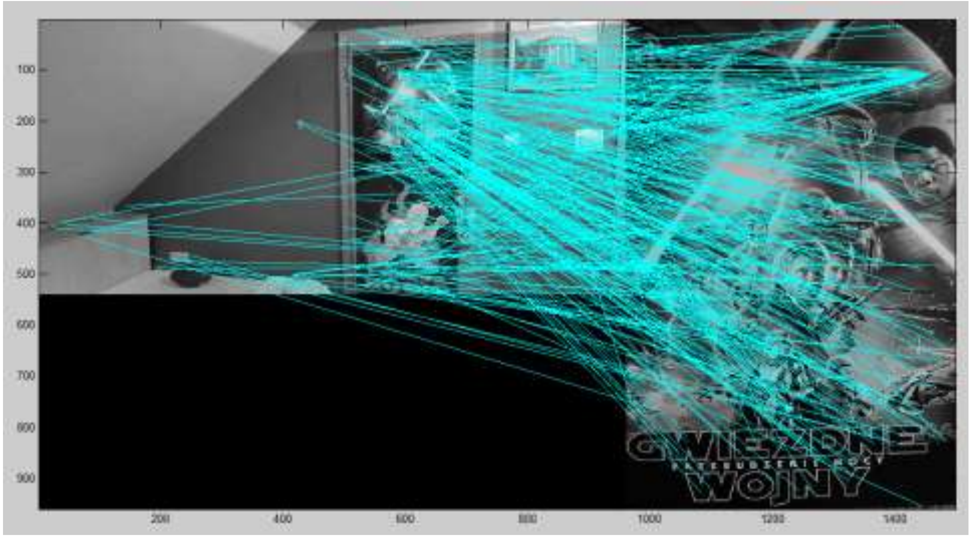
```
[image, descrips, locs] = sift('obraz1.pgm');
showkeys(image, locs);
```



The "match" command is given two image file names. It extracts SIFT features from each image, matches the features between the two images, and displays the results.

match('obraz1.pgm','obraz2.pgm');

ratio: 0.9



ratio:0.2

Other pairs of pictures to test are in main catalog SIFT.

SURF

Speeded Up Robust Features (SURF) is a local feature detector and descriptor that can be used for tasks such as object recognition or registration or classification or 3D reconstruction. It is partly inspired by the scale-invariant feature transform(SIFT) descriptor. The standard version of SURF is several times faster than SIFT and claimed by its authors to be more robust against different image transformations than SIFT.

To detect interest points, SURF uses an integer approximation of the determinant of Hessian blob detector, which can be computed with 3 integer operations using a precomputed integral image. Its feature descriptor is based on the sum of the Haar waveletresponse around the point of interest. These can also be computed with the aid of the integral image.

SURF descriptors can be used to locate and recognize objects, people or faces, to make 3D scenes, to track objects and to extract points of interest.

Below you have commands to find keypoints for **SURF**. Write the rest of program based on previous code.

Read image and detect interest points.

```
I = imread('localisation image');
points = detectSURFFeatures(I);
```

Display locations of interest in image.

```
imshow(I); hold on;
plot(points.selectStrongest(10));
```

To run this program it is required to install OpenCV. For properly work, pictures should be in the same folder as program. After run it will take pictures automaticly and on the screen you should see pictures with keypoints.

ORB

ORB (Oriented FAST and Rotated BRIEF) is a fast robust local feature detector, first presented by Ethan Rublee et al. in 2011, that can be used in computer vision tasks like object recognition or 3D reconstruction. It is based on the FAST keypoint detector and the visual descriptor BRIEF (Binary Robust Independent Elementary Features). Its aim is to provide a fast and efficient alternative to SIFT and SURF.

Detects keypoints and computes ORB descriptors for them

```
keypoints = cv.ORB(im)
keypoints = cv.ORB(im, 'OptionName', optionValue, ...)
[keypoints, descriptors] = cv.ORB(...)
```

```
descriptorSize = cv.ORB('DescriptorSize')   % descriptor size (32)
```

Write the rest of program based on **SIFT** code.

To run this program it is required to install OpenCV. For properly work, pictures should be in the same folder as program. After run it will take pictures automaticly and on the screen you should see pictures with keypoints.